School of Electrical Engineering and Computing
Department of Computing

Automated Framework for Robust Content-Based Verification of Print-Scan Degraded Text Documents

Yaniv Shulman

This thesis is presented for the Degree of
Master of Philosophy
of
Curtin University

December 2012

To the best of my knowledge and belief this thesis contains no material previously published by any other person except where due acknowledgement has been made. This thesis contains no material which has been accepted for the award of any other degree or diploma in any university.

____ ____________________

Yaniv Shulman Date

Abstract

Fraudulent documents frequently cause severe financial damages and impose security breaches to civil and government organizations. The rapid advances in technology and the widespread availability of personal computers has not reduced the use of printed documents. While digital documents can be verified by many robust and secure methods such as digital signatures and digital watermarks, verification of printed documents still relies on manual inspection of embedded physical security mechanisms.

The objective of this thesis is to propose an efficient automated framework for robust content-based verification of printed documents. The principal issue is to achieve robustness with respect to the degradations and increased levels of noise that occur from multiple cycles of printing and scanning. It is shown that classic OCR systems fail under such conditions, moreover OCR systems typically rely heavily on the use of high level linguistic structures to improve recognition rates. However inferring knowledge about the contents of the document image from a-priori statistics is contrary to the nature of document verification. Instead a system is proposed that utilizes specific knowledge of the document to perform highly accurate content verification based on a Print-Scan degradation model and character shape recognition. Such specific knowledge of the document is a reasonable choice for the verification domain since the document contents are already known in order to verify them. The system analyses digital multi font PDF documents to generate a descriptive summary of the document, referred to as "Document Description Map" (DDM). The DDM is later used for verifying the content of printed and scanned copies of the original documents. The system utilizes 2-D Discrete Cosine Transform based features and an adaptive hierarchical classifier trained with synthetic data generated by a Print-Scan degradation model. The system is tested with varying degrees of Print-Scan Channel corruption on a variety of documents with corruption produced by repetitive printing and scanning of the test documents. Results show the approach achieves excellent accuracy and robustness despite the high level of noise.

Acknowledgements

I would like to thank my supervisors, Dr. Patrick Peursum and Professor Tele Tan for the continuous support and guidance in all aspects of my research. In particular I want to thank Dr. Patrick Peursum for going way beyond any expectation I might have had, his help and honest concern is exceptional and something I will forever be grateful for.

Special thanks goes to my beloved wife and daughter for their patience and support, and for accepting my frequent absence, without your support I would not have been able to complete this thesis.

And finally, thanks to the many individuals who have spent time discussing and making constructive comments about this work throughout the course of this research.

Contents

List of Figures

List of Tables

Chapter 1

Introduction

Fraudulent documents frequently cause severe financial damage and impose security breaches to civil and government organizations. In a typical year, about 75,000-100,000 fraudulent documents are intercepted at U.S. ports of entry (Hesse, 1999; Kuklinski, 2004; Customs and Protection, 2009). The rapid advances in technology and the widespread availability of desktop and mobile personal computers has not reduced the use of printed documents as a primary means of identification, transferring information, archiving information such as records and contracts, and more (Sellen and Harper, 2003; Lyman and Varian, 2003). While digital documents can be authenticated by many robust and secure methods such as digital signatures and digital watermarks, authentication of printed documents still relies on physical security mechanisms such as: physical watermarks, security patterns, ultraviolet and near infrared security inks, magnetic inks, barcodes, micro printing, ghost photos, holographic patterns, embedding a person's picture, check digits, plastic seals, stamps, handwritten signatures and more (Vinicius *et al.*, 2007; Kuklinski, 2004). The significant visual impact caused by traditional authentication schemes is often an undesired side effect of these approaches. Furthermore, the aforementioned physical security mechanisms do not provide a solution in the case where the authors of the documents are interested in allowing the creation of physical printed copies for any purpose such as: archiving, distribution, usage as temporary medium prior to signing and/or transmission, and more, while still providing the author with an efficient and trusted way of verifying that the contents have not been changed by natural occurrences or by deliberate tampering.

The aforementioned physical security mechanisms for authenticating printed documents divide into two main categories. In the first and most common authentication category the media (mainly paper) on which the content is printed is annotated in a predetermined uniform way while ignoring the differences in the content between different documents of the same type. Authentication schemes under this category rely on generating and maintaining a visual standard shared across all documents of the same type which aim to make it more robust to tampering and counterfeiting and easier to authenticate. These documents are later authenticated and verified by humans who are expected to remember the subtle visual differences between many different types of documents which they encounter on a daily basis, a highly difficult mission (Kuklinski, 2004). Thorough manual authentication and verification of printed documents usually calls for specially trained professionals to operate specialized hardware such as microscopes. Such professional authenticators and the equipment needed are not available in most real life ad-hoc situations where authentication of printed documents is desired, thus creating significant shortcomings in the traditional physical authentication schemes.

In the second category unique identifiers are embedded on the document such as barcodes and other unique identifying elements (known as fingerprints) which allow uniquely identifying a copy (Zhu *et al.*, 2003). However barcodes and fingerprints are not usually related to the contents of the document and are used to only uniquely identify the medium.

Authentication of digital documents is performed by two main approaches: digital signatures and watermarking (Ming *et al.*, 2007). Digital signatures compute a digest of the digital file using a hash function. The digest is computed as a function of the document and the private encryption key which belongs to the issuing authority (Goldwasser *et al.*, 1988; Mao, 2003). Digital watermarking is the process of embedding information into the document in a way that is difficult to remove (Cox, 2008). A digital watermark is called imperceptible if the watermarked content is perceptually equivalent to the original content (Wikipedia, 2012; Khan and Mirza, 2007). A digital watermark is called fragile if it fails to be detectable after the slightest modification. Fragile watermarks are commonly used for tamper detection in digital documents. In general, it is easy to create robust watermarks or imperceptible watermarks, but the creation of robust and imperceptible watermarks has proven to be difficult (Cox, 2008).

However, existing approaches to authentication of digital documents do not adapt well to printed documents. The geometric transformations and added noise that are part of the printing and scanning process ultimately do not allow the authentication of printed and scanned documents by pure mathematical comparison to the digital document, nor will fragile watermarks survive the printing and scanning process to continue to be useful. Consequently, the content perceived in the document image is the only reasonable basis that may be used for automated verification of Print-Scan degraded documents. This approach is in fact similar to the way humans perceive symbols and images and ideally would only reject the document image if the geometric transformations in the document image are of such extent so that the elements change their meaningful interpretation.

Very little research has been conducted on such content-based print-and-scanned verification. Vinicius *et al.* (2007) suggested a content-based method which partially addresses this problem. They suggested using a general OCR engine in order to classify the characters in the printed and scanned document. General OCR is considered a solved problem for over two decades and typically is able to provide recognition of about 99% in good quality low noise modern document images (Rice *et al.*, 1995; Rose, 2009). While 99% accuracy may be acceptable for many character recognition applications it is not accurate enough for the purpose of verification. For example, consider that an average page contains roughly 2000 characters, an error rate of 0.5% means 10 errors per page. Also it is important to note that 99% accuracy is achieved by commercial general OCR systems only in quality scanned, low noise document images, with much lower results usually achieved in highly degraded document images.

Ming *et al.* (2007) suggested a content-based method which considers the shapes of different symbols that may appear in the document and does not rely on a particular character set as most OCR engines do. However, their method has a few drawbacks including a security flaw which allows an attacker to replace all of the occurrences of a character in the document with another character without being detected.

Considering the shortcomings in both methods suggested by Vinicius *et al.* (2007) and Ming *et al.* (2007), robust content-based verification of printed documents still remains an unsolved problem requiring further investigation.

1.1 Aims and Approach

The aim of this research is to develop an efficient automated framework for robust content-based verification of printed typed text documents which is based on analyzing an image of the presented document obtained through standard off-the-shelf printing and scanning devices. In particular, the suggested verification framework aims to significantly improve upon the error rate of contemporary general use OCR systems with a focus on text document images containing relatively high levels of Print-Scan corruption and noise generated by repeated printing and scanning of the same document.

The two main factors causing increased error rates in highly degraded document images are segmentation errors and classification errors (Baird, 2000; Eikvil, 1993). As the level of degradation increases, the frequency of the occurrence of broken and joined characters due to noise related transformation and erroneous binarization increases rapidly, which in turn results in many classification errors.

Thus the objectives of this thesis can be detailed as follows:

1. To research an efficient automated framework for robust content-based verification of Print-Scan degraded typed text documents images generated by standard off-the-shelf printing and scanning devices. The focus is on highly Print-Scan degraded documents images such as images generated after multiple repeating cycles of printing and scanning. Verification is performed for the purpose of detecting intentional tampering. This includes the evaluation of the effectiveness of different candidate features, classifiers and model-based synthetic training data for classifying noisy Print-Scan degraded characters.

2. To develop a method for accurate segmentation of highly degraded document images by utilizing prior document-specific knowledge. The method must be robust to Print-Scan induced degradations and transformations, but should support accurate verification in the presence of critical changes that occur in the document due to intentional tampering.

This thesis approaches the problem as one of analyzing the ideal digital documents textual contents, and storing the resulting document verification meta-data, referred to as "Document Description Map" (DDM). The information stored by the DDM is later used in the verification stage for: *(i)* aiding in robust segmentation by simplifying the tasks of separating joined characters, and joining broken characters; *(ii)* providing font type information to an adaptive classifier which assists in further lowering classification error rates; *(iii)* for verifying the content of printed and scanned images of the original document by comparison of the documents DDM against the current state of the document image.

Furthermore, OCR systems typically use high level linguistic models such as n-gram statistics and dictionaries to improve recognition accuracy (Govindan and P., 1990). Using linguistic analysis mechanisms such as dictionaries and n-grams statistics is in fact equivalent to the action of inferring what would be the most reasonable content in the document. However, statistical "best guess" defy the objective of this thesis, which is to verify that the documents content has not changed, and therefore any use of such methods must be avoided completely as a method of correcting segmentation and classification errors.

1.2 Significance and Contributions

This thesis makes two main contributions to the field of computer vision:

1. The development of a robust and accurate document image segmentation method in

Print-Scan Channel degraded document images suffering from significant amounts of noise and symbol degradation due to repeated print-scan cycles.

2. Establishing the region-based 2-D Discrete Cosine Transform (DCT) calculated from the entire character region as an accurate shape feature highly robust to typical noise and degradations generated by the Print-Scan Channel.

Together, these two contributions are used to build and evaluate a highly accurate document verification system that is shown to outperform state-of-the-art OCR on print-scan degraded documents.

1.2.1 Robust and Accurate Document Image Segmentation Method

Recognition of characters in degraded documents images is challenging due to the high frequency of broken and joined characters occurrence (Droettboom, 2003). Joined and broken characters contribute substantially to classification errors and therefore must be dealt with in order to obtain high accuracy (Casey and Lecolinet, 1996). Being such a critical step of the recognition process, this thesis proposes a highly accurate and robust segmentation method based on prior document information. The proposed method is used for separating joined characters and joining broken characters by utilizing priori document contents information and classifier feedback. The suggested method substantially increases the accuracy of segmentation over pure statistical methods typically used by OCR systems, and therefore contributes substantially to the increased accuracy of the entire system.

1.2.2 Region-based 2-D DCT as an Accurate and Robust Shape Feature

One of the fundamental and most important steps in content-based verification of document images is the accurate classification of the characters present in the documents, as they are the fundamental building blocks of text documents. A major contribution of this thesis is a quantitative analysis of the effectiveness of several fast shape descriptors under conditions of increasing print-scan degradation, showing that region-based 2-D DCT II is an accurate and robust shape feature achieving overall excellent accuracy with various classifiers trained with model-based synthetic generated training data. The DCT based features displayed good resilience to the substantial Print-Scan noise present in the tested documents images.

1.3 Structure of the Thesis

This thesis is organized as follows. In Chapter 2, literature review of related concepts and methods such as shape descriptors, document image degradation models and classifiers is given.

Existing 2-D shape recognition methods are briefly explored and the properties of different suggested shape descriptors, and hence their suitability for allowing the accurate classification of characters in Print-Scan degraded documents images is discussed.

This is followed by a discussion of established classification methods and their properties that are used in conjunction with the shape descriptors to classify the shapes present in the document images.

Next a discussion is given on the effects of the printing-and-scanning process on recognition. The recognition is substantially enhanced by providing to the classifier examples of the degraded characters as they might appear in the verified documents. These examples can be taken from "real" world collected data or by synthetically generating such data. This Section discusses both options, their advantages and disadvantages, and degradation models that enable the generation of such synthetic training data.

In Chapter 3, framework overview of the proposed system for robust document image verification and authentication is presented. The main components of the framework are introduced and described, and their interaction is presented.

In Chapter 4, the recognition accuracy of characters in noisy Print-Scan degraded documents is evaluated for different shape descriptors and classifiers combinations. The Chapter commences by defining selection criteria for choosing candidate shape descriptors, and discusses the relevant properties of the selected shape descriptors, which are leading to their selection. Following is a discussion of the generation of suitable training data for improving classification. And a physical-based parameterized model for generating synthetic Print-Scan degradation modeled training data is discussed. Finally tests are performed on a uniform test set taken from real scanned documents images to achieve comparative results for the accuracy of the different descriptors-classifiers combinations.

In Chapter 5, a detailed discussion is given on the suggested methods for performing verification of noisy documents images. The discussion commences by presenting the typical characteristic phenomenon occurring in degraded document images. Following are the descriptions of the varied methods used for performing verification and for overcoming the characteristic difficulties found in noisy documents images. Following, three experiments are performed to provide an empirical evaluation of the proposed framework. The results of the experiments are presented and discussed.

Finally, Chapter 6 provides a summary of the thesis, its contributions and potential future advancements and research possibilities.

Chapter 2

Background

This Chapter presents an overview of common computer vision related techniques that are used in various applications including OCR systems and in previously suggested systems for verification of text documents in particular. The Chapter commences by discussing 2-D shape recognition, a crucial step in numerous applications in computer vision in general and in the framework for automated verification of text documents suggested in this thesis. The properties of several shape features are discussed and the presented features are grouped according to different criteria. Furthermore, shape descriptors are discussed in the context of character recognition and of OCR systems. The Chapter continues by providing detailed theoretical presentation of a number of popular classification techniques including Naive Bayes, k-Nearest Neighbor, Support Vector Machine, Random Forrest and Artificial Neural Network. In addition, the generation of both real and synthetic training data is discussed along with a review of the suggested methods for modeling degradation for classifier training. Following this is a review of the previously suggested document verification and authentication systems in the literature. And finally, document character segmentation is briefly discussed.

2.1 2-D Shape Recognition

This Section commences by providing an overview the different categories and properties of shape descriptors. Following this is an in-depth discussion on a number of shape descriptors that are referred to in the following Chapters. Finally this Section concludes with a discussion on the usage of shape descriptors by commercial OCR systems and other methods which were suggested in the literature.

2.1.1 Overview

Experimental results in human psychophysical research suggest that the “shape” of an object, which is usually defined by its boundary or region, contributes the most to understanding the meaning of the presented image, much more than other features as texture and colour. In fact, if the only information presented in an image is the boundaries of the objects, it is usually sufficient for image content understanding (Milan *et al.*, 2007). Despite of the fundamental importance of "shape" to the perception and understanding of the world by humans it is a very difficult task to accurately describe shapes linguistically. Pure mathematical description of shapes is achieved in many different ways however none of the proposed methods has been generally accepted. Moreover, current effective syntactic and mathematic descriptions of shapes are unusable for shape recognition tasks (Milan *et al.*, 2007). Since recognition of characters is an important part of this thesis, the remainder of this Section will discuss shape recognition in the context of character recognition.

Characters are a well-defined set of 2-D shapes and more often than not, when characters are printed to form a document they share a uniform fill texture and colour. Therefore, under the assumption that it is possible to successfully segment a document into the characters comprising it, a major part of the task of character recognition in document images is essentially a special case of 2-D shape recognition. This is in fact the predominant approach in existing OCR systems and suggested document analysis and verification systems (Mori *et al.*, 1995; Trier *et al.*, 1996a; Vinicius *et al.*, 2007; Ming *et al.*, 2007).

2-D shape recognition is a fundamental part of solving a myriad of problems in computer vision such as scene understanding, OCR, biometric recognition, medical imaging and content-based retrieval of documents and pictures amongst many others, and as such it is still a highly active field of research. Commenced in the middle of the previous century, 2-D shape recognition has materialized in many suggested methods in the literature, some of which are to be reviewed in later parts of this Chapter.

The task of 2-D shape recognition in images can be generally summarized as follows:

1. Decide on the method for extracting and computing the descriptive features for the shapes. These features are commonly referred to by the term "shape descriptors".
2. Choose one or more classifier(s) and train it/them with shape descriptors extracted from a training set.
3. Perform low level pre-processing to the image such as noise suppression and other local enhancements.
4. Segment the image into shape components.
5. Perform additional low level pre-processing if required to do so in order to satisfy

requirements imposed by the shape descriptors such as: thinning, boundary extraction, skeletonization etc.

6. Extract/compute the shape descriptors for each of the shapes.
7. Feed the shape descriptors into one or more classifier(s) trained to discriminate between different shapes of interest.
8. Label the segmented shape components with the output obtained from the classifier(s).

In Yazici and Sener (2003) feature extraction is defined as "extracting from the raw data the information which is most relevant for classification purposes, in the sense of minimizing the within-class pattern variability while enhancing the between-class pattern variability". Moreover, selecting the most suitable shape descriptor is considered to be the most influential step on the final features dataset and thus on the overall recognition accuracy. With that said, it is important to stress that selecting the best feature descriptor is not the only significant step in building a 2-D shape recognition system. The other steps also need to be optimized in order to achieve the best possible recognition rate (Trier *et al.*, 1996a).

The task of choosing a suitable shape descriptor is further complicated by a number of factors such as (Trier *et al.*, 1996a; Latecki, 2005):

- The large number of suggested methods in the literature.
- The fact that different descriptors might be more suitable for different applications and particular classes of shapes.
- The significant variations in performance which are reported when different classifiers are used with the same shape descriptor.
- Different descriptors have different properties such as invariance to some of the affine transformations such as translation, rotation and scaling.
- Relying on the descriptors' reported success rate as a comparative measure of suitability is not practical since in many cases different classifiers, different training sets and different testing sets were used for obtaining classification accuracy results for the proposed shape descriptors.

In the following Sections the focus is on feature extraction methods and related concepts for the purpose of 2-D shape recognition.

2.1.2 Shape Features Groups

Shape descriptors extraction methods can be generally divided into the following groups (Veltkamp and Hagedoorn, 1999; Milan *et al.*, 2007; Latecki *et al.*, 2000):

- Voting schemes.
- Global shape descriptors
  - Region-based.
  - Contour-based.
  - Skeleton-based.

2.1.2.1 Voting Scheme Shape Descriptors

Voting scheme shape descriptors consider only a set of interest points or sub regions on the shape's boundary or in the shape's region. An example for a suggested approach which uses this technique is the basic peephole method in which certain interest points in the image "light up" if they are covered by foreground pixels. Different combinations of zones which are covered by foreground pixels correlates to different shape classes. This was in fact the method used in "Electric Reading Automation" (ERA) by Solatron Electronics Group LTD, one of the earliest commercial OCR systems but has since been abandoned in favour of more accurate region or boundary based techniques (Mori *et al.*, 1995).

2.1.2.2 Global Shape Descriptors

Global shape descriptors consider the entire shape region or boundary as a whole when extracting the descriptor. The main disadvantages of this approach as Veltkamp and Hagedoorn (1999) points out, are that the shape needs to be accurately segmented and extracted from the image before calculation of the descriptors, which is usually a complicated problem in itself, and the relatively high sensitivity to occlusions.

Region-Based Shape Descriptors Global region-based shape descriptors consider the entire region of the shape when calculating the descriptors (Milan *et al.*, 2007). These include global shape image transforms from the pixel space domain to the frequency domain such as the 2-D discrete Fourier transform, 2-D discrete cosine transform, 2-D wavelet transform and more. Another popular approach is calculation of geometric (Flusser and Suk, 1993), Zernike (Teague, 1980) and Legendre (Hosny, 2007) affine moment invariants and using the low order moment coefficients as entries in a feature vector, as they are less sensitive to noise. Additional region-based shape features such as: top-bottom or left-right ratios, major and minor axis orientation, Euler number, compactness and more can be calculated from the entire shape region. An advantage of the region-based approach over the contour-based and the skeleton-based approaches is that the colour or grey intensity values of the shape's region pixels are maintained and available for consideration. However, these descriptors usually do not convey any local information about the shape.

Contour-Based Shape Descriptors Global contour-based shape descriptors consider only the outer boundary of the shape when calculating the descriptors (Milan *et al.*, 2007). It is not conclusive if the shape's region indeed contain more information about the shape than the contour of the shape (provided that the colour channels or the intensity values of the shape's region pixels are not of interest), or which approach is superior. And indeed, there are many contour-based shape descriptors suggested which consider only the contour, or in other words, the boundary of the shape. In fact, it is claimed by Zhang and Lu (2005) that contour-based methods are the predominant approach. According to Zhang and Lu (2005) it is easier to derive contour-based methods since contours can be represented by a one dimensional signal therefore making it more approachable and more stable numerically than the region-based approaches that involve two-dimensional signals.

Example for contour-based shape descriptors is the commonly used and thoroughly studied Fourier Descriptors (Zahn and Roskies, 1972) and its shape signature based variants such as Multiscale Fourier Descriptors (Kunttu *et al.*, 2003), Chord-Length Functions Fourier Descriptors (Wang and Shi, 2006), Position Function (complex plane coordinates) Fourier Descriptors (Zhang and Lu, 2005) and many other; other suggested methods are Inner Distance (Ling and W. Jacobs, 2007); Chain Codes (Milan *et al.*, 2007), Curvature Scale Space Contours (Mokhtarian, 1995), and geometric boundary representations such as Hierarchical Polygonal Approximation (Veltkamp, 1998), and more.

As mentioned above, an advantage of contour-based methods is that a shape's contour can be conveniently represented by a one dimensional signal. This compactness in the representation of a shape offers substantial savings in the amount of data which needs to stored and processed. On the other end, this creates a disadvantage in the form of increased sensitivity to noise over the region-based representation since the boundary of the shape is more susceptible to deformations caused by noise. In addition, points which are close in the spatial domain may be far away in the boundary representation of the shape which creates a problem in reflecting the relationship between neighboring points (Milan *et al.*, 2007).

Skeleton-Based Shape Descriptors Skeleton-based methods are the result of another approach which is by far not as common as the boundary-based and region-based approaches. These methods extract information from a skeleton of the shape region usually generated by mathematical morphology thinning techniques or by the medial axis transform of the shape's region (Milan *et al.*, 2007; Blum, 1973). The resulting skeletons are than usually transformed into a graph and are measured in relation to other observations by a graph similarity measure. An example of such method is given in Latecki *et al.* (2000) which compares skeletons of shapes by converting the skeleton into a directed acyclic graph and use a graph matching algorithm to obtain a dissimilarity measure.

Skeletons of regions are very sensitive to changes in the boundary and as a result small variations in the boundary of the shape can cause a substantial impact on the shape's skeleton. To mitigate this problem a few skeleton generation techniques have been suggested that are less sensitive to changes in the shape's boundary such as scale-space approach presented in Maragos (1989) and varying smoothing presented in Wright and Fallside (1993). Another major problem in skeleton representation of the shape's region is that non-similar simple shapes might have the same skeleton.

2.1.3 Shape Features Properties

Different shape descriptors have different approaches to considering and analyzing shapes therefore they often have different properties. The difference in the properties between the different shape descriptors implies they are influenced differently by changes and degradations caused by noise, occlusions, transformations and more. A list of the prominent properties of shape descriptors is now given and discussed briefly (excluding robust and accurate retrieval rates which are obvious criteria):

- Prerequisites.
- Invariance.
- Global, local and hierarchical information representation.
- Reconstruction ability.

- Compactness.
- Generality.
- Clarity.
- Computational complexity.

2.1.3.1 Prerequisites

As discussed in Section 2.1.2 different shape descriptors operate on different representations of the shape such as the boundary, the entire region or the skeleton. These require different pre-processing of the image in order to prepare the shape for analysis and extraction of features. Additional prerequisites are binarization of the image which is required by a large number of shape descriptors, and coupling with certain types of classifiers or specialized classifiers.

2.1.3.2 Invariance

Since different people may classify a shape to different classes, it indicates that the decision whether a shape belongs to one class or the other is a subjective decision. Also deciding at what level of deformation a shape no longer belongs to its “true” class is often a non-conclusive decision for humans. Despite the subjective nature of classifying deformations of shapes and the difficulty to understand and define this process mathematically in an accurate way (Machajdik and Hanbury, 2010) it is expected that a good shape descriptor be invariant to intra-class shape variations, reasonable deformations caused by linear and nonlinear transformations and especially be invariant to characteristic noise which is relevant to the application domain. Real world data intrinsically suffers from inaccuracies such as calibration of devices, different lighting, sampling jitter, numerical rounding, misplacements as well as rotations of the objects and more.

In particular some linear transformations such as: scale, translation minor skew and rotation are considered important since they preserve the shape and therefore invariance to these transformations is considered to be an important property when comparing different shape descriptors. Invariance to rotation may be required only to a certain degree, depending on the application, or may be waived completely under the assumption that rotations would be fixed to a normalized orientation by pre-processing techniques such as the Principal Axes Transformation (Alpert *et al.*, 1990). Invariance to noise is also a very important consideration since some degree of noise is present in almost all practical applications and moreover reasonable amounts of noise can also be considered to preserve the shape.

2.1.3.3 Global, Local and Hierarchical Information Representation

Different shape descriptors provide information on different scopes such as global, local and hierarchical scale representation. Local or hierarchical representations might be a desirable property for systems which are interested in shape information across different scales, usually with increasing resolution, or to be able to make a granular local distinction between shapes. This property may be useful for systems which increase efficiency by quickly ruling out invalid options on a coarse scale.

An example of a global shape descriptor is the global image transforms such as 2-D Fourier transform results in a set of coefficients that provide frequency information about the entire shape and therefore do not provide any local shape information. In contrast, the Curvature Scale Space method presented in Mokhtarian (1995) is an example of a descriptor which provides information on a shape in a hierarchical scope which allows discriminating between shapes on different resolution scales.

2.1.3.4 Reconstruction Ability

A shape descriptor is said to have the reconstruction ability property if the original shape can be reconstructed from the derived descriptors.

This property holds if and only if isomorphism exists from the shape space into the feature space. This implies that the features are unique for each shape regardless of how small the variations are between two similar shapes. It further implies that some direct and distinct physical or mathematical connection exists between the shape and the descriptor.

Although for some description methods, exact reconstruction of the original shape from the features may require an arbitrarily large or infinite number of features, it is possible to perform partial reconstruction to different levels of fidelity and detail by taking a subset of the features. This has the implication that some of the features contain more “relevant” data about the shape than others.

Examples of such descriptors are the global image 2-D Fourier transform, 2-D Discrete Cosine Transform and the boundary chain code representation. An example for a descriptor which does not hold this property is the geometric affine moment invariants (Flusser and Suk, 1993) since the monomials on which the shape is projected upon are not orthogonal and therefore they do not provide the ability to uniquely reconstruct a shape from the descriptors (Teague, 1980).

2.1.3.5 Compactness

This property relates to the dimensionality of the data which is required in order to represent a shape. In some features, the amount of data required for representation is correlated with the amount of granularity of detail that is desired which means that the dimensionality of the data can be increased in order to provide finer level details.

Lower dimensionality results in lower storage requirements, faster computations, ease of analysis and understanding and does not suffer from the peaking phenomenon and the infamous “curse of dimensionality” as badly as high dimensional data (Sima and Dougherty, 2008; Bellman, 1957). Furthermore, to a limit, as the dimensionality of the descriptors decreases, the size of the training required to obtain the same accuracy also decreases. Therefore, compactness (or in other words lowered dimensionality) of the descriptor vector is highly desirable.

2.1.3.6 Generality

Generality is a measure of the scope of different applications and problem domains for which the suggested shape features are applicable. Low generality (or in other words high specialization) of shape features generally implies that the performance decreases quickly outside of the domain for which they have been developed. Therefore shape features with low generality are usually computationally less stable and predictable in the way they respond to deformations of shapes.

2.1.3.7 Clarity

Clarity refers to intuitive and simple derivation of the descriptors. Descriptors which are developed with a straightforward mathematical or physical approach usually result in having a smaller number of parameters, which leads to less tuning and less prone to errors in parameter selection and hence higher generality. Moreover, clearly derived shape descriptors are generally easily and intuitively understood. Clarity may also relate to the similarity measure being intuitive and clear. A simple and intuitive descriptor would not be considered to possess the clarity property if the similarity measure used to distinguish between different shapes is very complicated or unintuitive.

2.1.3.8 Computational Complexity

Computational complexity relates to the number of computations and therefore the running time that is required for generating the shape features and for calculating the similarity measure. Since many practical applications may involve a large set of shapes to classify, and usually a far larger set of observations in the training set which need to be considered during classification, computational complexity is one of the most important properties to consider when selecting a shape descriptor for a practical application.

2.1.4 Shape Descriptors Used in Character Recognition

Follows is in depth description of three shape descriptors which were proven to previously give satisfactory results in OCR applications (Trier *et al.*, 1996b; Öztürk *et al.*, 2001; Amanatiadis *et al.*, 2011). In addition one of the methods (Fourier Descriptors) was used by Ming *et al.* (2007) in their suggested document images verification system.

- Geometric Affine Moment Invariants (GAMI).
- Centroid Distance Function Fourier Descriptors (FD).
- 2-D DCT II (DCT).

2.1.4.1 Geometric Affine Moment Invariants

Geometric Affine Moment Invariants (GAMI) interpret a shape's image as a normalized probability density of a 2-D random variable (Milan *et al.*, 2007). The shape signal is projected onto different monomials to generate various statistical measures of the shapes. Initially presented by Hu in Hu (1962), and corrected by Reiss (Reiss, 1991), the theory of

GAMI has been further developed by Maitra (1979), Li (1992), Flusser and Suk (1993) and others to be invariant to general affine transformations such as rotation, scale , translation, skew and more. GAMI are usually applied to a shape's region, however they may be applied to a shape's boundary as well in order to extract the shape descriptor.

There are a few inherent weaknesses to using geometric moments as a shape description method. First to note are the non-orthogonal properties of the monomials which decrease the separation between different shapes representations in the transformed feature space. Secondly, there is no understanding of what the higher-order moments represent in the spatial domain, and as the order of the moments increases, each pixel contributes substantially to the summation, making the higher order moments very sensitive to noise and deformations (Zhang and Lu, 2004).

GAMI were applied successfully to general shape recognition tasks and OCR applications in particular in various languages such as English, Arabic and others (Trier *et al.*, 1996b; Flusser and Suk, 1994, 1993; El-Khaly and Sid-Ahmed, 1990; Milan *et al.*, 2007; Veltkamp and Hagedoorn, 1999; Nagy, 1992). They were studied extensively and there are many papers which compare them to other shape recognition methods with satisfactory results (Dhanya and Ramakrishnan, 2002; Trier *et al.*, 1996a; Amanatiadis *et al.*, 2009; Mehtre *et al.*, 1997; Amanatiadis *et al.*, 2011).

There are two parameters which are to be decided when using GAMI as shape descriptors. The first is the order and number of features to use. Combining the problematic nature of the higher-order moments with the fact that low order moments do not provide sufficient discrimination when the number of classes is large implies it is not a trivial task. The second is the optional segmentation of the shape to subareas on which the moment calculation is applied. Extracting features from the entire shape may result in substantial loss of detail and on the other hand segmenting a shape to areas which are too small causes the features to be too sensitive to local deformations and possible noise (Dhanya and Ramakrishnan, 2002).

For full description of the general theory of moments invariants see Flusser and Suk (1993) and Reiss (1991).

The centralized moment of order $(p+q)$ denoted by m_{pq} for an image f with dimensions MxN and pixel coordinates x, y is given in Equation (2.1):

$$m_{pq} = \sum_{x=1}^{M} \sum_{y=1}^{N} (x - x_c)^p (y - y_c)^q f(x, y) \tag{2.1}$$

where x_c and y_c denote the shape's region center of mass coordinates which are given by:

$$x_c = m_{10}/m_{00} \qquad\qquad y_c = m_{01}/m_{00} \tag{2.2}$$

In this thesis, four centralized moment based invariants were calculated for the purpose of creating the shape descriptor feature vector as proposed in Flusser and Suk (1993) and Flusser and Suk (1994):

$$\begin{aligned}
l1 =& (m_{20}m_{02} - m_{11}^2)/m_{00}^4 && (2.3)\\
l2 =& (m_{30}^2 m_{03}^2 - 6m_{30}m_{21}m_{03} + 4m_{30}m_{12}^3 + 4m_{21}^3 m_{03} - 3m_{21}^2 m_{12}^2)/m_{00}^{10} && (2.4)\\
l3 =& \Big[m_{20}(m_{21}m_{03} - m_{12}^2) - m_{11}(m_{30}m_{03} - m_{21}m_{12}) + m_{02}(m_{30}m_{12} - m_{21}^2)\Big]/m_{00}^7 && (2.5)\\
l4 =& \Big[m_{20}^3 m_{03}^2 - 6m_{20}^2 m_{11}m_{12}m_{03} - 6m_{20}^2 m_{02}m_{21}m_{03} + 9m_{20}^2 m_{02}m_{12}^2 + 12m_{20}m_{11}^2 m_{21}m_{03}\\
& + 6m_{20}m_{11}m_{02}m_{30}m_{03} - 18m_{20}m_{11}m_{02}m_{21}m_{12} - 8m_{11}^3 m_{30}m_{03} - 6m_{20}m_{02}^2 m_{30}m_{12}\\
& + 9m_{20}m_{02}^2 m_{21}^2 + 12m_{11}^2 m_{02}m_{30}m_{12} - 6m_{11}m_{02}^2 m_{30}m_{21} + m_{02}^3 m_{30}^2\Big]/m_{00}^{11} && (2.6)
\end{aligned}$$

2.1.4.2 Fourier Descriptors

Fourier descriptors are a very popular and large family of shape descriptors based on the principle of applying the Fourier transform to a one-dimensional function of the shape's boundary. This function is generally referred to as the shape's signature and it can be derived from the shapes boundary in various ways such as: the center of mass distance, tangent angle, complex coordinate position, cumulative angular function, wavelet transform, bending energy and many others (Zahn and Roskies, 1972; Zhang and Lu, 2005; Kunttu *et al.*, 2003; Wang and Shi, 2006; Van Otterloo, 1991; Milan *et al.*, 2007; Zhang and Lu, 2004). Each of the suggested ways for calculating the shape signature would result with a high probability in a different shape signature (representation of the boundary) and therefore the results given by the various Fourier descriptors methods are highly dependent on the way the shape signature is calculated.

A generally considered drawback of Fourier descriptors is its inherent inability to successfully overcome occlusions, however for the purpose of document image verification it is in fact a desirable property since it enables the detection of substantial occlusions. Another potential drawback is the global descriptive nature of Fourier descriptors as they provide general information about the shape signature's frequency contents without any local shape information. In order to address this drawback, methods such as short time Fourier descriptors (Eichmann *et al.*, 1990) and wavelet signature (Kunttu *et al.*, 2003) were suggested that are intended to provide both local and global information about the shape's boundary.

Fourier descriptors have been successfully applied in general shape recognition applications and OCR applications (Taxt *et al.*, 1990; Trier *et al.*, 1996b; Amanatiadis *et al.*, 2009; Zhang and Lu, 2004). They were studied extensively and there are many papers which compare some of the Fourier descriptors methods to other shape recognition methods such as autoregressive models, chain codes, various moments and multiscale curvature scale-space with results suggesting that Fourier descriptors outperform those other shape descriptors (Kauppinen *et al.*, 1995; Mehtre *et al.*, 1997; Amanatiadis *et al.*, 2011).

There are three considerations when using Fourier descriptors as shape descriptors. The first and most important is selecting the method in which the shape's signature function is calculated. The second is the number and selection of coefficients to use as features. The third is the optional normalization of the number of sample points on the shapes boundary. Moreover, there is ambiguity as to how to handle shapes containing holes. Most authors address this issue by considering only the outer perimeter, or equivalently filling all holes prior to extracting the boundary.

Selecting the shape's signature calculation method can dramatically change the general performance of the descriptor. In Taxt *et al.* (1990), three Fourier descriptors have been compared: Kuhl and Giardina' Elliptic Fourier Descriptors (Kuhl and Giardina, 1982), Lin and Hwang's Elliptic Fourier Descriptors (Lin and Hwang, 1987) and Cumulative Angular Function (Zahn and Roskies, 1972), and it has been concluded that the Elliptic Fourier Descriptors (Kuhl and Giardina, 1982) outperform the other two methods. Zhang and Lu (2005) has evaluated six different methods for calculating the shape descriptor: Complex Coordinates (Granlund, 1972), Centroid Distance Function (Zhang and Lu, 2005), Chord Length Signature (Zhang and Lu, 2005), Cumulative Angular Function, Curvature Signature (Krzyzak *et al.*, 1988) and Area Signature (Zhang and Lu, 2005), and has concluded that for general shape classification tasks, the Centroid Distance Function gives the most accurate results. Zhang and Lu (2001) compared Fourier descriptors to short time Fourier descriptors and concluded that Fourier descriptors are superior both in accuracy and in compactness of representation. Wang (2011) lists the Cumulative Angular Function, Centroid Distance Function, Complex Coordinates and Curvature Signature as the most commonly used shape signatures but without specifying application domains. Wang (2011) also provides comparative results between the following methods: Centroid Distance Function, Complex Coordinates, Curvature Signature, Area Signature, Wavelet Signature, Perimeter Area Function (Wang, 2011) and a combination of the Perimeter Area Function and of the Centroid Distance Function. It concludes that the best performance is given by Centroid Distance and Perimeter Area shape signatures.

The second important consideration is the number and selection of coefficients to use as features. In general, many publications such as: Wang (2011), Zhang and Lu (2005), Zhang and Lu (2001), Krzyzak *et al.* (1988), and Amanatiadis *et al.* (2009) reported the typical number of coefficients used as a feature vector to be 30-60. Persoon and Fu (1986) determines that if the boundary contains adjacent linear curve sections the descriptors obtained by the Cumulative Angular Function converge slower to zero relative to descriptors obtained from the Complex Coordinates Signature. This occurs due to the Gibbs phenomenon around points of discontinuities. This suggests that the Cumulative Angular Function descriptors require a higher dimensionality of coefficients, and overall reduces their suitability to describe binary rasterized characters (Trier *et al.*, 1996b; Krzyzak *et al.*, 1988). Zhang and Lu (2005) has evaluated the convergence of the Centroid Distance Function and has concluded that for general shape classification tasks, using only the 10-15 low order coefficients as the feature vector is sufficient. Moreover, according to Milan *et al.* (2007), 10-15 low order coefficients are sufficient for OCR.

Normalization of the number of sample points is an optional step some researchers have utilized. This thesis' approach is the normalization is important as it results in uniformity of the discrete frequencies of the basis functions. Therefore the comparison between the different resulting coefficients of the DFT comprising the feature vector is more meaningful.

Centroid Distance Function Since the Centroid Distance Function has proved to give good results in comparison to other shape signatures, and moreover, holds additional benefits in the simplicity and clarity of its calculation and implementation, its robustness to boundary noise and very low dimensionality, further details are provided below. The discrete center of mass distance function or discrete centroid distance function $S(k)0 \leq k < T$ expresses the distance of the shape's discrete boundary pixels to the shape's region center of mass in coordinates x_c and y_c. The center of mass distance shape signature is calculated as follows:

$$S(k) = \sqrt{\left(B(k)_x - x_c\right)^2 + \left(B(k)_y - y_c\right)^2} \tag{2.7}$$

where

$$x_c = \frac{1}{T}\sum_{i=0}^{T-1} B(i)_x \qquad\qquad y_c = \frac{1}{T}\sum_{i=0}^{T-1} B(i)_y \tag{2.8}$$

T is the number of pixels comprising the shape's boundary, $B(k)_x$ and $B(k)_y$ are respectively the x and y coordinates of the k^{th} boundary pixel. x_c and y_c denote the shape's region center of mass coordinates.

Following this initial calculation is a step to resample $S(k)$ to a uniform length N (denoted by $S_N(k)$) so that the discrete Fourier series coefficients a_n $0 \leq n < N$ represent the magnitude and phase of the same discrete frequencies $2\pi n/N$. Moreover, if N is selected so that it is a power of 2, the Fast Fourier Transform (FFT) algorithm may be used to calculate the discrete Fourier series coefficients more efficiently.

Next the discrete Fourier transform (DFT) is applied to $S_N(k)$ to result in a set of discrete Fourier coefficients $\{a_n \mid 0 \leq n < N\}$ by the following operation:

$$a_n = \frac{1}{N}\sum_{k=0}^{N-1} S_N(k) e^{-\frac{j2\pi nk}{N}} \qquad\qquad n = 0, 1, \ldots, N-1 \tag{2.9}$$

$S(k)$ is invariant to translation by definition since x_c and y_c are translated along with the shape, therefore the discrete Fourier coefficients $\{a_n\}$ are also invariant to translation. Assuming that the origin is located at (x_c, y_c), selecting a different starting pixel along the boundary for tracing, or rotation of the shape around the origin results in a phase shift of $S(k)$, which is reflected in a phase shift of the phasors represented by $\{a_n\}$. Scaling of the original shape results in a linear scaling of $S(k)$ which linearly increases the DC component a_0 (Zhang and Lu, 2001, 2005).

Since $S(k)$ is real-valued, $a_n = a^*_{N-n}$, where a^*_n denotes the complex conjugate of a_n. It follows that a_0 and $a_{(N/2)}$ are both real-valued, and the DFT is entirely defined by the first half of its coefficients $\{a_n \mid 0 \leq n \leq N/2\}$.

To obtain $\{b_k \mid 1 \leq k \leq N/2\}$ a set of invariant features to scale, rotation or selection of starting pixel, and translation, it is defined:

$$b_k = \frac{|a_n|}{|a_0|} \qquad\qquad k = 1, 2, \ldots, N/2 \tag{2.10}$$

Moreover, by using only the low order b_k coefficients, it is possible to reduce the noise which usually resides in the high frequency range and also remove transient changes in the boundary while maintaining the most substantial information of the shape. This has the effect of applying low pass filtering to the boundary, but also results in loss of fine detail of the shape (Zhang and Lu, 2001).

2.1.4.3 2-D Discrete Cosine Transform

The Discrete Cosine Transform (DCT) (Ahmed *et al.*, 1974) is a real valued unitary transform related to the Discrete Fourier Transform (DFT). DCT transforms a real, band limited, N periodic and even signal $x(n)$ from the spatial domain to the frequency domain by expressing the signal as a superposition of discrete sinusoids with varying amplitudes and linearly increasing frequencies. This alternative representation of the signal (an image in the case of shape recognition) in the frequency domain is then used as a mathematical framework for the representation of different shapes in a compact and relatively accurate manner. The DCT is usually applied to a shape's region however it may be applied to a shape's boundary as well in order to extract the shape descriptor.

There are four standard DCT variants which differ by the implied symmetry at the signal boundaries (Zhou and Chen, 2009). The most common variant is the type-II DCT (Ahmed *et al.*, 1974), which implies the signal $x(n)$ is even around $n = -1/2$ and even around $n = N - 1/2$. Herein after the type-II DCT shall be refered to simply as the "DCT" and its definition is given in Equation (2.11) (Ahmed *et al.*, 1974):

$$X_k = w(k) \sum_{n=0}^{N-1} x(n) \cos\Big[\frac{\pi}{N}\Big(n + \frac{1}{2}\Big)k\Big] \qquad\qquad k = 0, 1, \ldots, N-1 \tag{2.11}$$

where:

$$w(x) = \begin{cases} \frac{\sqrt{2}}{N}, x = 0 \\ \frac{2}{N}, x > 0 \end{cases} \tag{2.12}$$

The type-III DCT implies the signal's boundary conditions are even around $n = 0$ and odd around $n = N$. The definition of the type-III DCT is given in Equation (2.13) (Ahmed *et al.*, 1974):

$$x(n) = \frac{1}{\sqrt{2}} X_0 + \sum_{k=1}^{N-1} X_k \cos\Big[\frac{\pi}{N} n \Big(k + \frac{1}{2}\Big)\Big] \qquad n = 0, 1, \ldots, N-1 \tag{2.13}$$

The type-III DCT is the inverse of the type-II DCT and shall be refered to in short as the "inverse DCT" (IDCT).

In Figure 2.1 (taken from Stevenj (2009), used with permission under the GNU Free Documentation License, Version 1.2), the different symmetries implied by the different types of DCT are illustrated.

The DCT-II's even extension on both boundary points of the signal guarantee a smooth periodic extension of the signal which result in better approximation of the signal with less sinusoids. Or in other words, as shown empirically by Ahmed *et al.* (1974) and generalized by Hamidi and Pearl (1976), the DCT converges faster than other linear integral transforms, except for the optimal Karhunen-Loéve transform, and in particular converges faster than the DFT, which results in higher compactness of the data required to approximate the signal.

The DCT can be applied to multi-dimensional signals in a straightforward extension of the one-dimensional definition, a composition of DCTs along each dimension of the signal Milan *et al.* (2007). It follows that a two-dimensional DCT-II of a two-dimensional signal (e.g. an image) is the one-dimensional DCT-II, performed along one of the dimensions, followed by repeating the transform in the second dimension (rows then columns or vice versa).

$$X_{p,q} = w_p(p) w_q q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x(m,n) \cos\Big[\frac{\pi}{M}\Big(m+\frac{1}{2}\Big)p\Big] \cos\Big[\frac{\pi}{N}\Big(n+\frac{1}{2}\Big)q\Big] \begin{cases} p = 0, 1, \ldots, N-1 \\ q = 0, 1, \ldots, N-1 \end{cases} \tag{2.14}$$

where:

$$w_p(x) = \begin{cases} \frac{1}{\sqrt{M}}, x = 0 \\ \sqrt{\frac{2}{M}}, x > 0 \end{cases} \tag{2.15}$$

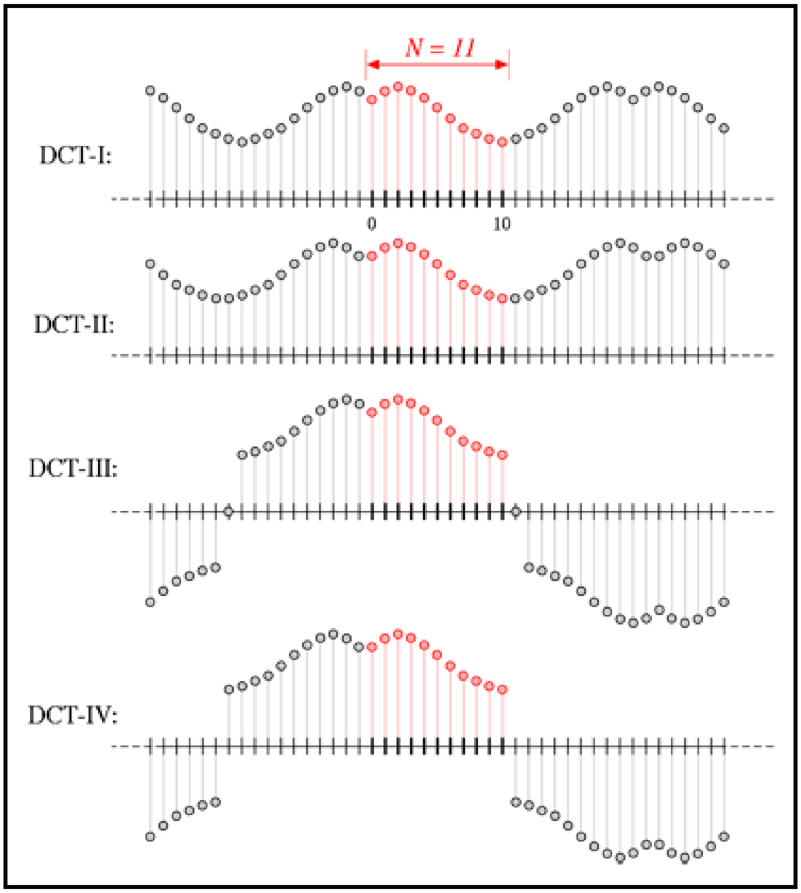
$$w_q(x) = \begin{cases} \frac{1}{\sqrt{N}}, x = 0 \\ \sqrt{\frac{2}{N}}, x > 0 \end{cases} \tag{2.16}$$

Similarly, the multi-dimensional IDCT is calculated by the one-dimensional IDCT, performed along one of the dimensions, followed by repeating the transform in the second dimension. In Figure 2.2 (taken from Devcore (2008), public domain), the DCT basis functions in the two-dimensional case where $M = N = 8$ are illustrated:

A well-known drawback of the DCT as a shape descriptor is its global descriptive nature as it provides general information about the shapes signature's frequency contents without any local shape information. Moreover, the DCT has variance to translation, scale and rotation by definition. However this can be solved by normalization of the shape's translation, size and rotation prior to transforming.

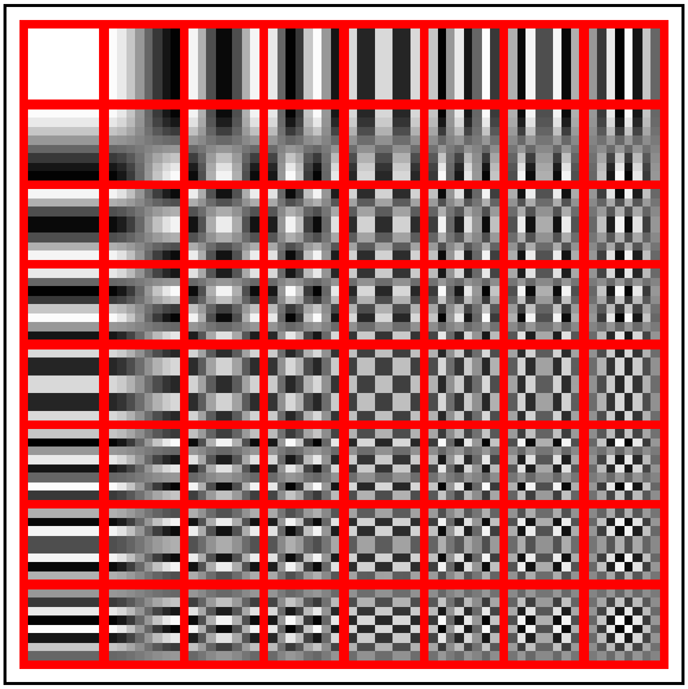
The DCT is widely used for many and varying applications and in particular compression and encoding of video in well known formats such as JPEG and MPEG Khayam (2003) Milan *et al.* (2007), and the modified discrete cosine transform, derived from the DCT-IV, is used in many popular lossy audio compression formats such as AAC, WMA, and MP3. The application of the 2-D DCT in the context of Bangla language OCR has been reported by Abul Hasnat and Khan (2007) where the authors have horizontally segmented each character into several frames, used 2-D DCT transforms to extract features from each frame and then fed the features sequencing to a HMM classifier. Dhanya and Ramakrishnan (2002) has compared features obtained from Geometric Moments, 2-D DCT and the Discrete Wavelet Transform for a bilingual Tamil/Roman OCR system. To generate the 2-D DCT features Dhanya and Ramakrishnan (2002) divided each symbol into four sub-blocks and 2-D DCT is taken on each sub-block. As features only the $H/6 \times W/6$ low frequency coefficients are retained where H and W are the dimensions of the sub-blocks. It was concluded that the best overall results were given by the DCT based features. Similarly, Charan (2006) used 2-D DCT on sub-blocks of characters in the Telugu script.

Figure 2.1: *Illustration of the four different symmetries implied by the different types of the DCT.*



There are a number of important factors which are to be decided when using 2-D DCT transform coefficients as shape descriptors. First, a decision is to be made if the transform is applied to the shape's region or to its boundary only. In the case of transforming the shape's region, it is also necessary to decide whether the transformation is applied to the binarized region or to the intensity or colour channels values of the shape's region. Furthermore, since the DCT is variant to rotation and scaling, the methods of rotation and scale normalization need to be predetermined. And finally, one must decide upon the number and selection of coefficients to use as features.

Figure 2.2: *Illustration of the DCT basis functions in the two-dimensional case where $M = N = 8$.*



2.1.5 Shape Descriptors and OCR Systems

Recognition of character shapes present in images of documents is a process which is generally referred to by the term "Optical Character Recognition", or OCR in short. OCR has been an active field of research for over six decades now and so it is imperative to understand which methods were used in the past and are used presently in OCR systems and what success rates these OCR systems obtain. This information is important in helping make a decision on which approach to take in order to succeed with the thesis' goal.

There are many methods which have been published on 2-D shape descriptors with OCR as the main intended application. A considerable part of the most promising and prominent approaches up until the mid-90s have been surveyed in Trier *et al.* (1996b) and Eikvil (1993). Unfortunately these surveys do not provide experimental results on a common training set, classifier and testing set and so no baseline for comparison between the accuracy of the different methods is available. Since the publication of Trier *et al.* (1996b) and Eikvil (1993) a myriad of new shape features has been suggested in the context of OCR or as general methods for shape discrimination. Unfortunately, as discussed in

Section 2.1.1 using the available results provided by the publications for comparing the suitability, accuracy and efficiency of the different methods for this thesis' particular goal is an impossible task.

Another possible source of information on effective features in the context of OCR is their implementation in contemporary OCR systems. Unfortunately most of the established and well-regarded OCR systems available today such as: ABBYY's FineReader, I.R.I.S.'s ReadIRIS and Nuance's OmniPage are commercial closed propriety systems which do not provide to the public details of the algorithms and methods used in their implementations.

However, there are a small number of state-of-the-art open source systems such as Tesseract (Google, 2012b), OCRopus (which is partially based on the Tesseract OCR engine) (Google, 2012a) and GOCR (Schulenburg, 2012) which provide partial documentation and details on their approach and implementation.

The most notable of these open source systems today is Google's Tesseract. A brief description of Tesseract's background is given in Google (2012b) "An OCR Engine that was developed at HP Labs between 1985 and 1995 and now at Google. The Tesseract OCR engine was one of the top 3 engines in the 1995 UNLV Accuracy test. Between 1995 and 2006 it had little work done on it, but since then it has been improved extensively by Google and is probably one of the most accurate open source OCR engines available".

For shape matching, Tesseract uses a method in which scanned characters are estimated by short segments of polygonal approximations as features (Smith, 2007). However, according to Smith (2007), Tesseract relies heavily on recognition of words to improve segmentation and classification. Tesseract determines the best match for a particular blob or a set of blobs by utilizing a weighted distance of a possible segmentation from matches found in a number of dictionaries.

It is not surprising that Tesseract uses word statistics and dictionaries to improve accuracy. In fact, it is very reasonable to assume that all other commercial OCR systems probably use some sort of linguistic analysis such as dictionaries, n-grams statistics and other methods to improve their accuracy rates. Unfortunately for the purpose of authentication and verification of document images it is not reasonable to take the same approach and to try and infer knowledge about the document image presented for verification since that would defeat the entire purpose of the system. Therefore any use of statistical linguistic analysis such as dictionaries and n-grams statistics must be avoided completely as a method of correcting classification mistakes and improving classification results.

2.2 Classifiers

To be able to recognize symbols in document images the shape descriptors must be processed by a classifier in order to ascribe the input features calculated from a given symbol in the document to a certain character class. It is known in the literature that different combinations of shape features and classifiers will achieve different results. Therefore shape features and classification both play an important part in the recognition of characters. In the next Sections an in-depth discussion of a number of the most prominent classifiers is given.

Before discussing the classifiers a number of definitions are given. Let $C_{set} = \{C_j \mid j = 0, 1, \ldots, M\}$ to be the dependent categorical class variable conditional on the values of a set of random variables $X_{set} = \{X_i \mid i = 0, 1, \ldots, N\}$, (i.e. the feature variables), where in many practical applications $|C| \ll |X|$. And let an observation $\mathbf{X}$ be defined as the random row vector $\mathbf{X} = \{x_1, x_2, \ldots, x_n\}$ where $x_1 \in X_1, x_2 \in X_2, x_3 \in X_3, \ldots, x_n \in X_n$ are referred to as the components of the observation.

2.2.1 Naive Bayes

Naive Bayes (NB) is regarded as one of the most accurate and computationally efficient supervised learning algorithms (Zhang, 2004; Milan *et al.*, 2007). The NB classifier uses a conditional probability model to classify random vectors of an arbitrary dimensionality whether continuous or discrete.

A NB classifier simplistically assumes the random variables X_i are mutually independent, and thus simplifies the model generation and the classification process drastically. In other words, a NB classifier assumes that the value of a particular feature of a class is uncorrelated to the value of any other feature, given the class variable. The assumption of independence reduces the calculation of the multivariate conditional probability to a series of single variable conditional probabilities. Despite the simplistic assumption of mutual independence of the features, NB classifiers are known to give satisfactory results in various applications.

Under the assumption that the features are independently normally distributed, the NB classifier requires only a relatively small number of training observations to estimate the model's parameters, the means and variances of the variables. Estimation of the parameters is usually achieved by maximum likelihood estimates of the probabilities, in other words, the relative frequencies of values in the training data. Class prior distributions $p(C_j)$ can be determined by assuming equiprobable classes so that $p(C_j) = 1/M$ where $j = 0, 1, 2, \ldots, M$, or by estimating the class probabilities from the training data.

By Bayes' theorem the posterior probability for a given observation $\{x_1, x_2, \ldots, x_n\}$ to be classified as C_j is given by:

$$p(C_j|x_1, x_2, \ldots, x_n) = \frac{p(x_1, x_2, \ldots, x_n|C_j)p(C_j)}{p(x_1, x_2, \ldots, x_n)} \tag{2.17}$$

Since $p(x_1, x_2, \ldots, x_n)$, the unconditional probability of the given observation $\{x_1, x_2, \ldots, x_n\}$ remains constant for all C_j the denominator can be ignored. Therefore Equation (2.17) is reduced to:

$$p(C_j \mid x_1, x_2, \ldots, x_n) \propto p(x_1, x_2, \ldots, x_n \mid C_j)p(C_j) \tag{2.18}$$

However, since the variables are assumed to be statistically independent, $p(x_1, x_2, \ldots, x_n \mid C_j)$ can be decomposed into a product of terms as follows:

$$p(x_1, x_2, \ldots, x_n \mid C_j) = \prod_{i=1}^{N} p(x_i \mid C_j) \tag{2.19}$$

and therefore:

$$p(C_j \mid x_1, x_2, \ldots, x_n) = p(C_j) \prod_{i=1}^{N} p(x_i \mid C_j) \tag{2.20}$$

Using Equation (2.20) above and the maximum a posteriori decision rule, a new observation $\{x_1, x_2, \ldots, x_n\}$ is labeled with a class identity C_j that results in the highest posterior probability.

2.2.2 k-Nearest Neighbor

k-Nearest neighbor (k-NN) is regarded as probably the simplest classification algorithm (Milan *et al.*, 2007). k-NN is a lazy learning technique where the model is approximated only locally, and no computation is performed until classification. k-NN classifies random vectors of an arbitrary dimensionality whether numerical or categorical.

k-NN classifies a given new test observation $\{x_1, x_2, \ldots, x_n\}$ by a majority vote of its neighbors, with the observation assigned a class value C_j as the most common class value amongst its k nearest neighbors. The neighbors are chosen during classification from the set of training observations for which the correct class is known. In practice usually Euclidean or Mahalanobis distance is used to calculate the distance from the new test observation to the training data. If any of the random variables X_i are categorical, an appropriate distance function should be defined, such as Hamming distance.

The most common distance function is the Euclidean distance function which is defined by:

$$\begin{aligned} d_{euc}(V, U) &= \sqrt{\sum_{n=1}^{N} (v_n - u_n)^2} && (2.21) \\ &= \sqrt{(V - U)^T (V - U)} && (2.22) \end{aligned}$$

where V and U are column vectors in $\mathbb{R}^n$.

The Euclidean distance of V from the origin is given by $d_{euc}(V, \vec{0}) =\| V \|_2= \sqrt{V^T V}$, which is also known as the Euclidean norm of V. Therefore, all of the points with the same distance c from the origin satisfy $c^2 = V^T V$ which is the equation of an origin centered n-sphere in $\mathbb{R}^n$ with radius c. It follows that all the components $\{v_1, v_2, \ldots, v_n\}$ of the observation V have equal contribution to the Euclidean distance of the point V from the origin.

However, in the case where it is of interest to account for the variance of the random variables $X_{set} = \{X_i \mid i = 0, 1, \ldots, N\}$ when calculating the distance of a random vector (observation) $\mathbf{X}$, the normalized Euclidean distance can be used, which is a special case of the Mahalanobis distance when the covariance matrix S is diagonal:

$$d_{mah}(V,U) = \sqrt{\sum_{n=1}^{N} \left(\frac{v_n - u_n}{s_n^2} \right) 2} \tag{2.23}$$

$$= \sqrt{(V-U)^T S^{-1} (V-U)} \tag{2.24}$$

where V and U are column vectors in $\mathbb{R}^n$, $S = diag(s_1^2, s_2^2, \ldots, s_n^2)$ and $\{s_i\}$ are the standard deviations of $\{X_i\}$.

The parameter k is normally a positive integer, in the special case where $k = 1$, the classified observation is simply assigned to the same class as its nearest neighbor, and the entire process is referred to as the "nearest neighbor classification". The optimal value of k depends upon the data. Selecting higher values of k can reduce the errors introduced by outliers and noise, however has the drawback of possibly making boundaries between classes less distinctive. Estimating the optimal k can be achieved by different heuristics such as cross-validation (Witten and Frank, 1999).

An important drawback of the k-NN algorithm is its inherent disposition to suffer from the curse of dimensionality. As the dimensionality of the observations increases, the probability of having more components pairs to be unequal increases, resulting in arbitrary increase of the distance between any two given points. Therefore, the measure of distance between points loses its discriminative capability as the dimensionality of the data increases (Sima and Dougherty, 2008; Elkan, 2011). However, despite the possibly high dimensional space in which observations are represented as points in, if the different classes' points cluster "nicely" in a subspace of lower dimensionality then the likelihood to obtain good accuracy by classifying using k-NN-based methods increases (Elkan, 2011).

An additional disadvantage of the basic k-NN algorithm and other simple majority voting classification schemes in general is that when $k > 2$ an averaging or voting scheme is required for combining the voting C_j of the multiple training neighbors, which may be difficult to define. In addition, classes with higher prior probability are likely to dominate the labeling decision of the test observation, since they are more likely to be chosen in the set of k nearest neighbors in areas where classes are mixed. Weighing the nearest neighbor's voting in reverse ratio to their distance or using alternative weighted voting schemes such as presented in Coomans and Massart (1982) have been suggested to compensate for the bias caused by training data with non-equiprobable classes.

In this thesis a variant of the basic algorithm is used where the contribution of the k nearest neighbors is weighted in opposite ratio to their distance from the test observation.

The contribution of the class C_j of each of the k nearest points to the final decision is weighted in proportion to the inverse of the distance between the pair C_j/d.

Another drawback of the k-NN algorithm is its high sensitivity to the scaling of data. Components with high variance are more likely to contribute more to the overall distance although they may be less discriminative than other components. This is indeed an important issue and can be reasonably overcome by normalizing the different components. The normalization accuracy increases as the number of observations in the training sets increases.

And finally the last drawback to be discussed is the relatively high computational complexity of making predictions. Consider the case where there are p training observations in $\mathbb{R}^n$. Subsequently classifying one new test observation requires $O(pn)$ calculations. If $p \leq 10$ more efficient data structures such as a kd-tree (Bentley, 1975) can be used to accelerate finding the nearest neighbors. Unfortunately for larger p, and in particular when $p > 20$, there is no known method to be used for improving the running time over the simple linear case (Kibriya and Frank, 2007).

k-NN has a significant and desirable consistency property. For sufficiently large training set size p, and for $k = 1$ the error rate of the k-NN classifier will not exceed twice the Bayes error rate, the optimal theoretical possible error rate (Milan *et al.*, 2007). Moreover it is shown in Devroye *et al.* (1994) that if:

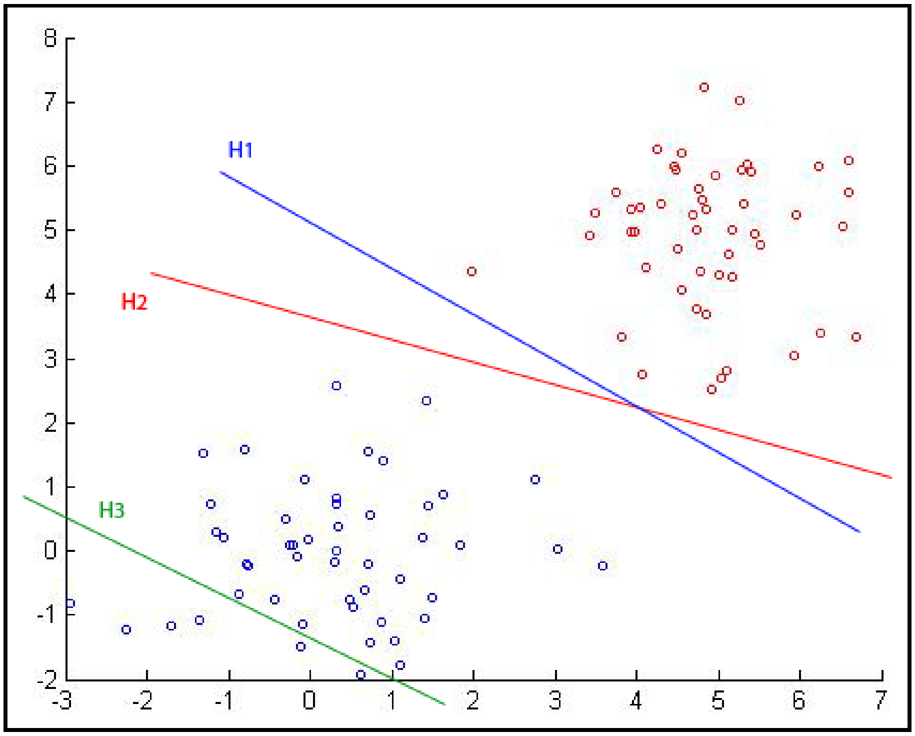
$$k, p \to \infty, \text{ while } \lim_{k\to\infty, p\to\infty} k/p \to 0 \text{ and } \lim_{k\to\infty, p\to\infty} k/\log p \to \infty \tag{2.25}$$

then the k-NN's error rate optimally converges to the Bayes error rate.

2.2.3 Support Vector Machine

Support Vector Machine (SVM) (Cortes and Vapnik, 1995) is a non-probabilistic binary linear classifier. Given a training set of observations, the SVM classifier builds a model which is then used to predict for a given test observation to which of the possible two output classes, C_1 or C_{-1}, it belongs.

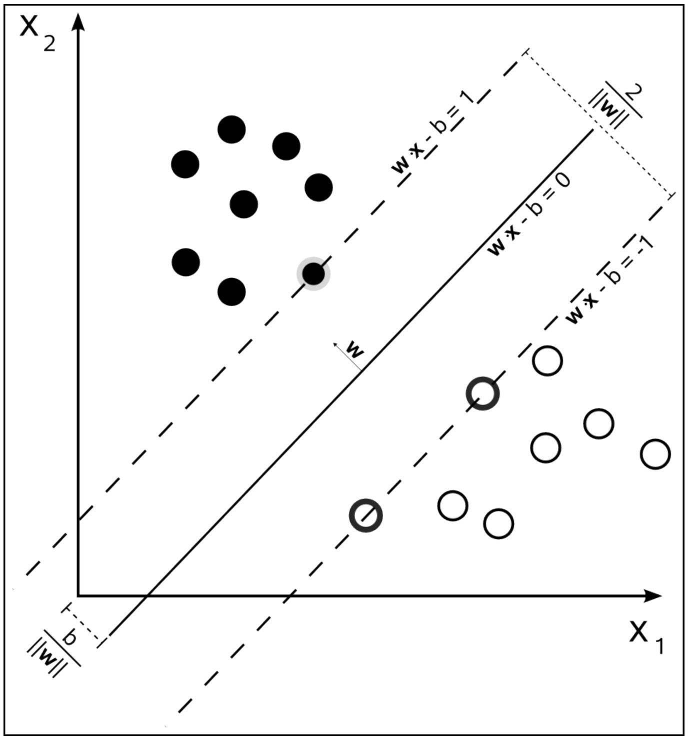
Figure 2.3: *Illustration of separating and non-separating hyperplanes.*



SVM classifier model represents observations as points in $\mathbb{R}^n$. If the classes are linearly separable then usually there are many (n-1)-dimensional hyperplanes that can separate the data so that each of the resulting half spaces contains observations of only the same type of class. In Figure 2.3 (inspired by Cyc (2008), public domain), H1 and H2 are two possible hyperplanes linearly separating the two different classes, while H3 does not linearly separate the two classes:

SVM aims at finding the hyperplane which has the maximum distance to the closest observations in both classes. Provided that such a hyperplane exists, it is referred to as the maximum-margin hyperplane and can be used to define a linear classifier known as the "maximum margin classifier". The observations which are closest to the separating hyperplane in both classes are referred to as the "support vectors". Test observations are then classified to belong to a class based on which side of the hyperplane they map to as illustrated in Figure 2.4 (taken from Buch (2011), public domain).

Figure 2.4: *Illustration of the margins (dashed lines) and the optimal maximum margin hyperplane (solid line). The observations located on the margins are the support vectors.*

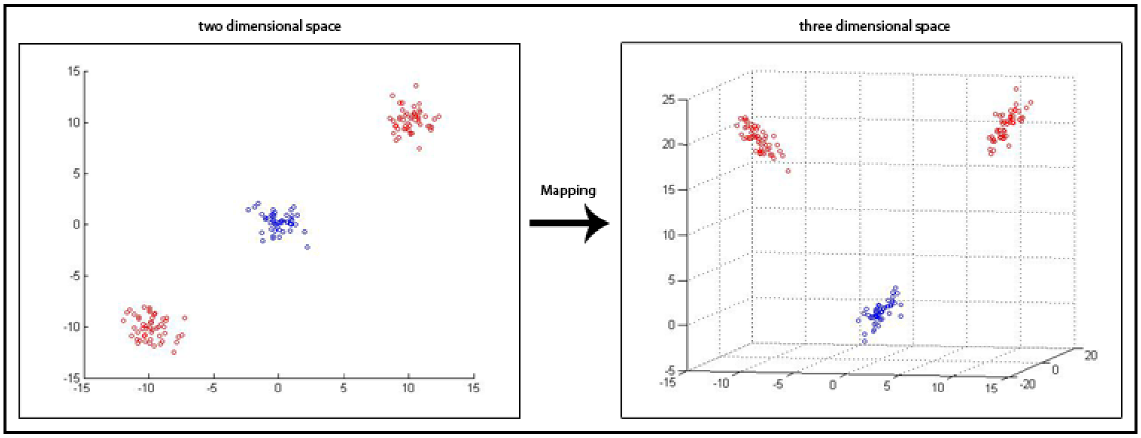


The SVM methodology can be extended to handle data that is not fully linearly separable. Referred to as “soft margin SVM”, this extension of SVM introduces the “slack” or “soft margin” parameter η which allows for the presence of observations on the incorrect side of the margin boundary and enables trade-off between establishing the optimal margin and minimal occurrence of misclassifications.

SVM can also efficiently perform classification of non-linearly separable data. By using a technique known as the “kernel trick”, SVM implicitly maps observations into feature spaces with higher dimensionality. The “kernel trick” becomes useful if the observations, being linearly inseparable in the original lower dimensionality space, are separable in a higher dimensionality feature space given a proper mapping.

For example, data which is linearly inseparable in the two dimensional space can become linearly separable in the three-dimensional space given the mapping $\phi(x_1, x_2) = (x_1, x_2, |x_1| + |x_2|)$ as demonstrated in the Figure 2.5:

Figure 2.5: *Illustration of a mapping transforming data that is linearly inseparable in the two dimensional space to linearly separable data in the three-dimensional space.*



To classify or perform regression using SVM on linearly inseparable data, it is initially required to decide upon a kernel which is expected to map the linearly inseparable data into an alternative feature space where it is possible to linearly separate the mapped data. Unfortunately, there is no method to perform this automatically, therefore finding a suitable kernel requires priori or domain specific knowledge to be applied, and usually evaluation and optimization by empirical methods such as trial and error. Examples of common kernels are the Polynomial kernel, Sigmoid kernel, and the Gaussian Radial Basis kernel (Lin and Lin, 2003; Fletcher, 2009; Milan *et al.*, 2007).

In practice, the kernel functions used for implicitly mapping the inputs are based entirely on the inner products of vector pairs. Therefore, there is no need to explicitly define or calculate the mapping but only explicitly define the new inner product on the higher dimensionally mapped input observations. This has significant practical impact on lowering the computational complexity of classification using kernel methods. However it is important to remember that the resulting accuracy of the SVM classifier is determined by the selected kernel and its parameters, and the soft margin parameter η.

A drawback of the SVM is that it is only directly applicable for binary classification problems. Hence, multi-class classification is only achievable by using procedures that reduce the multi-class problem into a number of binary problems such as the one-versus-all (OVA) (Milan *et al.*, 2007), one-versus-one max-wins voting (Duan and Keerthi, 2005), large margin DAGs (Platt *et al.*, 2000), error correcting output codes (Dietterich and

Bakiri, 1995), pair wise coupling (Hastie and Tibshirani, 1998) and others.

The mathematics for calculating the maximal margin hyperplane are somewhat lengthy therefore they are not brought here, full details are given in Cortes and Vapnik (1995), Burges (1998) and Milan *et al.* (2007).

2.2.4 Random Forest

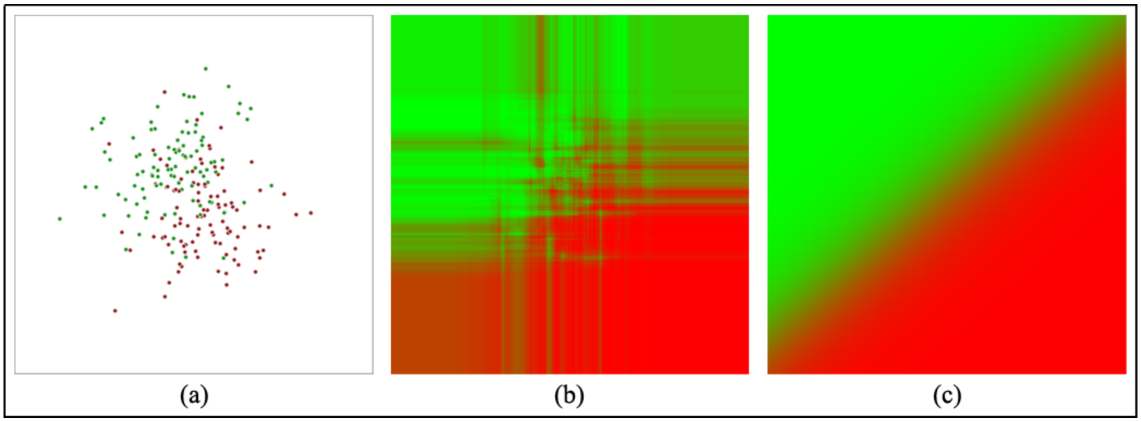
The Random Forest (RF) is an ensemble predictor which yields the mode voting outcome of a large number of randomly built classification trees (Breiman, 2001; Ho, 1995). RF can classify observations of an arbitrary dimensionality whether numerical or categorical. RF is an ensemble of weak learners suitable for solving both classification and regression tasks. In RF, the weak learners are implemented by a decision tree (CART).

Assuming the global training data set is comprised of M observations each of dimensionality N. The P trees in the RF are constructed according to the following guidelines:

1. For each a new tree p_{new}, randomly select with replacement M observations from the training data set to be used as training data for p_{new}.
2. For each node of p_{new}, randomly select n components (feature variables) ($n \ll N$) which are to be the decision variables at that node. The value of n is constant in the course of the forest growing.
3. For each node of p_{new}, determine the optimal split depending on its n components in the training set randomly selected in step 1.
4. The tree is grown as large as possible, no pruning is performed.

To classify using RF, the test observation is pushed down all the trees. Each tree outputs the class label of the training observation corresponding to the leaf the test sample ends up in. the final prediction of the RF ensemble is the unweighted mode vote of all the trees comprising the forest.

Figure 2.6: *Illustration of over-fitting of a Random Forest classifier model.*



RF has been recognized to give superior results both in direct comparisons to other classification methods and in many practical applications (Caruana *et al.*, 2008; Meyer *et al.*, 2003; Verikas *et al.*, 2011; Caruana and Niculescu-Mizil, 2006). In addition, RF has many other advantages such as consistent accuracy for data of varied dimensionality (Caruana *et al.*, 2008); it requires no calibration and in many cases presents optimal out-of-the-box performance (Caruana and Niculescu-Mizil, 2006); it has a single parameter - n, the number of random components used as decision variables at the nodes; RF is capable of splitting the feature space into a number of subspaces each affiliated with a different class, whilst this may be an advantage with certain types of data, in other cases it results in over-fitting to the training data and therefore poor generalization (Segal, 2004). Figure 2.6 (taken from Ronhjones (2012) and Headlessplatter (2011), public domain) illustrates over-fitting of an RF model, Figure 2.6 (a) visualizes a dataset containing 200 observations (100 green and 100 red). Both green and red points were randomly generated having a Gaussian distribution with circular variance of 1 unit. The mean for the green points is (0,1), and the mean for the red points is (1,0). A RF consisting of 50 trees was trained on this data. Figure 2.6 (b) illustrates the feature space representation of an RF model for this dataset where the purity of the colour indicates the portion of the 50 trees that voted in agreement. It is observed that due to over-fitting, each of the classes is represented in the feature space by a set of smaller disjoint subspaces. In contrast, Figure 2.6 (c) illustrates the feature space representation of a logistic regression model for the same dataset.

The error rate of RF depends on the correlation between any two trees in the forest and the accuracy of each tree in the forest (Breiman, 2001). Higher correlation between the trees increases the error rate of the RF. Moreover, increasing the accuracy of the individual trees decreases overall error rate of RF. Unfortunately the parameter n generates a trade-off between the low correlation and the accuracy of the individual trees. Decreasing n decreases both the correlation and the accuracy of the individual trees, and in contrast, increasing n increases both. Fortunately, the optimal n lies within a reasonably sized range and is relatively easy to determine (Breiman and Cutler, 2012).

To help achieve low correlation, randomization is applied in the selection of components used as decision variables at the nodes. To obtain low bias the trees are grown to full depth (no pruning). However Segal (2004) has determined that limiting the depth of the trees decreases over-fitting.

2.2.5 Artificial Neural Network

Artificial Neural Network (ANN) decision models typically involve an interconnected network of basic processing elements (also referred to as neurons). ANNs achieve advanced decision making capabilities by defining an evolving connection patterns between the different neurons and the individual parameters of single neurons. Typically, ANNs take advantage of algorithms that modify the connections (referred to as synaptic weights) in the network to produce a desired signal flow thus creating an adaptive system that effectively changes its organization during a training (learning) phase. ANNs are used for myriad of applications that involve classification (Zhang, 2000), regression analysis, pattern recognition in data and modeling intricate relations between inputs and outputs. A detailed introduction to this subject is given in Haykin (1999).

Training an ANN is performed by selecting one configuration for the model from the set of allowed configurations that minimizes an arbitrary previously defined cost function. For classification purposes ANNs are typically trained by a supervised learning procedure where the cost function reflects the mismatch between the known data to class mapping and the ANNs output therefore implicitly assuming and utilizing prior knowledge about the problem domain.

Being a data driven self-adaptive method, the usefulness of ANN models manifest in the ability of the ANN to infer a function from a given set of inputs without requiring any explicit specification of the underlying function or distribution. This property is useful primarily for the inference of a function where the data or task is so complex that the explicit manual design of such function is unfeasible. Moreover, ANNs are capable of approximating any linear or non-linear function with arbitrary accuracy (Devijver and Kittler, 1982) which makes them effective in modeling real world complex relations.

On the other hand, ANNs suffer from a number of significant and critical disadvantages (Haykin, 1999; Mehta and Kaur, 2013). The first disadvantage is the complexity of designing an optimal ANN for solving most real life learning and classification problems. This is due to the many different ANN heuristics available and for the infinite number of possible models that may be defined. Moreover, ANNs come at a very significant computational complexity when dealing with real-world problem sets. Furthermore, the model parameters that are ultimately selected by the learning algorithm often have no direct physical meaning to the data and the resulting tables are very hard to understand and to interpret by humans. And finally, ANN requires large amounts of training observations to perform effective classification (Haykin, 1999; Mehta and Kaur, 2013).

ANNs have been broadly used in various applications to document analysis and character recognition (Cybenko, 1989). However according to Cybenko (1989), various experiments exhibit critical issues of accuracy and of computational complexity when dealing with real-world problem sets. Given the amount of time available for completion of this thesis, the difficulties in designing successful ANN and the potential performance issues it was decided to not evaluate the recognition accuracy of ANNs as part of this thesis.

2.3 Print-Scan Channel Model and Training Data

Following the discussion of the features and classifiers, the next task is to consider the effects of the printing-and-scanning process on recognition. This printing-and-scanning process will be referred to as the "Print-Scan Channel" and is occurring outside of the domain of the system, being performed usually by a human user with access to printed copies of the original document. Moreover, it is possible that the document has been printed and scanned multiple times before being presented for verification. Hence there are many unknown factors which affect the properties of the Print-Scan Channel and therefore the final channel's output (e.g. the scanned document image which is presented for verification). The printing and scanning process results in local and global degradations and transformations such as thinning, thickening, translation, skew and rotation. These can be thought of as transformed and noisy variations from the ideal image representation of the document and obviously have a major impact on the shape, location and orientation of the characters in the printed and scanned document image. These degradations and transformations are caused by a myriad of physical factors which affect the final image appearance of the print-scan process. Some of the most significant factors are (Baird, 2000; Lins, 2009):

- Paper positioning due to manual placement of papers in scanners.
- Non-uniform illumination induced by defects in design and production.
- Blurring or defocusing caused by the optical components and by mechanical vibrations caused by movement of parts in the printers and scanners, and by uniform or non-uniform movement of the paper in the printer and scanners.
- Finite and discrete spatial and brightness sampling rate, and quantization errors inherent to current electronic equipment.
- Poor print brightness fidelity such as low or high contrast and brightness offset.
- The unique qualitative properties of the different types of printing and scanning equipment such as differences between makes and models.
- The paper's type and quality resulting from usage of different raw materials and production procedures.
- Document wear and tear.
- The number of concurrent prints-and-scans which results in accumulated degradation.

The large number of factors and their possible different varying relative impact means that theoretically no two Print-Scan Channels are identical. As a quick and intuitive illustration to the randomness of the channel consider the randomness of the paper's texture as it is determined by the composition of many cellulose fibers of different sizes and directionality with over 10,000 physical and chemical parameters (Deng and Dodson, 1994). Interestingly, the randomness of paper's texture is used in Fournel *et al.* (2007) as a random number generator for creating a random encryption key. The paper's texture in turn affects the way the ink is immersed and spreads onto the paper, or the way in which toner particles attach to the paper (Zhu *et al.*, 2003), which results in slightly different degradations on each sheet of paper.

To conclude, the Print-Scan Channel is a highly complex stochastic process capable of producing a broad range of degradations depending on many physical and user (human) related factors.

2.3.1 Modeling Degradation for Classifier Training

As discussed in the following Chapters, verification will be performed by using a classifier on the features to recognize characters. Thus it becomes important to be able to provide a reasonable prediction of what the characters may look like after some degrading, and use such degraded symbols as the training data provided to the classifier. Without this, classifier accuracy will suffer since the classifier would be represented by ideal characters, which are already known to not exist in the scanned image. Moreover according to Baird (2000); Zi and Doermann (2004); Cheriet and Moghaddam (2008), there is strong evidence that larger training sets which are well balanced with respect to the number of observations for each of the different classes, and in addition, are also well distributed in respect to covering the range of expected variations in the test data contribute significantly to improving classification accuracy. Therefore there is valid justification to use a training set as large as is useful to achieve optimal accuracy.

Real and Synthetic Training Data The most straightforward method of generating accurate training data to train the classifier is to manually scan a large number of documents, segment and extract the character images and ground truth them. It has also been determined, according to Baird (2000); Rice *et al.* (1992), that for optimal accuracy, the training sets used should be as real and close to the test data as possible. However generating large amounts of training data by such a manual process requires a great deal of labor in producing the documents, printing and scanning them, and labeling tens of thousands of images. In addition, only a limited range of printers and scanners could reasonably be used to create this data. This implies a limited fidelity of the generated training data to real-world situations, in which any arbitrary combination of printers and scanners may be used. Moreover, performing changes and adaptations to the training data would require a repeat of the manual generation process, either entirely or partially. In the long term this would be prohibitively expensive and tend to discourage extending or enhancing the system, such as adding support for new font typefaces.

An alternative to the collection of such “real” training data is the generation of synthetic training data that would be used to train the classifier. Generation of synthetic training data is usually performed by following a model which is aimed at mimicking the degradations caused by the Print-Scan Channel so that the end result would appear to be similar to training data observations collected from real scans of documents. Once the model is implemented, generation of additional training data is done at almost no additional cost. This allows generation of training sets which are orders of magnitude larger than manually generated training sets. An additional benefit is that the labeling and formatting of the output data is done automatically.

Use of Synthetic Only Training Data An important disadvantage of using model-based training data to train the classifier is that the synthetic model is an approximated generalized version of reality and so does not necessarily reflect what real training data obtained from actual scans would produce. Furthermore, it is very hard to accurately model all the factors causing the different transformations, degradation and effects of the Print-Scan Channel. The difficulty arises due to the extremely large number of factors and their variations. Therefore, there is no guarantee of the accuracy and relevance of the generated training data for accurately assisting with the classification of real-world observations. However, given the findings in the literature (Baird, 2000; Nonnemaker and Baird, 2009) that large amounts of reasonably-modeled synthetic data generally outperforms smaller amounts of real data, and the fact that synthetic models for the Print-Scan Channel exist and have been experimentally verified, it was considered more worthwhile to investigate the approach of synthetic model based training data first and determine its validity by testing and measuring the accuracy of the predictions given by different classifiers trained with such synthetic data.

2.3.2 Literature Review of Suggested Degradation Models

Despite the usefulness of document image degradation models and the explicit or implicit use of such models over the last six decades by many OCR systems, only two such models have been suggested prior to the year 2000 (Kanungo *et al.*, 2000, 1993; Baird, 1995) and a few more since (Zi and Doermann, 2004; Moghaddam and Cheriet, 2009; Pezeshk and Tutwiler, 2010). According to Baird (2000), document degradation models are based on two different methodologies. The first approach is physics-based and is based on imitating the physical mechanisms involved in producing the final document image as it is degraded by the Print-Scan Channel. The accuracy and relevance of these models are explained by referring to the physics involved with the print scan process. Example of a physics-based model is given in Baird (1995). The second approach is appearance based. These models aim to be as simple as possible while keeping the appearance of the output generated by the Print-Scan Channel and do not intend to be physically accurate. These models' accuracy and relevancy is explained only by statistical analysis (Baird, 1995), an example of such model is found in Kanungo *et al.* (1993).

Baird (1995) suggested a physical-based document degradation model that by applying sequentially a small number of simple and parameterized per-symbol and per-pixel transformations models the defects and degradations caused by the Print-Scan Channel. This model has the advantage of relying on modeling of physical phenomena to justify its validity and being comprised of straightforward parameters, however it has the drawback of not being readily applicable to an entire image, and therefore it is mainly used to process single character images. Also it is important to stress that the Baird (1995) model suffers from the drawback of being purely symbol-based and therefore not accounting for deformations caused due to joining of characters, or any other degradations which are present at the document level.

Kanungo *et al.* (1993) has suggested a model capable of modeling both global and local degradations such as perspective distortion (global), non-linear illumination (global), non-linear optical point spread function (local) and a probability-based morphological model for local degradation of characters. In contrast to Baird (1995), this model is applicable to an entire image and may be used to model the degradation of entire image or part of. However the local degradations are morphological and probability-based and therefore it is harder to justify its validity.

Zi and Doermann (2004) suggested using custom print drivers to render ground truth TIFF images of documents and applying the model suggested in Kanungo *et al.* (1993) for pixel level degradation. For generating page level degradation, Zi and Doermann (2004) use noise templates extracted from the background of real scanned images and merge them with the ideal image. This model has the same advantages and drawbacks as Kanungo *et al.* (1993), however in addition it also suffers from justifying the use of real scans for noise templates as being applicable to all scenarios as the noise signature depends on the particular equipment, settings, paper type and other factors.

In addition to the aforementioned general degradation models, two more specialized degradation models have been suggested to address specific recognition needs. Moghaddam and Cheriet (2009) addresses the defects generated due to aging and ink seepage for the purpose of restoration of bleed-through corruption in double-sided document images. Pezeshk and Tutwiler (2010) extends the model defined in Baird (1995) to mimic the various artifacts and degradations typical in characters extracted from maps. These models are highly specialized and therefore are not applicable to the research of this thesis.

In the following Section the physical-based degradation model suggested by Baird (2000) is now discussed in more detail.

Degradation Model Suggested In Baird (2000) The Print-Scan Channel modeling suggested in Baird (2000) generates synthetic training data according to the physical-based document degradation model defined in Baird (1995). It models defects and degradations by applying sequentially a small number of simple and parameterized per-symbol and per-pixel transformations. The transformations set applied in the model include (Baird, 1995, 2000):

- ***size***: output size in points (point = 1/72 of an inch).
- ***resn***: output spatial sampling rate (pixels/inch).
- ***skew***: rotation in degrees.
- ***xscl***, ***yscl***: linear scale factors for the horizontal and vertical axes respectively.
- ***xoff***, ***yoff***: linear translation offsets in units of output pixels.
- ***kern***: offsets the horizontal placement of the high resolution image relative to the down sampling grid, in units of output pixel size. This slightly changes the locations of the "pixel sensor" centers and mitigates systematic resampling effects.
- ***jitt***: jitter, the distribution of pixel sensor centers locations from the ideal square grid in units of output pixels. Jitter is modeled by randomly generating for each "pixel sensor" a two-dimensional vector representing its location offset from the ideal square grid.
- ***blur***: defocusing by a circularly symmetric Gaussian point spread function with a standard deviation measured in units of output pixel size. The Gaussian point spread function is centered at the "pixel sensor" center.
- ***ssnv***: symbol sensitivity, intensity offset for the entire symbol in units of intensity.
- ***psnv***: per-pixel sensitivity, the distribution of per pixel additive noise in units of intensity.
- ***thrs***: binarization threshold in units of intensity where 0 represents white and 1 represents black.

For each modeled character, the values of the parameters of the above transformations are either constant or re-chosen pseudo-randomly from a uniformly or normally distributed population with predetermined mean and variance values.

The values of the parameters to all the transformations above, except for *jitt* and *psnv*, are selected for to the entire symbol. *jitt* and *psnv* are per-pixel transformations and their parameters control the randomization of each pixel value. Therefore, the entire symbol transformations can be controlled directly by providing a constant value, however the effects of the per-pixel transformations can only be controlled indirectly, an important issue in parameter estimation.

2.4 Document Verification Systems

Authentication and verification of digital documents is performed by two main approaches: digital signatures and watermarking (Ming *et al.*, 2007) which do not adapt well to printed documents due to the noisy nature of the Print-Scan Channel. The geometric transformations and added noise that are part of the printing and scanning process ultimately do not allow the authentication of printed and scanned documents by pure mathematical comparison to the digital document or its digest as is the case with digital signatures. Consequently, the content perceived in the document image is the only reasonable basis that may be used for automated verification of Print-Scan degraded documents. Thus the following Sections will discuss two different content-based methods that were suggested for verification and authentication of document images contents; an OCR-based hashing approach by Vinicius *et al.* (2007) and a symbol-clustering approach by Ming *et al.* (2007).

2.4.1 OCR Hashing

Vinicius *et al.* (2007) suggested a content-based authentication and verification method in which a binary hash value is calculated as a function of the binary representation of the original text string (the text in the verified document) and a key that depends on the string itself. The binary hash has the same length as the original text string, and it is embedded in the document itself by modulating the characters' luminance.

To verify a printed document copy, OCR is applied to the document image and in addition the average luminance of each character is determined. The hash calculation process is repeated for the output from the OCR and the observed characters luminance, and if any discrepancies are found a document is rejected.

According to the authors, this approach delivers a number of key benefits. First, the verification does not require retrieval of information from an external database as the verification information is completely contained in the document itself. Moreover, it is difficult to forge the authentication process which provides a high level of security and confidence. Finally, the approach can be applied to both the digital and printed documents.

Despite the aforementioned advantages, this method has three significant shortcomings. First it is assumed that no OCR errors occur as any OCR error would change the resulting hash value for the document image. Unfortunately, despite being thoroughly researched for a number of decades, general OCR is typically unable to provide recognition accuracy of over 99% in good quality low noise modern document images (Rice *et al.*, 1995; Rose, 2009). While 99% accuracy may be acceptable for many character recognition applications it is not accurate enough for the purpose of verification. For example, consider that an average page contains roughly 2000 characters, an error rate of 0.5% means 10 errors per page and therefore in practice this method would likely reject almost 100% of documents. Also it is important to note that 99% accuracy is achieved by commercial general OCR systems only in quality scanned low noise document images, with much lower accuracy usually achieved in highly degraded document images.

Moreover, OCR systems are generally black box type systems that do not offer extensive configuration options and do not reveal the fine segmentation and accurate layout details, information which is required for accurate verification.

Finally, since only the global hash value calculated from all of the characters recognized in the document image is compared to the embedded recognized hash, only global rejection is determined without any information suggesting where the changes were detected locally.

2.4.2 Symbol Clustering

Ming *et al.* (2007) suggested a content-based method which considers the shapes of different symbols that may appear in the document and does not rely on a particular character set as most OCR systems often do. In the proposed method, symbols in a binary document are partitioned into classes based on the results of k-means clustering in the feature space. Each class is then assigned a different label. The sequence of symbols in the document is then mapped to produce an ordered sequence of labels and the position of the first occurrence of the symbol label on the document is recorded for each cluster. The ordered sequence of labels and a secret key is then input into a cryptographic hash function to produce a hash value. The hash value and the positions of the first occurrences of character classes are joined to produce a digital signature. The computed signature is suggested to be embedded on a printed hardcopy document in the form of a bar code.

To perform verification of the document image the following steps are performed (assuming symbols from the document image are properly segmented):

- Features are calculated for the symbols segmented from the document image.
- The feature values of the symbols having locations agreeing with the pre-stored positions contained in the digital signature are used as initial mean starting points in the feature space for the k-means algorithm.
- Recognition and labeling of the symbols recognized in the document image is performed based on the k-means clustering in the feature space.
- An ordered label sequence is formed from the labeled symbols in the document image and compute a hash value is computed from the label sequence. The calculation of the hash value from the document image is performed using the same cryptographic hash function and secret key as was used to generate the digital signature of the original document.
- The computed hash value is compared to the expected hash value encoded in the digital signature and the document is rejected in case of any discrepancy.

Using this method, Ming *et al.* (2007) suggest that it is possible to detect any intentional tampering, while at the same time achieve robustness to a moderate amount of Print-Scan Channel induced noise. However, their method has a number of significant drawbacks, including major security vulnerability.

The most prominent shortcoming of the method suggested in Ming *et al.* (2007) is a security flaw which allows an attacker to replace all of the occurrences of an existing character in the document with another non-existing character without being detected. This is possible since the first occurrence for each symbol class is noted as the starting mean for the k-means clustering and therefore as long as all of the following occurrences of the replaced symbol are consistent the document would be successfully verified. Consider the following example, an attacker replaces all occurrences the number '0' in "1,000,000" with the number '9' which results in the number "1,999,999" when verification is performed the features of the leftmost number '9' would be taken as the starting point for the k-means algorithm. Therefore, all of the following '9' would have the same label and the document would be successfully verified.

Moreover, Ming *et al.* (2007) also ignores other known characteristic Print-Scan Channel phenomenon such as the joining of characters (see Section 5.1.4) which requires segmentation (decomposition of a document image into sub images of individual characters) logic to solve. Unfortunately, in commercial OCR systems, segmentation is known to be incorrect often enough to contribute substantially to the error rate of the system. In fact, according to Casey and Lecolinet (1996), even when ideal noise free patterns were input to commercial OCR systems, spacing errors alone averaged a 0.5% error rate.

In addition, although the overall concept suggested by Ming *et al.* (2007) is an interesting bootstrap method of using the scanned characters population itself for the clustering, it is difficult to guarantee that the degraded characters will cluster naturally together. Furthermore if the first occurrence for some symbol class is degraded in a way which does not change its perceived appearance but changes the features sufficiently from the rest of its class it could cause a chain reaction of errors throughout the entire document as these key characters are the starting mean points for the *k*-means clustering during document verification.

And finally, in a similar manner to Vinicius *et al.* (2007), only global rejection is determined without any information suggesting where the changes were detected locally.

2.5 Text Document Character Segmentation

Most OCR systems work relatively well in mildly noisy modern documents images. However, OCR in low-quality or degraded documents images is more challenging due to the high frequency of broken and joined character occurrences (Droettboom, 2003). Joined and broken characters contribute substantially to classification errors and therefore must be dealt with in order to obtain high accuracy (Casey and Lecolinet, 1996). In the following discussion, the process of decomposing a document image into sub images of individual characters shall generally be referred to as “segmentation”.

Ultimately, segmentation can be viewed as a process that decides whether a sub image isolated from the document image is that of a complete single character. As with any decision making process, segmentation can be correct or incorrect. Unfortunately, in commercial OCR systems, it is known to be incorrect often enough to contribute substantially to the error rate of the system (Casey and Lecolinet, 1996).

Being such a critical step of the recognition process, character segmentation methods have been researched extensively over the past decades. Unfortunately, as the large number of suggested methods imply, the segmentation of a document image into separate characters is not a trivial task when no prior information is available about the contents of the document.

Achieving error-free segmentation with no prior information about the document's contents is still an unresolved problem. In cases where the document layout is not constrained and when no prior information about the content of the document such as font types and layout is available, all existing document segmentation techniques rely on statistical analysis of patterns in the document in attempt to segment it into paragraphs, sentences and ultimately characters. Such statistical segmentation methods often tend to fail more in the presence of high levels of noise since the corruption changes the shape of the characters and moreover increases the occurrences of joined characters in the document image as discussed in Section 5.1.

Many simple and advanced document segmentation methods were suggested in the literature for specific applications where some constraints are assumed on the content, and also for the case where no information is known about the document and there are no restrictions on the content. The interested reader is referred to Casey and Lecolinet (1996) as a starting point to explore this large subject.

Given that pure statistical segmentation methods are not utilizing prior document knowledge they are not ideal to use as part of this work, considering that such information is explicitly available. In Section 5.2.5 a new segmentation method is proposed that utilizes such prior information about the content of the verified document in order to perform highly accurate segmentation when the document image content is in agreement with the expected contents. And on the other hand, it is designed that it would fail when the content has been tampered with, thus increasing the chance of detecting tampering in the verified text document.

2.6 Summary

In this Chapter an overview was given of the different methods and concepts related to this thesis. The subjects discussed were shape descriptors, classifiers, document image degradation models, related suggested verification systems and text documents segmentation. In the next Chapter an overview of the proposed framework for robust document image verification is presented.

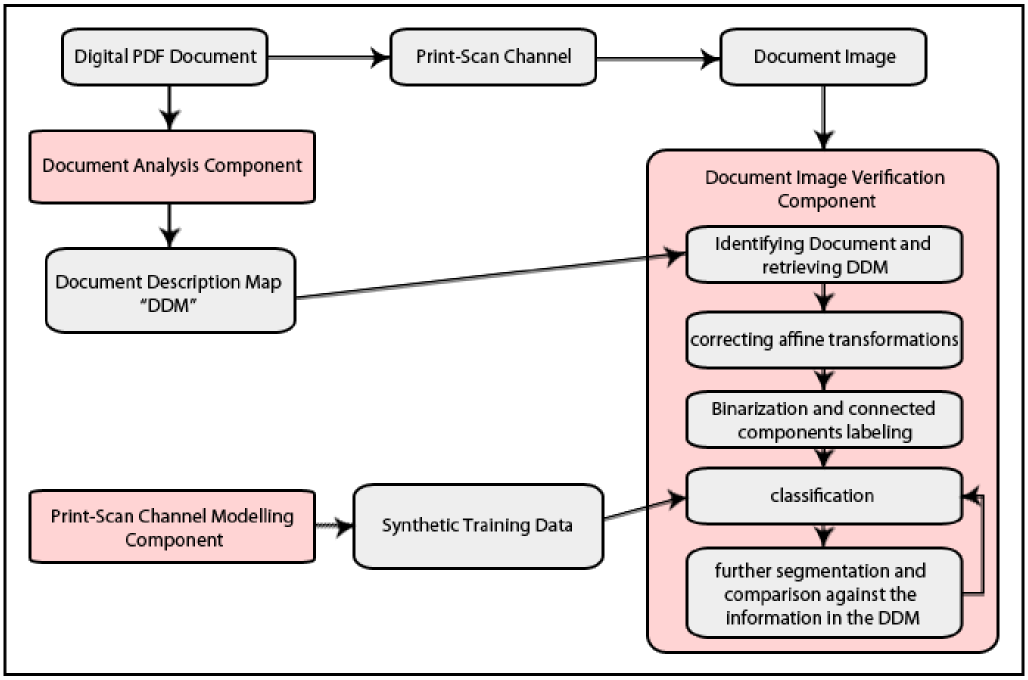
Chapter 3

Document Verification Framework

This Chapter presents an overview of the proposed framework for robust document image verification. The objective of the system is to analyze scanned documents and verify that they have not been tampered with in comparison to the original digital documents. To achieve this objective, the system analyzes the original digital documents stored in the popular PDF format to extract descriptive information which is stored and used in the document image verification stage. This information is refered to by the term “Document Description Map”, or in short by the acronym “DDM”. The proposed approach for robust and accurate verification is to use an adaptive character recognition and segmentation algorithm which would optimize the classifier’s scope by considering only the relevant information for each document or document part. In Section 5.2.4.1 the term “scope of classifier” is defined, along with presentation of the inherent benefits to adapting the scope of the classifier as a method to increase accuracy. Moreover it is demonstrated that by considering the DDM for the purpose of segmentation and for the purpose of narrowing the scope of the classifier (e.g. only relevant fonts) combined with adequate feature extraction, classification and training, much higher accuracy is achieved in comparison to current state-of-the-art OCR systems.

The automated document verification process is comprised of three main components as enumerated below and shown in figure 3.1:

Figure 3.1: *Overview of the main process flow for the document verification system proposed in this thesis.*


Digital PDF Document
Print-Scan Channel
Document Image
Document Analysis Component
Document Image Verification Component
Identifying Document and retrieving DDM
Document Description Map "DDM"
correcting affine transformations
Binarization and connected components labeling
Print-Scan Channel Modelling Component
Synthetic Training Data
classification
further segmentation and comparison against the information in the DDM

1. The Document Analysis Component performs analysis and summarization of the original digital document contents. It extracts the location and font type (typeface family and size) of each character in the digital PDF document. In addition, this thesis simplifies image registration via embedding graphics landmarks in the document's corners (see Section 3.3.2), and their locations are noted by the system and stored as part of the DDM. This information comprise the DDM of that document and is stored for later retrieval and reference in the verification stage where it assists with the estimation and correction of global transforms due to the Print-Scan Channel degradation, classification, segmentation and comparison of the scanned document image to the original digital document.

2. The Print-Scan Channel Modeling Component generates synthetic training data to emulate the effect of the printing and scanning process on characters using the document degradation model defined in Baird (1995). This model-generated synthetic data is used to train a classifier for the particular character sets referenced by the DDM for use in the next stage.

3. The Document Image Verification Component performs the verification of the scanned document images by segmenting document characters and classifying them accord-

ing to the synthetically-trained classifier. The classified characters in the document image are verified against the DDM and any discrepancies are identified.

3.1 Document Analysis Component

The digital document analysis process is comprised of the following principle steps:

1. Read in the document in its digital form (currently supporting PDF documents) and perform extraction of the bounding box and font type for each character in the document.
2. Generating the DDM from the information gathered and storing it for later retrieval by verifying clients.

3.1.1 Extracting Document Content Information

The bounding box coordinates of each character are extracted from the PDF file in PDF coordinates. The origin of the PDF coordinate space is by default in the bottom left corner of the document. The vertical and horizontal axes grow by default in the up and right direction respectively. The units in the PDF space are referred to as "points" and are equal to 1/72 of an inch, traditionally considered to be a standard measure in print applications. More information on the PDF coordinate system is available in the official PDF reference (Adobe Systems, 2009).

However, since the verification step will be performed on a scanned image of the document, it is necessary to be able to convert PDF coordinates to document image pixel coordinates. Fortunately this is a simple computation as long as the image scan dots-per-inch (DPI) is provided:

$$pixel_{xy} = \frac{pdf_{xy}}{72} \times dpi \tag{3.1}$$

where $pixel_{xy}$ is the image coordinates, pdf_{xy} is the pdf coordinates and dpi is the scanned image DPI, typically 300 DPI for good-quality scans.

Extracting and storing each character's bounding box location in PDF coordinates allows the system to support different scan resolutions. In the current system implementation only 300 DPI scanned images are supported, however nothing prevents it from supporting different scan resolutions in the future.

The font type of each character is recorded for the purpose of assisting with segmentation and for adapting the scope of the classifier later in the verification stage. More details on this subject are given in Section 5.2. When performing verification the system queries the DDM for the expected characters' font type and dynamically changes the classifier's decision model in respect to the expected fonts.

3.1.2 Document Identification, DDM Storage and Retrieval

In the current implementation the DDM is stored locally on the verifying machine and its location and document ID is provided manually to the verifying client. However, it is possible to compress, encrypt, digitally sign and embed on the document (e.g. in the form of a 2-D barcode) the DDM as a whole. An alternative approach is to encode in the barcode a key which uniquely identifies the original document, or a URI identifying a network resource from which the DDM may be retrieved. This procedure would provide a secure and convenient means of embedding information on the document which could then be used by the system to identify the base document for comparison. However, such a storage mechanism would use existing well-known methods and so was not chosen to be included in the scope of this research thesis.

3.2 Print-Scan Channel Modeling Component

Once knowledge of the DDM for a particular document is obtained and stored, the goal of the overall system is to later accept a digital scan image of that document and verify that the contents match the original document (as encoded in the DDM). In between these two stages, the document would have been printed and scanned back in, possibly multiple times.

The printing and scanning process results in local and global degradations and transformations such as thinning, thickening, translation, skew and rotation. These can be thought of as transformed and noisy variations from the ideal image representation of the document and obviously have a major impact on the shape, location and orientation of the characters in the printed and scanned document image. These degradations and transformations are caused by a myriad of physical factors which affect the final image outcome of the print-scan process (see Section 2.3).

Since verification will be performed by using a classifier (see Sections 2.2 and 5.2.4) on the features discussed in Section 5.2.4.2 to recognize characters, it thus becomes important to be able to provide a reasonable prediction of what the characters may look like after some degrading, and use such degraded symbols as the training data provided to the classifier. Without this, classifier accuracy will suffer since the classifier would be represented by ideal characters, which are already known do not exist in the scanned image. Moreover according to Baird (2000) Zi and Doermann (2004) Cheriet and Moghaddam (2008), there is strong evidence that larger training sets which are well balanced in respect to the number of observations for each of the different classes, and in addition, are also well distributed in respect to covering the range of expected variations in the test data contribute significantly to improving classification accuracy. Therefore there is valid justification to use a training set as large as is useful to achieve optimal accuracy.

The most straightforward method of generating accurate training data to train the classifier is to manually scan a large number of documents, segment and extract the character images and ground truth them. However as discussed in Section 2.3.1 generating large amounts of training data by such a manual process suffers from many shortcomings. An alternative to the collection of such “real” training data is the generation of synthetic training data that would be used to train the classifier (see Section 2.3.2).

The Print-Scan Channel Modeling Component implemented in the system proposed by this thesis generates synthetic training data according to the document degradation model defined in Baird (1995). This synthetic data is used to train the classifier. In subsequent Chapters it is shown that classifying accuracy improves substantially when training the classifier with a training set that is comprised of synthetic character variations as opposed to training only on the ideal noise free characters. The generation of synthetic training data occurs once for each font type since the artifacts of this component can be stored and recycled for reoccurring use.

3.3 Document Image Verification Component

The process of verifying scanned documents images is comprised of the following principle steps:

1. Identifying the document and retrieving the relevant DDM information which was extracted from the original digital document.
2. Correction of transformations caused by the printing and scanning process.

3. Binarization of the document.

4. Extraction of connected components (connected components labeling).

5. An iterative process of analysis of the connected components which includes classification, further segmentation, and comparison with the information extracted from the digital document (DDM). Based on the analysis performed at this stage a decision is made if the document has been tampered with or not.

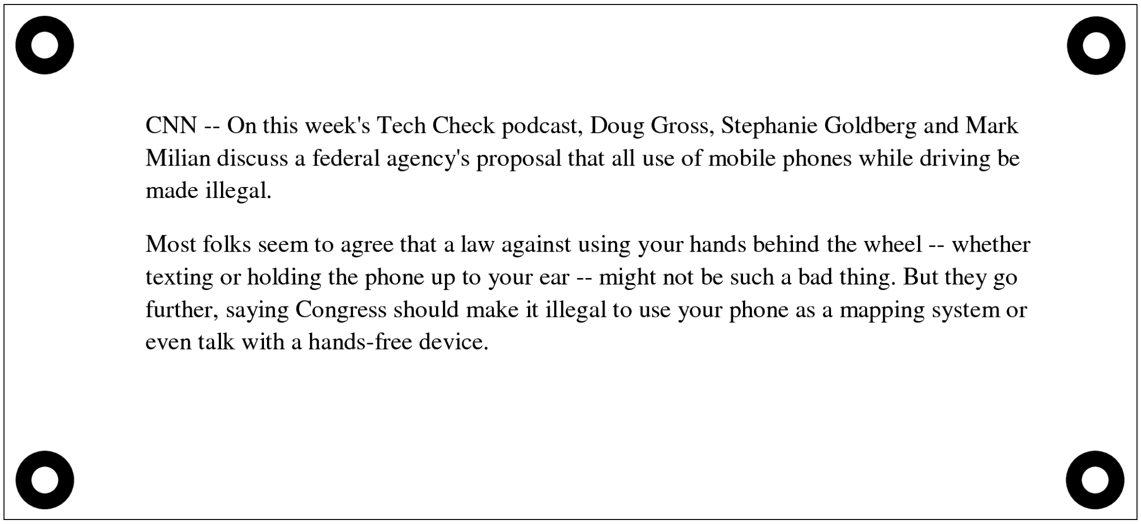
3.3.1 Identifying the Document

In order to verify a presented document image it is required first to determine which document is the relevant basis for comparison. Possible methods that could allow the system to perform this have already been covered in Section 3.1.2, ranging from manually identifying the document to embedded barcodes.

3.3.2 Estimation and Correction of Transformations

Following the determination of a particular document as the ground basis for comparison a correction of the major transformations caused by the printing and scanning process is performed next. This is necessary since the printing and scanning process generates local and global degradations and transformations such as thinning, thickening, translation, skew and rotation. These can be thought of as variations from the ideal image representation of the document and obviously have a major impact on the shape, location and orientation of the characters in the printed and scanned document image. These degradations and transformations are caused by a myriad of physical factors which affect the final image outcome of the Print-Scan Process (as discussed in Section 2.3). In this thesis the approach is to focus on estimating a small but critical subset of the aforementioned degradation factors and considering the others as part of the general noise caused by the Print-Scan Channel. In particular the following linear conformal transformations are estimated and corrected: translation, scale, and rotation.

Figure 3.2: *Illustration of registration landmarks embedded in a text document*



CNN -- On this week's Tech Check podcast, Doug Gross, Stephanie Goldberg and Mark Milian discuss a federal agency's proposal that all use of mobile phones while driving be made illegal.

Most folks seem to agree that a law against using your hands behind the wheel -- whether texting or holding the phone up to your ear -- might not be such a bad thing. But they go further, saying Congress should make it illegal to use your phone as a mapping system or even talk with a hands-free device.

Translation and scale changes the location and bounding boxes of the characters in the document and therefore would lead to mistakes in verifying the location of the characters in the document image against the information provided in the DDM and therefore must be estimated and corrected. Another reason for estimating and correcting any scaling and translation is that the locations of the characters are also used to assist in accurate and robust segmentation of the document image into characters. Rotation must also be estimated and corrected before extracting features from the symbols in the scanned document image since the shape description features used in this framework are not rotation invariant.

Accurate image registration is required by the segmentation algorithm and for verifying the perceived symbols in the document by matching against the original digital document contents. In the current implementation, specifically designated graphic symbols are embedded on the document (typically at the corners of the document, but not necessarily so) to assist with the estimation and correction of the global document transformations scale, rotation and translation. The locations and shape(s) of these embedded designated graphic symbols may be unique for each document and as such they may be stored as part of the DDM. Figure 3.2 shows an example of a text document embedded with the registration assisting symbols.

This approach was chosen since it is simple and accurate and allows the thesis to focus on developing the methods required for content-based verification. Nonetheless, image registration can be performed by a number of alternative well-developed methods such as RANSAC (Fischler and Bolles, 1981) which do not necessarily require the aid of any modifications to the document to perform registration of the scanned document image with the ideal document image, but at the cost of computational complexity and possibly lower accuracy in registration.

To perform registration, the locations of the assisting symbols in the scanned document image are extracted and used in conjunction with their respective pairs in the original digital document to estimate the linear conformal transformation (also referred to as the similarity transform) which can be expressed as:

$$x' = sR_\theta x + t \tag{3.2}$$

where s is an arbitrary scale factor and R_θ is the 2-D rotation matrix by the angle θ:

$$R_\theta = \begin{bmatrix} \cos(\theta) & -\sin(\theta) \\ \sin(\theta) & \cos(\theta) \end{bmatrix} \tag{3.3}$$

To compute the estimated input space coordinates $[u, v]$ for the given transformed space coordinates $[x, y]$ equation (3.4) is solved for sc, ss, tx and ty.

$$[u, v] = [x\ y\ 1] \begin{bmatrix} sc & -ss \\ ss & sc \\ tx & ty \end{bmatrix} \tag{3.4}$$

where:

$$sc = s\cos(\theta) \tag{3.5}$$

$$ss = s\sin(\theta) \tag{3.6}$$

$$t = [tx\ ty] \tag{3.7}$$

At least two pairs of corresponding points are required to solve for the four unknown parameters.

3.3.3 Binarization

After the global transformations are estimated and corrected it is necessary to isolate characters from their surrounding background. To perform this the simple and popular approach of binarization is used. Binarization of the document is performed in order to separate the elements of interest (foreground) from the background and is based on the assumption that brightness intensity levels of the characters will be sufficiently and uniformly different to the background's (usually the paper) brightness intensity (Otsu, 1979). A side effect of binarization is the loss of the foreground pixels' grayscale values. However, since the extracted features which are used for classification are region-based, only the characters' shapes are of interest. Therefore the grey scale values of the characters' pixels can be safely discarded, and this loss of information is in fact a desirable side effect of the binarization process.

3.3.4 Connected Components Analysis

Binarization is closely related to (and is usually used for) extraction of connected components from the document. Connected components labeling (or analysis) is the process of dividing a set of pixels in the images to mutually exclusive subsets based on their connectivity (Milan *et al.*, 2007). It is a reasonable to assume that in the ideal image representation of a document a high likelihood of separation between neighbor characters pixels is maintained. In such a perfect document image, by using connected components analysis it is possible to accurately and efficiently segment the document to form the set of characters of which the document is comprised.

3.3.5 Segmentation and Classification

Unfortunately, as shown and discussed in Chapter 5, the likelihood of having characters successfully separated by connected components labeling decreases quite rapidly with increased noise levels and degradations in the document image. This is caused mainly by some of the degradations discussed in Section 2.3 such as blurring, smearing and thickening which causes neighboring characters to be joined. Moreover, the opposite process in which characters are broken into a few connected components also occurs in higher frequency due to irregularities in brightness and low contrast which are introduced as the noise levels increase. However, despite the limitations and problems discussed above, connected components labeling is used to provide an initial segmentation of the foreground pixels to reasonably accurate sets correlated to individual characters in the document.

Finally, an iterative process of analysis of the extracted connected components which includes classification, further segmentation, and comparison with the information extracted from the digital document (DDM) is performed. Based on the analysis performed at this stage a decision is made if the document has been tampered with or not. A detailed discussion on this procedure is given in Chapter 5.

3.4 Summary

In this Chapter an overview of the proposed framework for robust document image verification is presented. The framework aims to overcome the difficulties arising from the degradations and the noise generated by the Print-Scan Channel. The framework consists of three major subcomponents: *(i)* the Document Analysis Component, *(ii)* the Print-Scan Channel Modeling Component, and *(iii)* the Document Image Verification Component.

The Document Analysis Component extracts descriptive information about the document contents (DDM) by analyzing the digital PDF document. The DDM is stored for later retrieval and reference in the verification stage where it assists with the classification, segmentation and comparison of the document image to the original digital document.

The Print-Scan Channel Modeling Component implements the physical based degradation model suggested by Baird (1995). This model is utilized to synthesize reasonable predictions of what the symbols may look like after some Print-Scan degrading, and use the generated degraded symbols as the training data provided to a classifier. This synthetic training data substantially increases the classifier accuracy as opposed to training only on the ideal noise free characters.

The Document Image Verification Component performs the content-based verification of the presented documents images. The component retrieves the DDM for the verified document; performs a series of pre-processing steps such as estimating and correcting the global linear conformal transformations introduced to the document image, binarization and more; and performs the segmentation, classification and verification of the document's observed content in comparison to the expected state of the document which is stored in the DDM.

Chapter 4

Disjoint Character Verification for Feature and Classifier Selection

In the previous Chapter an overview of the proposed framework for robust document image verification was given. This Chapter presents an evaluation of recognition accuracy of individual characters in noisy Print-Scan degraded documents for different shape descriptors and classifiers combinations.

One of the most important steps in content-based verification of document images is the accurate classification of the characters present in the documents, as they are the fundamental building blocks of text documents. In this Chapter, the recognition accuracy of individual characters in noisy Print-Scan degraded documents is evaluated for different shape descriptors and classifiers combinations. This Chapter aims to evaluate the recognition accuracy for the individual characters given they are properly segmented from the scanned document image. Chapter 5 addresses the additional issues that are typical to scanned text document images due to the typically closely spaced layout of the characters in the documents.

The Chapter commences by defining selection criteria for choosing candidate shape descriptors, and discusses the relevant properties of the selected shape descriptors, which are leading to their selection. It also introduces an image degradation model that is used to generate suitable synthetic Print-Scan degradation modeled training data for improving classification. Finally tests are performed on a uniform test set taken from real scanned documents images to achieve comparative results for the accuracy of the different descriptors-classifiers combinations.

4.1 Selection Criteria of Shape Descriptors for Evaluation

Selecting the most suitable shape descriptor based on information given in the literature for the task of document images verification is a challenging task considering the difficulties in comparing and assessing the suitability of the different shape descriptors such as:

- The large number of suggested shape descriptors in the literature.
- The fact that different descriptors might be more suitable for different applications and particular classes of shapes.
- The large variations in performance which are reported when different classifiers are used with the same shape descriptor.
- Different descriptors have different properties such as invariance to some of the affine transformations such as translation, reflection, rotation and scaling.
- Evaluation of the different descriptors was done independently by each author on different testing and training sets, and for different application domains, therefore, the reported results are not directly comparable.
- The noise levels present in the test data used for evaluating the different descriptors are usually not measured and not reported. Therefore, no conclusion can be made on the effectiveness of the evaluated shape descriptors in noisy document images, which is the primary interest of this thesis.

Therefore in order to accurately assess the suitability of candidate shape descriptors it is required to implement and perform a comparative test of selected shape descriptors over the same training, testing sets and choice of classifiers. However it is impossible to study and test all of the shape descriptors suggested in the literature within the scope of this thesis due to the large number of available methods. Therefore, selection criteria based on the theoretical and practical properties presented in Section 2.1.3 is stated herein after which is used to select suitable candidate methods for evaluation purposes:

- Minimal and simple prerequisites.
- Clear.
- Low coupling to classifiers.
- Computationally fast.
- Compact.
- Variance to major inter-class deformations and occlusions.

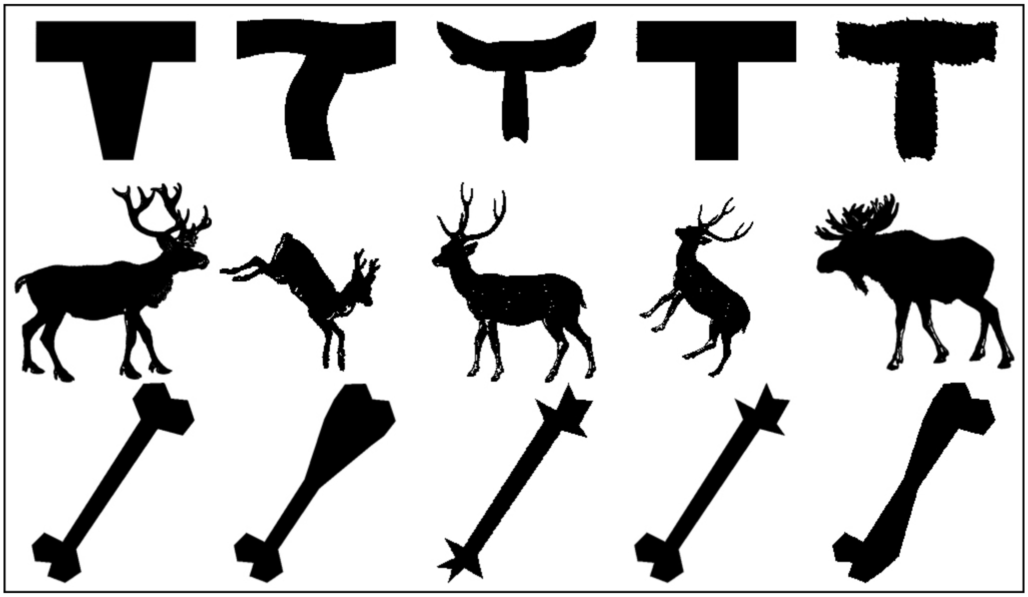
- Invariant and robust to typical Print-Scan Channel noise.
- Thoroughly reviewed in the literature and proven to previously give satisfactory results in OCR applications.

Relevance and Potential Impact of Selection Criteria To illustrate the difference between the goals of contemporary shape descriptors and this thesis's goal of symbol verification, consider properties such as invariance to affine transformations including translation, skew, rotation, reflection and scaling which may have a substantial potential impact on other parts of the system.

For example, consider the following pairs of characters ('d', 'p'), ('d', 'b'), ('6', '9'), ('m', 'w'), ('x', 'X') and ('l', '-'), they are essentially rotated and/or linearly transformed variations of the other (up to a certain error). Consider a scenario where a shape descriptor that is invariant to rotation is used. In order to discriminate between the character pair ('6', '9') additional logic would need to be applied during the classification stage to determine the orientation of the character. Moreover, when deciding on the different classes available as the classifier's output, and therefore in the training data, perhaps it would be better to unite classes which are very similar but rotated such as the pairs ('d', 'p') and ('6', '9') into a single class. The reasons are: *(i)* they are essentially the same (or a very similar) shape, therefore the same shape would receive twice the representation in the training set if both characters are kept as separate classes, and if equal priori representation is assumed for all classes in the training set; *(ii)* reducing the overall number of classes available for the classifier to choose from, could potentially increase classification accuracy and therefore the overall performance of the system. Finally, an additional potential impact is that rotation would not be needed to estimated and corrected, which could potentially improve running time. However, such equivalence between symbols that differ in rotation is not actually desirable for verification purposes.

Another significant factor is that many shape descriptors aim to be able to distinguish between classes in the presence of large deformations such as nonlinear transformations and occlusions in a similar way to human perception. In Figure 4.1 some examples of nonlinear deformations in three shape classes are presented (examples taken from the MPEG-7 Core Experiment CE-Shape-1 part B data set).

Figure 4.1: *Examples of nonlinear deformations in three shape classes.*



Most shape descriptors and in particular the newer state-of-the-art descriptors aim to classify such deformed shapes as equivalent. However, for verification purposes it is desirable that the shape descriptor would be relatively sensitive to such changes in the symbols as the detection of deformations and alterations that effect the meaning of the symbol is required.

Another important criterion for choosing shape descriptors is their computational complexity and its derived running time. For many practical applications a running time not exceeding the order of tens of milliseconds per shape is desirable. However, some of the recently suggested state-of-the-art shape descriptors report running times in the excesses of 30 seconds per query (over the MPEG-7 Core Experiment CE-Shape-1 part B data set (Bai *et al.*, 2010)). Since documents may contain a large number of characters, the time required to classify the characters in documents may become prohibitive for any practical application if such descriptors are used.

4.2 Selected Shape Descriptors for Evaluation

Three methods which satisfy the above criteria were chosen for the evaluation stage.

- Geometric Affine Moment Invariants.
- Centroid Distance Function Fourier Descriptors.
- 2-D DCT II.

For a detailed description of the selected shape descriptors for evaluation refer to Section 2.1.4.

Selected Shape Descriptors Properties The three selected shape descriptors for evaluation share a number of important and desirable properties. All three methods have simple prerequisites; are derived with clear reasoning and small number of parameters, and have a strong relation to the physical properties of the shape's boundary and/or region; computationally inexpensive; can be used with all general purpose classifiers; and are thoroughly reviewed in the literature and used in a multitude of applications.

The uniqueness and difference in the properties for each of the selected features for evaluation are summarized below:

4.2.1 Geometric Affine Moment Invariants (GAMI)

GAMI features are based on a set of a few central moments calculated from the foreground silhouette of a shape. As such, they are computationally efficient in both time and space requirements and have been shown to be useful for recognition under noisy conditions.

- **Compactness** – GAMI are potentially highly compact, as little as 4–7 dimensions may be used (Flusser and Suk, 1993, 1994; Milan *et al.*, 2007). Localization is better achieved by partitioning the character image into blocks and calculating the GAMI features for each of the blocks (Dhanya and Ramakrishnan, 2002) but at the expense of higher dimensionality.
- **Computational Complexity** – The computational complexity of GAMI descriptors is linear in respect to the number of pixels in the shapes region (Milan *et al.*, 2007) positioning them as one of the fastest shape descriptors to calculate.
- **Proven to Give Satisfactory Results in OCR Applications** – GAMI were studied extensively and were proven to be useful in various shape recognition tasks and OCR applications in particular. Refer to Section 2.1.4.1 for more details.

- **Affine Transformation Invariance** – GAMI are specifically designed to be invariant to affine transformations.

- **Robustness to Noise** – Low order moments are considered relatively robust to noise. As the order of the coefficients grows each pixel contributes substantially to the sum making the higher order moments more sensitive to pixel noise (Zhang and Lu, 2004).

4.2.2 Centroid Distance Function Fourier Descriptors (CDFD)

CDFD features are based on the principle of applying the Fourier transform to a one-dimensional function of the shape's boundary and have proved to give good results in comparison to other shape signatures.

- **Compactness** – CDFD are considered to be relatively compact features. Convergence rate of the coefficients has been studied in various publications (Zhang and Lu, 2005; Amanatiadis *et al.*, 2009; Zahn and Roskies, 1972; Trier *et al.*, 1996a) which recommended the number of coefficients used as a feature vector to range from 10 to 32. According to Milan *et al.* (2007); Mori *et al.* (1995), between 10 and 15 low order coefficients are sufficient for OCR.

- **Computational Complexity** – The computational complexity of CDFD descriptors is $O(N \log(N))$ utilizing the Fast Fourier transform (FFT) algorithm where N is the length of the shape signature (Milan *et al.*, 2007).

- **Proven to Give Satisfactory Results in OCR Applications** – CDFD has been used successfully in many general shape recognition applications and OCR applications in particular. They were compared to other shape recognition methods such as autoregressive models, chain codes, various moments and multi-scale curvature scalespace with results suggesting that CDFD outperform these other shape descriptors. Refer to Section 2.1.4.2 for more details.

- **Affine Transformation Invariance** – In general CDFD are invariant to translation, but are sensitive to other general affine transformations and to selection of the boundary's tracing starting pixel. However, CDFD can be made invariant to rotation and to the selection of the boundary's starting pixel by discarding the phase information. Invariance to scale is also achieved by normalizing the norm of the coefficients in respect to the DC coefficient (Zhang and Lu, 2005).

- **Robustness to Noise** – CDFD is considered to be generally robust to noise (Amanatiadis *et al.*, 2009).

4.2.3 2-D DCT II (DCT)

DCT features are based on the principle of applying the 2-D DCT transform to the entire region of the shape. This representation of the shape in the frequency domain is then used for the representation of various shapes in a compact and relatively accurate form.

- **Compactness** – The DCT exhibits excellent convergence rate of the coefficients which makes it more compact in comparison with other integral transforms (Ahmed *et al.*, 1974; Khayam, 2003). However for shape description applications, to the best of this author's knowledge, information is not available on the number of coefficients required for successful recognition.
- **Computational Complexity** – A fast algorithm for calculating the 2-D DCT II in $O(N \log(N))$ operations is available utilizing the FFT algorithm where N is the number of pixels in the shape image (Milan *et al.*, 2007).
- **Proven to Give Satisfactory Results in OCR Applications** – The application of 2-D DCT II for character recognition purposes was proposed in a small number of relatively recent non-commercial frameworks for recognition of Bangla, Tamil/Roman and Telugu scripts. These proposed frameworks however use different methods to extract the descriptors from the characters in comparison to the method proposed by this thesis, and do not specify if they were evaluated on noisy Print-Scan degraded documents images. Refer to Section 2.1.4.2 for more details.
- **Affine Transformation Invariance** – The 2-D DCT II is not affine transformations invariant, however, through normalization of the shape's size, translation and rotation prior to applying the 2-D DCT transform, the resulting features are invariant to linear conformal transforms such as scale, translation and rotation.
- **Robustness to Noise** – The 2-D DCT II can be made relatively resilient to the effects of noise by discarding the higher order coefficients, where the signal-to-noise ratio is typically smaller (Milan *et al.*, 2007).

4.3 Print-Scan Channel Model and Training Data

Given the selected features, the next task is to consider the effects of the printing-and-scanning process on recognition. This printing-and-scanning process will be referred to as the "Print-Scan Channel" and is occurring outside of the domain of the system, being performed usually by a human user with access to printed copies of the original document. Moreover, it is possible that the document has been printed and scanned multiple times before being presented for verification. As discussed in Section 2.3 there are many unknown factors which affect the properties of the Print-Scan Channel and therefore the final channel's output (e.g. the scanned document image which is presented for verification).

4.3.1 Modeling Degradation for Classifier Training

Since verification will be performed by using a classifier on the features discussed in Section 4.4.1.2 to recognize characters, it thus becomes important to be able to provide a reasonable prediction of what the characters may look like after some degrading, and use such degraded symbols as the training data provided to the classifier. Without this, classifier accuracy will suffer since the classifier would be trained on ideal characters, which are already known do not exist in the scanned image. Moreover according to Baird (2000); Zi and Doermann (2004); Cheriet and Moghaddam (2008), there is strong evidence that larger training sets which are well balanced in respect to the number of observations for each of the different classes, and in addition, are also well distributed in respect to covering the range of expected variations in the test data contribute significantly to improving classification accuracy. Therefore there is valid justification to use a training set as large as is useful to achieve optimal accuracy.

Real and Synthetic Training Data The most straightforward method of generating accurate training data to train the classifier is to manually scan a large number of documents, segment and extract the character images and ground truth them. It has also been determined, according to Baird (2000); Rice *et al.* (1992), that for optimal accuracy, the training sets used should be as real and close to the test data as possible. However as discussed in Section 2.3.1 generating large amounts of training data by such a manual process suffers from many shortcomings.

An alternative to the collection of such "real" training data is the generation of synthetic training data that would be used to train the classifier. Generation of synthetic training data is usually performed by following a model which is aimed at mimicking the degradations caused by the Print-Scan Channel so that the end result would appear to be similar to training data observations collected from real scans of documents.

Given the findings in the literature (Baird, 2000; Nonnemaker and Baird, 2009) that large amounts of reasonably-modeled synthetic data generally outperforms smaller amounts of real data, and the fact that synthetic models for the Print-Scan Channel exist and have been experimentally verified, it was considered more worthwhile to investigate the approach of synthetic model based training data first and determine its validity by testing and measuring the accuracy of the predictions given by different classifiers trained with such synthetic data. For a detailed discussion of this topic refer to Section 2.3.

4.3.2 Degradation Model Selected

As discussed in Section 2.3.2 document degradation models are based on two different methodologies. The first approach is physics-based and is based on imitating the physical mechanisms involved in producing the final document image as it is degraded by the Print-Scan Channel. The second approach is appearance based. These models aim to be as simple as possible while keeping the appearance of the output generated by the Print-Scan Channel and don't intend to be physically accurate.

The Print-Scan Channel modeling selected in this thesis generates synthetic training data according to the physical-based document degradation model defined in Baird (1995) (see Section 2.3.2). It models defects and degradations by applying sequentially a small number of simple and parameterized per-symbol and per-pixel transformations.

Model Parameters Since the synthetically degraded characters output by the model are a population of observations having statistical attributes defined by the model's parameter values, it is expected that the model's generated output would cover variations across a range of degradation levels. However, if simulating populations with higher levels of degradation are of interest, two main approaches can be taken. The first is changing the model's parameter values such as offsetting the mean and/or increasing variance for the different transformations where applicable. Alternatively, the model could be applied multiple times repetitively to its own output with the same parameter configuration to simulate the accumulating degradation of repeating printing and scanning. The former approach was used in this thesis to generate sufficiently degraded training data.

Baird (1995) recommended parameters values for the transformations defined in the model excluding size and resn which are to be configured according to the desired output size and resolution. Note that Baird (1995) specifies all normal distributions' parameters in terms of mean and standard error. In this thesis' implementation the standard error was replaced with the standard deviation so it would not decrease as the sample population grows. Following is a review of the recommended parameters values suggested by Baird (1995), along with the parameters values used in this thesis for generating the synthetic data:

- ***size***: equivalent to the modeled font size (i.e. 12 for a 12 point font).
- ***resn***: constant at 300, to model 300 pixels per inch (DPI) scans.
- ***skew (rotation)***: Baird (1995) recommends a normal distribution with zero mean and 1.4 standard error. These are deduced from measurements taken from over 1000 scanned pages. The same distribution was used in this thesis. Baird (1995) does not specify the reference point for rotation. The point selected by this thesis as the reference point for rotation was the image center point.
- ***xscl***: Baird (1995) recommends a uniform distribution over the range $[0.85, 1.15]$. The same distribution was used in this thesis.
- ***yscl***: Baird (1995) recommends a normal distribution with zero mean and 0.02 standard error. The same distribution was used in this thesis.
- ***xoff, yoff***: Are fixed at zero since the translations of the characters are not of interest for this thesis at the classification training stage.
- ***jitt***: for both axis Baird (1995) recommends a normal distribution with zero mean and 0.7 standard error. The same distribution was used in this thesis.
- ***kerx, kery***: Baird (1995) defined a single parameter kern to be applied only in the horizontal direction. However in this thesis' implementation kerning was applied to both axes therefore the single parameter kern was expanded into the two independent parameters kerx and kery. Baird (1995) recommends kern to be modeled with a uniform distribution over the range $[-0.5, 0.5]$. The same distribution was used in this thesis but applied for both axes independently.
- ***blur***: Baird (1995) recommends a normal distribution with 0.7 mean and 0.3 standard error. The same distribution was used in this thesis.
- ***ssnv***: Baird (1995) recommends a uniform distribution over the range $[0, 1]$. The same distribution was used in this thesis.

- ***psnv***: Baird (1995) recommends a normal distribution with 0.125 mean and 0.04 standard error. The same distribution was used in this thesis.
- ***thrs***: Baird (1995) recommends a normal distribution with 0.25 mean and 0.04 standard error. This thesis' approach was different. For each symbol the optimal threshold was calculated using Otsu's method and then offset by a pseudorandom value taken from a population with normal distribution with zero mean and 0.05 standard deviation. This thesis' belief is that this approach models variation in erroneous selection of the thresholds values better than the method proposed in Baird (1995) since selection of threshold value(s) in document images is usually performed by an attempt to perform an optimal selection, which in practice results in some degree of error from the optimal choice.

In this thesis, the inputs to the model are ideal high resolution (3,000 DPI) grayscale images representing the ideal character in high resolution spatial domain. For each input image, modeling the Print-Scan Channel degradations was achieved by applying the transformations in the order illustrated in Figure 4.2.

The implementation of the model was used to generate training data for the set of alphanumeric characters and four punctuation symbols: apostrophe, comma, period and dash for the Times New Roman 12 point font. Figure 4.3 presents the ideal character set for which the synthetically degraded variations were created. In this Chapter, as discussed in Section 4.4, the evaluation is performed on single characters in the times new Roman 12 point font. In Chapter 5 verification of multi-font documents is addressed and the required training data is generated for these additional font types.

Figure 4.2: *Illustration of the order of applied transformations used for the generation of synthetic training data according to the document degradation model defined in Baird (1995).*

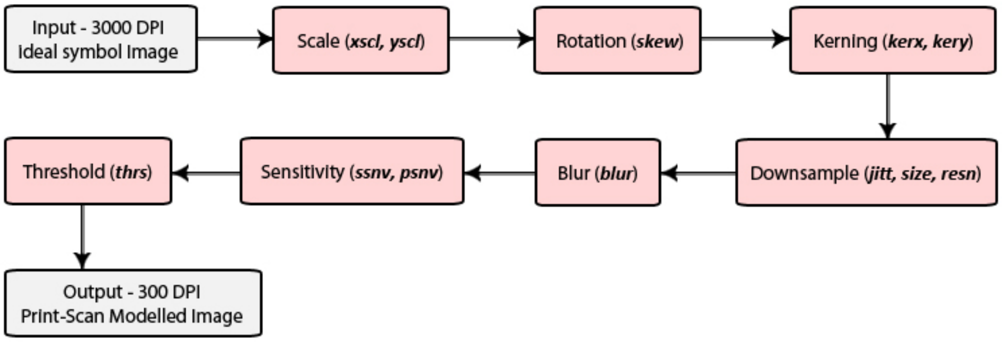
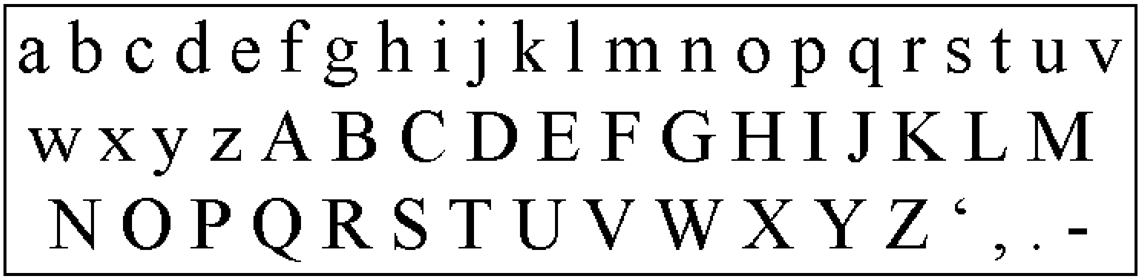


Figure 4.3: *The ideal character set of the Times New Roman 12 point font for which synthetic training data was generated using the synthetic degradation model defined in Baird (1995).*



a b c d e f g h i j k l m n o p q r s t u v
w x y z A B C D E F G H I J K L M
N O P Q R S T U V W X Y Z ‘ , . -

4.4 Evaluation

The aim of this evaluation is to measure the accuracy of the different combinations of shape descriptors and classifiers obtained when classifying real Print-Scan degraded characters. The three shape descriptors chosen for evaluation: Geometric Affine Moment Invariants (GAMI), Centroid Distance Function Fourier Descriptors (CDFD) and Discrete Cosine Transform type II (DCT) are discussed in Section 2.1.4. The four classifiers chosen for evaluation: k-Nearest Neighbor (k-NN), Random Forest (RF), Support Vector Machines (SVM) and Naive Bayes (NB) represent a range of popular, successful and fundamentally different approaches to classification, and are discussed in detail in Section 2.2. For evaluation, each of the shape descriptors is coupled with each of the classifiers resulting in a total of twelve feature-classifier combinations.

4.4.1 Method

In the following Sections an overview of the test data, features extraction and the classifiers that were used to evaluate the accuracy of the different combinations of shape descriptors and classifiers is presented.

4.4.1.1 Test Data

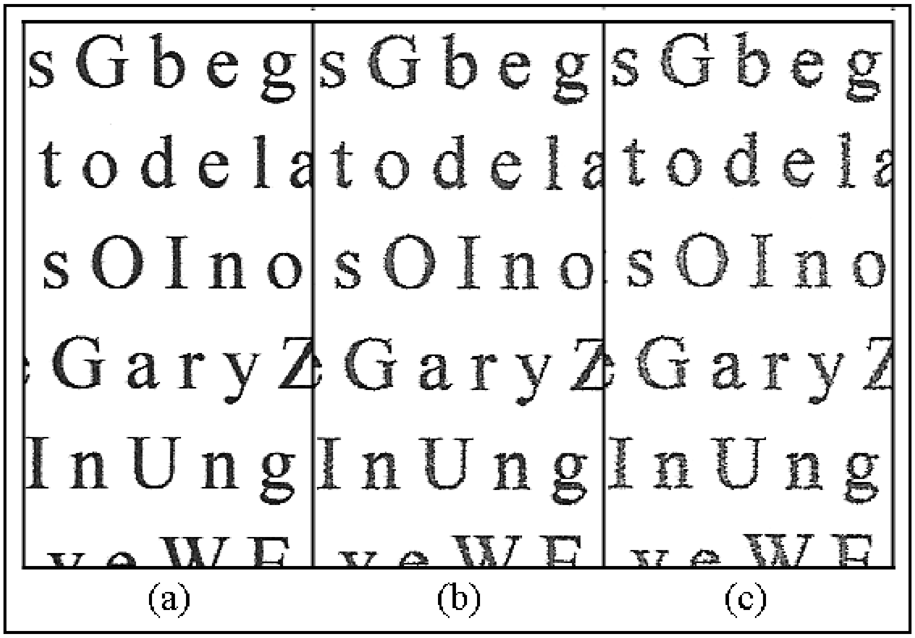
In order to directly compare these methods in a meaningful manner, they are compared when applying the same pre-processing operations, and on a uniform set of training and testing data.

For all of the following experiments, unless otherwise specifically noted, the test data was comprised of 3,193 characters extracted from self-produced PDF text documents. The characters are all instances of Times New Roman 12 point font and contain the entire English language alphanumeric character set except for lower case ‘i’ and ‘j’, a total of 60 characters (classes). The characters ‘i’ and ‘j’ are comprised of two shapes, and therefore require applying additional higher-level logic to extract them from the document images and to recognize them, an issue which will be addressed specifically in Chapter 5. Consequently, it was decided to omit ‘i’ and ‘j’ and concentrate on evaluation of shapes in the “purest” sense which are comprised of a single part.

The test documents’ contents were based on short paragraphs copied from various news articles, from which the characters ‘i’ and ‘j’ were removed for the reason explained above, and onto which the digits ‘0’-‘9’ and uppercase ‘A’–‘Z’ characters were artificially interleaved into the document in equal numbers to uniformly increase their frequency of appearance in the text. As a result, the occurrence frequency of the different test character classes is not uniform, but more inclined towards real world distribution. However, it is guaranteed to give sufficient representation to all character classes excluding ‘i’ and ‘j’.

Given that this thesis aims to handle verification under conditions where the document image is significantly degraded from printing and scanning noise, the test documents were printed and scanned repetitively three times in order to simulate three increasing levels of Print-Scan degradation. The documents were printed and scanned using a HP 7500A model standard inkjet consumer all in one printer and scanner on standard general use office paper. The printing was done using the default standard settings and the scanning was done using 300 DPI grayscale settings. The characters images used for testing were

Figure 4.4: *The cumulative degradation generated after each additional print-scan cycle is shown in increasing order from (a) to (c).*

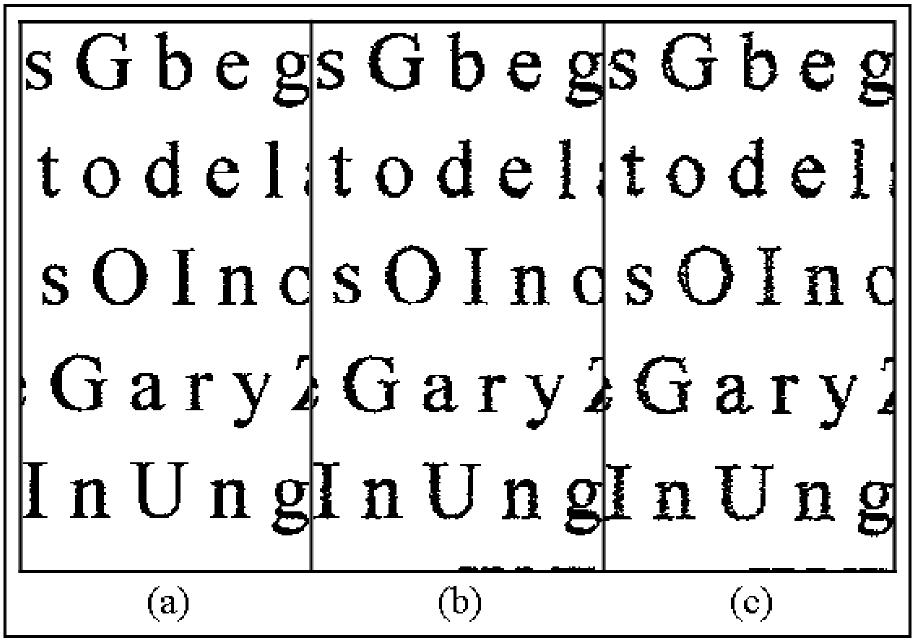


extracted only from the last (most degraded) repetitive print-scan image (after three accumulated repetitive print-scan applications).

The test characters were adequately spaced to guarantee that the cumulative degradation caused by the repeating print and scan process does not result in joined characters which would complicate the segmentation process. Moreover, the test data was inspected to remove all broken characters as this also requires higher-level segmentation methods to solve and would discriminate against Fourier descriptors which are more sensitive to this type of defect. Segmentation of document images for correcting joined and broken characters defects is addressed specifically in Chapter 5.

Prior to extracting the characters from the documents and generating the shape features, the document scan images are transformed to compensate for the possible linear conformal transformations scale, rotation and translation which are introduced by the recurring print-scan process. To simplify the estimation and correction of the transformations, specifically designated graphic symbols were embedded at the corners of the document. To separate foreground pixels from background pixels global binarization is used where the global

Figure 4.5: *The binarized documents excerpts illustrating the effects of the cumulative degradation generated after each additional print-scan cycle on binarization is shown in increasing order from (a) to (c).*



threshold was determined using Otsu's method (Otsu, 1979). Finally, connected components analysis was performed to extract the characters from the documents. It is important to stress that no noise filtering or any morphological operations were performed at any stage prior to or following binarization and connected components analysis.

It is noticeable in Figure 4.4 that with each repeated printing and scanning cycle, more irregularities are formed in the characters. These irregularities are present both in the shape's region and at the shape's boundary where they are prominent in particular. Also it is quite noticeable that the contrast between the characters and the paper (foreground and background pixel populations) is diminished with each repeating print-scan cycle. This is expressed in increased overlapping between the intensity levels of the foreground's and background's pixels and increases the separation error achieved by binarization methods, and results in severe irregularities of the binarized characters boundaries as can be observed in Figure 4.5. For clarity, it is noted that binarization occurs once after all repeating cycles of printing and scanning are completed.

4.4.1.2 Feature Extraction

For all of the performed experiments, unless otherwise specifically noted, both training and testing observations were calculated by the methods described in the following Sections.

GAMI Features For generating the GAMI descriptors each input character is scaled using bilinear interpolation so that the larger of its height or width is 32 pixels, and if required the other dimension is padded with zeros symmetrically resulting in a 32×32 pixel image without distorting the original proportions of the character. Following that, the character is divided into 16 disjoint 8×8 pixels blocks. For each block the $I1 - I4$ invariants as defined by Equations (2.3)–(2.6) are calculated from the entire block region, and grouped to form a 64 dimensional feature vector. This sub-blocking approach is similar to the approach taken by Dhanya and Ramakrishnan (2002) which is shown to increase localization and accuracy.

CDFD Features For generating the CDFD descriptors the outer boundary pixels of the character are extracted ignoring any holes in the character as illustrated by Figure 4.6.

The Euclidean distance for each boundary pixel from the character's centroid is calculated to generate the Centroid Distance Function shape signature as described in Section 2.1.4.2. To maintain uniform discrete frequencies as a sound basis for comparison of the different characters when calculating the Discrete Fourier Transform, the shape signature is normalized to a fixed length of 128 samples by linear interpolation to facilitate the Fast Fourier Transform (FFT) algorithm. An illustration of the extracted and normalized Centroid Distance Function shape signature for the character 'a' from Figure 4.6 is given in Figure 4.7.

The 16 low order invariant descriptors $\{b_k \mid 1 \leq k \leq 16\}$ are then computed as described in Section 2.1.4.2 and grouped to form a 16 dimensional feature vector. The number 16 was chosen based on recommendations found in Milan *et al.* (2007); Mori *et al.* (1995).

Figure 4.6: *Illustration of the boundary extracted from characters containing holes for calculation of the CDFD features.*

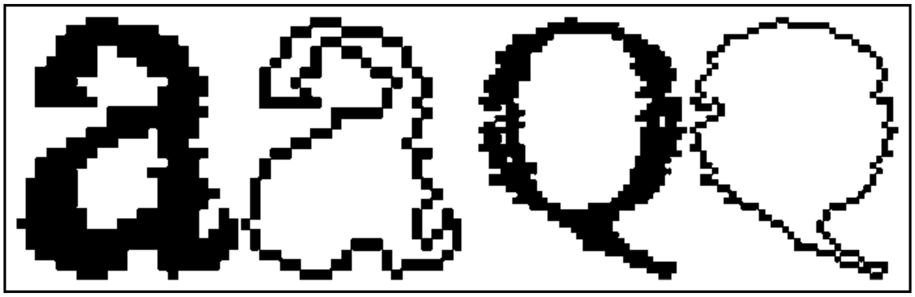
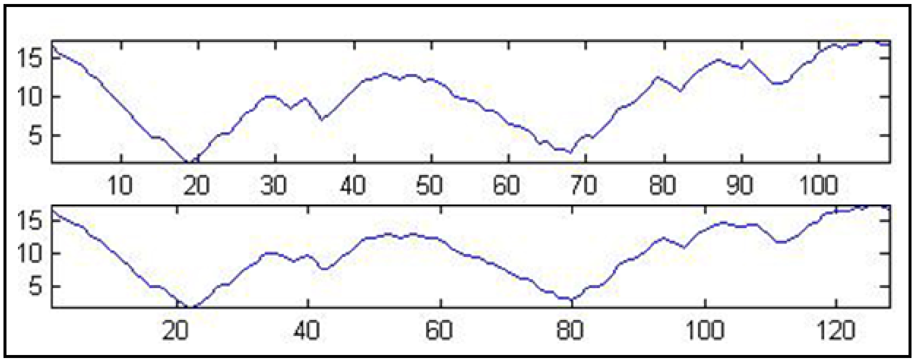


Figure 4.7: *An illustration of the extracted and normalized Centroid Distance Function shape signature for the character 'a'. the top Figure shows the Centroid Distance Function prior to resampling to length of 128, and the bottom Figure shows the same after resampling.*



DCT Features For generating the DCT features each input character is scaled in a similar manner to the GAMI features using bilinear interpolation so that the larger of its height or width is 32 pixels, and if required the other dimension is padded with zeros symmetrically resulting in a 32×32 pixel image without distorting the original proportions of the character. The 2-D DCT II transform is than performed on the entire character region and the low order 6×6 transform coefficients are grouped to form a 36 dimensional feature vector.

4.4.1.3 Classifiers

The four classifiers chosen for evaluation are: k-Nearest Neighbor (k-NN), Random Forest (RF), Support Vector Machines (SVM) and Naive Bayes (NB) represent a range of popular, successful and fundamentally different approaches to classification. K-NN is a simple, effective and very popular classifier which has proven successful in a multitude of applications. RF is an ensemble of decision trees where each of the trees is built by following a nondeterministic algorithm using a different subset of the training data for growing each tree. RF can be useful for classifying hard-to-separate data by dividing the feature space into a large number of sections where each section represents a different class output, but at the expense of lower generalization and possible overfitting. SVM is the perceptron of optimal stability performing classification by fitting a hyperplane that best separates the data in the maximum-margin sense. SVM is considered today one of the most effective classifier, especially when the data is “reasonably” linearly separable in the feature space. Finally NB is a probability-based classifier based on estimation of the distribution for the different classes, which has also proven to be useful in a large number of scenarios. For additional description and an in-depth discussion of these classifiers refer to Section 2.2. In the following experiments the Witten and Frank (1999) implementation was used for all classifiers.

Classifiers Parameters and Configuration For all of the performed experiments, unless otherwise specifically noted, the classifiers were configured with the following parameter values:

k-Nearest Neighbor (k-NN)

- $k = 3$.
- All components are standard score normalized so that components on different scales contribute equally to the distance function, $z_i = \frac{x_i - \mu}{\sigma}$, where z_i is the resulting normalized component, x_i is the raw non normalized component, μ is the mean value of the component and σ is the standard deviation of the component.
- Neighbors X_n $n = 1 \dots k$ votes are weighted by the inverse of their distance to the test observation, $1/d$ where d is the Euclidean distance between the neighbor X_n to the test observation X_t.

Note that in experiments 1-A and 1-B there is only one training observation of each class, therefore the weighted neighbors votes are in fact equivalent to nearest neighbor classification.

Random Forest (RF)

- P (number of trees) = 20.
- n (number of splitting components per node) = $(\log_2 N) + 1$ where N is the dimensionality of observations.

Support Vector Machines (SVM)

- Trained using SMO algorithm (Platt, 1998).
- Multi-class classification solved using pair wise classification (Krebel, 1999).
- Standard inner product kernel (linear kernel).
- All components are standard score normalized, $z_i = \frac{x_i - \mu}{\sigma}$, where z_i is the normalized component, x_i is the raw non normalized component, μ is the mean value of the component and σ is the standard deviation of the component.

Naive Bayes (NB)

- Estimating distribution by assuming single conditionally independent normal distribution per feature variable.

4.4.2 Experiments

Four experiments are performed for evaluating the accuracy of the different combinations of shape descriptors and classifiers, with respect to the ideal training observations and with the inclusion of different model-based synthetic training set sizes. Twelve feature-classifier combinations in total are evaluated, the three selected shape descriptors: Geometric Affine Moment Invariants (GAMI), Centroid Distance Function Fourier Descriptors (CDFD) and

Discrete Cosine Transform type II (DCT) when coupled with each of the following four classifiers: k-Nearest Neighbor (k-NN), Random Forest (RF), Support Vector Machines (SVM) and Naive Bayes (NB). The evaluation is performed on a uniform test set taken from real scanned documents images.

Experiment 1-A aims to evaluate how well the ideal noiseless examples translate to well-separated points in the feature space for the different features, as well as how well the Print-Scan degraded test data clusters in proximity to the points representing their ideal examples, whilst remaining separate from other classes. In experiment 1-B, the test is repeated with the same intention, however, confusions that occur in character pairs which are very similar (as discussed in Section 4.4.2.2) are not counted as errors to provide a comparison in terms of perceptual similarity.

Experiment 2-A aims to evaluate the relevancy of the synthetically generated training data in respect to real Print-Scan degradation, and its contribution to the classification accuracy. Moreover, this experiment evaluates the suitability of the different feature-classifier pairs when trained with model-based synthetic training data in accurately recognizing real Print-Scan degraded characters. In experiment 2-B, the test is repeated with the same intention, but as with experiment 1-B confusions between perceptually similar classes are not considered as classification errors.

An analysis of the results of experiments 1 and 2 and the conclusions drawn is then given, followed by an explanation to the motivation for performing the follow-up experiments number 3 and 4.

Experiment 3 aims to determine the optimal DCT feature dimensionality. In a similar manner to experiment 2-B, confusions between perceptually similar classes are not considered as classification errors.

Experiment 4 aims to further explore different combinations of feature dimensionality and training dataset size. Again, in a similar manner to experiment 3, confusions between perceptually similar classes are not considered as classification errors.

4.4.2.1 Experiment 1-A

In this experiment, the accuracy of the different combinations of shape descriptors and classifiers is measured when the training data is comprised of one ideal 300 DPI rasterized image per character class, a total of 60 images.

Table 4.1: *The error rates results for the different classifier-shape descriptor pairs evaluated in experiment 1-A. The rows represent the different classifiers and the columns represent the different shape descriptors.*

	GAMI	CDFD	DCT
NB	0.6156	0.7575	0.3186
KNN	0.6719	0.3010	**0.0978**
SVM	0.7485	0.3236	**0.0941**
RF	0.5477	0.5737	0.3780

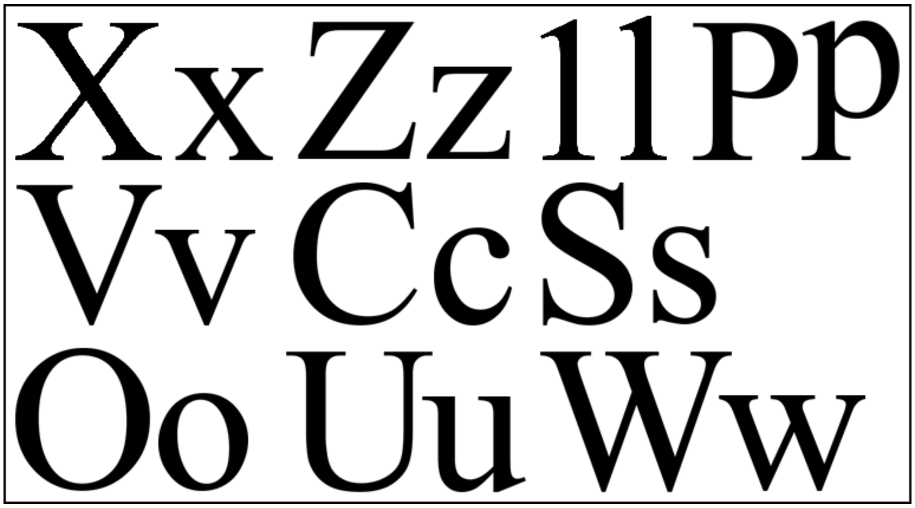
All classifiers were configured with the values specified in Section 4.4.1.3 except for k-NN and SVM which had normalization disabled.

Results The error rates *(misclassifications/total characters)* of experiment 1-A for the different descriptor-classifiers pairs are summarized in Table 4.1. Experiment 1-A results demonstrate that under the given conditions, DCT is consistently the most accurate shape descriptor by a large gap when coupled with all classifiers. The top five combinations when ranked by decreasing accuracy are: DCT-SVM, DCT-KNN, CDFD-KNN, DCT-NB and CDFD-SVM. The two most accurate pairs, DCT-SVM and DCT-KNN (bolded), are grouped closely with only 0.0037 separating the two. Following them by a large 0.2032 gap is the CDFD-KNN pair. The three least accurate pairs in decreasing error ratio are: CDFD-NB, GAMI-SVM and GAMI-KNN. A detailed analysis and discussion of these results is given later in Section 4.4.2.5.

4.4.2.2 Experiment 1-B

Experiment 1-B is conducted to account for the perceptual and mathematical (in terms of the calculated features) similarity that is inherent to several character groups in the English character set as discussed in the following Sections. Experiment 1-B is identical to experiment 1-A in terms of training and testing data, and classifier configuration, however confusions in the similar classes groups (discussed later in this Section) are not considered as classification errors. It is important to note that the relaxations allowed in this experiment are largely unhelpful to the DCT features since the DCT is unlikely to get confused between affine transformed characters except for scale-only transforms.

Figure 4.8: *Perceptually similar characters pairs in the Times New Roman 12 font.*

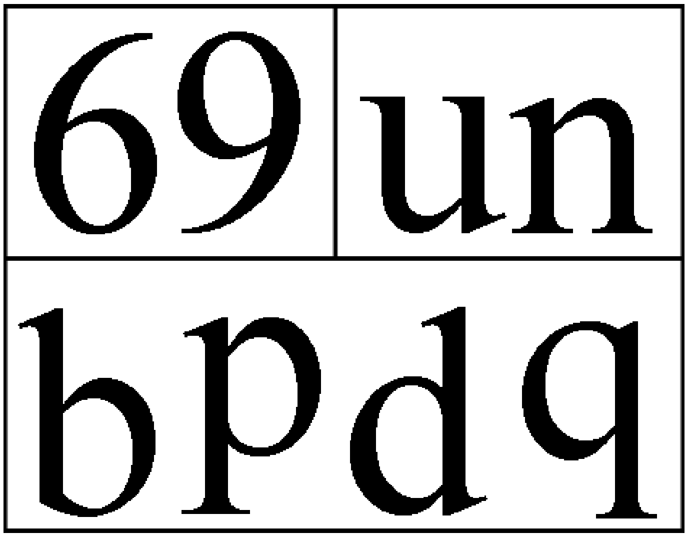


Perceptually Similar Characters It is important to note that Times New Roman 12 point font contains character pairs which are essentially a scaled version of one another, or resemble one another very closely, as illustrated in Figure 4.8. Moreover, the test characters extracted in practice from the documents have substantially lower resolution in pixels than the characters in Figure 4.8. As the resolution decreases, the rasterizing process diminishes the subtle differences between these character pairs, hence essentially they become more and more alike. Furthermore, increased amounts of noise further diminish the differences between these character pairs.

Since all three evaluated shape descriptors are invariant to scale (either by definition or by normalization of the input size prior to extracting the features), it is reasonable to assume that there would be a relatively high confusion rate for the character pairs presented in Figure 4.8. Moreover, it is very likely to assume that discriminating between those pairs when their size is normalized would prove to be a difficult and confusing task even for humans (although perhaps to a lesser extent for ‘P’-‘p’, ‘C’-‘c’ and ‘U’-‘u’ since they do have some perceptual distinction between their two versions).

Affine Transformed Similar Characters There are additional character groups in which the characters are affine transformations of one another (or very close to being such) and therefore are regarded as “perceptually similar” by shape descriptors which are invariant to the relevant affine transformations families.

Figure 4.9: *Groups of characters that are, or very close to being an affine transformations of the others (excluding only scale transforms) in the Times New Roman 12 point font.*



In particular GAMI are designed to be invariant to all affine transformations such as rotation, shear, reflection and others, and CDFD are derived to be invariant to rotation. For example, consider the following three characters ‘b’, ‘d’, ‘q’ and ‘p’, they are rotated and/or reflected versions of one another and therefore the GAMI features of these characters are very similar. In another example, consider the pair ‘6’ and ‘9’, since they are rotated versions of one another the CDFD features calculated for these characters are essentially almost the same. Therefore these features could result in high confusion rates between these classes especially in the presence of noise. In figure 4.9 the groups of characters that are affine transformations of the others (excluding scale-only transforms) in the Times New Roman 12 point font are presented.

Results The error rates *(misclassifications/total characters)* resulting from experiment 1-B are summarized in Table 4.2. These results confirm once more that under the conditions of this experiment DCT is consistently and by far the most accurate shape descriptor when coupled with all classifiers. The top five combinations in increasing error ratio are identical to experiment 1-A: DCT-SVM, DCT-KNN, CDFD-KNN, DCT-NB and CDFD-SVM. The two most accurate pairs, DCT-SVM and DCT-KNN (bolded), are now even closer to one another with almost no difference between the two.

Table 4.2: *The error rates results for the different classifier-shape descriptor pairs evaluated in experiment 1-B. The rows represent the different classifiers and the columns represent the different shape descriptors.*

	GAMI	CDFD	DCT
NB	0.5577	0.7256	0.2498
KNN	0.6494	0.2332	**0.0256**
SVM	0.7116	0.2532	**0.0250**
RF	0.5499	0.5027	0.2942

In third place is CDFD-KNN with a 0.2076 gap to second place, about the same gap as with experiment 1-A. The three worst pairs in decreasing error rate order are identical to experiment 1-A: CDFD-NB, GAMI-SVM and GAMI-KNN. A detailed analysis and discussion of these results is given later in Section 4.4.2.5.

4.4.2.3 Experiment 2-A

In this experiment, the accuracy of the different combinations of shape descriptors and classifiers is measured when the training data is comprised of one ideal and 100 synthetically generated 300 DPI rasterized images per character class, a total of 6,060 images. In this experiment the classifiers were configured with the values specified at Section 4.4.1.3.

Results The error rates *(misclassifications/total characters)* of experiment 2-A for the different classifier-descriptor pairs are summarized in Table 4.3. Under the conditions of this experiment DCT is again the most accurate shape descriptor by far when coupled with all classifiers. The top five combinations in increasing error ratio are: DCT-SVM, DCT-KNN, DCT-NB, DCT-RF and surprisingly GAMI-RF. The two most accurate pairs, DCT-SVM and DCT-KNN (bolded), are grouped very closely with only 0.0087 separating the two in favour of DCT-SVM. In third place is DCT-NB with a small 0.0025 gap to second place, a substantial decrease of about 0.2 in the gap from the second to the third place in experiment 1-A. The three worst pairs in decreasing error rate are: GAMI-SVM, GAMI-KNN and CDFD-NB.

Table 4.3: *The error rates results for the different classifier-shape descriptors pairs evaluated in experiment 2-A. The rows represent the different classifiers and the columns represent the different shape descriptors.*

	GAMI	CDFD	DCT
NB	0.3223	0.4683	0.0340
KNN	0.5018	0.2889	**0.0315**
SVM	0.8967	0.2710	**0.0228**
RF	0.0925	0.2767	0.0559

These results display some expected outcomes and also reveal a few surprising phenomenon. As expected, almost all pairs displayed significant improvement in accuracy with the addition of 100 synthetic training samples per class. The only surprising exception is GAMI-SVM which displayed a large *increase* in error rate from about 0.75 (which is very low to begin with) to about 0.9 when additional training data was provided. In contrast the pair GAMI-RF showed for the first time somewhat reasonable results for the GAMI descriptor with an error rate of about 0.09. A detailed analysis and discussion of these results is given later in Section 4.4.2.5.

4.4.2.4 Experiment 2-B

Experiment 2-B is identical to experiment 2-A in terms of training and testing data, and classifier configuration, moreover, in a similar manner to experiment 1-B, confusions in the aforementioned similar classes pairs are not considered as classification errors. This is done to account for the perceptual similarity between some of the classes as described in Section 4.4.2.2.

Results The error rates resulting from experiment 2-B are summarized in Table 4.4. As expected, improvement in all descriptor-classifier pairs is observed following relaxation of the confusion constraints for similarly looking character classes. The top five combinations in increasing error ratio are identical to experiment 2-A: DCT-SVM, DCT-KNN, DCT-NB, DCT-RF and GAMI-RF.

Table 4.4: *The error rates results for the different classifier-shape descriptors pairs evaluated in experiment 2-B. The rows represent the different classifiers and the columns represent the different shape descriptors.*

	GAMI	CDFD	DCT
NB	0.2448	0.4302	0.0075
KNN	0.4549	0.2219	**0.0053**
SVM	0.8695	0.2082	**0.0025**
RF	0.0647	0.2091	0.0187

The two most accurate pairs, DCT-SVM and DCT-KNN (bolded), are now closer to one another than in experiment 2-A with 0.0028 error rate separating the two in favour of DCT-SVM. In third place is DCT-NB with a very small 0.0021 gap to second place. The three worst pairs in decreasing error rate order are identical to experiment 2-A: GAMI-SVM, GAMI-KNN and CDFD-NB.

4.4.2.5 Analysis and Discussion

GAMI Results As evident from the performed experiments GAMI typically exhibit very poor results when coupled with all classifiers except for RF ranging from 0.8967 (2-A GAMI-SVM) to 0.0187 (2-B GAMI-RF) with an average error rate of 0.5312 across all experiments. To eliminate the possibility that the poor accuracy is due to the sub-block division, experiment 1-A was repeated when the GAMI were applied to the entire character (without block division) and worse results were observed. Very interesting is the fact that when substantially increasing the size of the training set from one to 101 examples per class, the deterministic linear kernel SVM classifier finds it more difficult to calculate the optimal maximum margin splitting hyperplanes resulting in a significant 0.8695 error rate. In contrast, the nondeterministic RF classifier, which uses a different subset of the training set each time the model is built, achieves an error rate of 0.0647 under the same conditions. These results suggest that the GAMI features are linearly inseparable in Hilbert Space and therefore attempts to perform generalization of the decision by splitting hyperspaces results in high error rates. On the contrary, RF classifier is comprised of an ensemble of random trees, where each tree essentially defines a different split on a different section of the original space. Therefore RF classifier is able to separate the feature space through different tree voting combinations into smaller sections where each section results in the output of a different class, which in the case of GAMI features, proves to be a valuable property. A graphic example illustrating a similar scenario with the RF classifier is given in Figure 2.6.

CDFD Results CDFD has displayed results with error rates ranging from 0.7575 (1-A CDFD-NB) to 0.2082 (2-B CDFD-SVM) with an average error rate of 0.3778 across all experiments. The reasons for CDFD's failure are varied. One such reason is the occasional joining of formerly disjoint parts in the characters due to increased noise levels which create new holes and thus result in extreme deformations of the perimeter as illustrated in Figure 4.10. Another possible reason is the breaking up of characters which may alter the shape's perimeter quite substantially especially when the character contains large holes as can be observed in Figure 4.11. Degradations such as presented in Figures 4.10 and 4.11 certainly have less effect on GAMI and DCT than their effect on the Centroid Distance Function shape signature which is used to generate CDFD.

And finally, another important possible reason for the low accuracy displayed by CDFD is the substantial loss of data occurring when the phase information is discarded when calculating $\{b_n\}$ (see Section 2.1.4.2). To estimate the effect of discarding the phase information when calculating the invariants $\{b_n\}$ an experiment was performed in which a shapes perimeter was reconstructed by applying the IDFT on the invariants $\{b_n\}$ calculated from the Complex Coordinates shape signature (Granlund, 1972). The characters reconstructed as part of this experiment were very different in comparison to the original characters to the point where they were unrecognizable, and moreover, they were perceptually very similar to one another. In Figure 4.12 a number of original and reconstructed characters by applying the IDFT to the Complex Coordinates $\{b_n\}$ are presented.

This implies that the phase information contributes to accurately describing the shape and that the method in which the invariants $\{b_n\}$ are calculated results in substantial loss of information which results in decreased accuracy of the CDFD.

Figure 4.10: *Examples of joining of formerly disjoint parts in characters due to increased noise levels which create new holes and thus result in extreme deformations of the perimeter.*

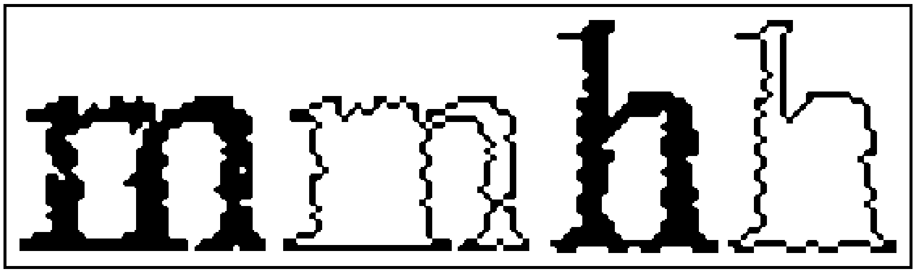


Figure 4.11: *Examples of breaking up in characters due to increased noise levels which result in extreme deformations of the perimeter.*

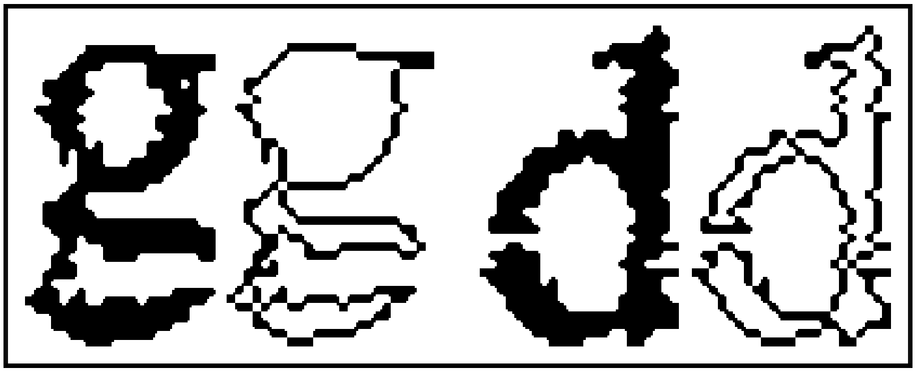
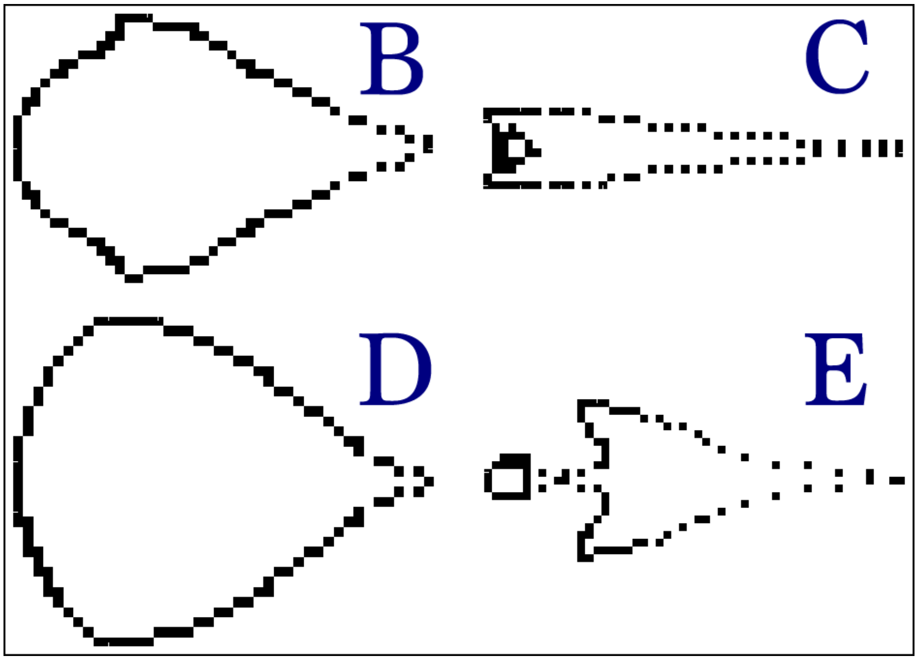


Figure 4.12: *The original (in blue) and reconstructed (in black) 'B', 'C', 'D' and 'E' characters generated by applying the IDFT to the Complex Coordinates $\{b_n\}$ as suggested in Granlund (1972).*



DCT Results In contrast to GAMI and CDFD, DCT based features displayed consistently excellent results and impressive robustness to significant Print-Scan degradation when coupled with all classifiers. In experiment 1-B, DCT-SVM and DCT-KNN have obtained 0.0250 and 0.0256 error rates respectively. This result is very impressive considering the dimensionality of the features was 36 and the training set size was only 60 observations, one ideal observation per class, a ratio of 60% between the dimensionality to the training set size. Since both linear SVM and nearest neighbor classifiers achieved similar results with the minimum possible training data, despite the "curse of dimensionality" and/or the peaking phenomenon (Sima and Dougherty, 2008; Elkan, 2011), it indicates that the DCT features are separated well in the feature space (Elkan, 2011). Moreover, since the error rate in experiment 2-B has decreased substantially to 0.0025 with the additional synthetically generated training data, it indicates that:

- The synthetic training data reasonably imitates the degradations generated by the Print-Scan Channel, at least in terms of the feature space.
- The impact of both the real Print-Scan Channel degradations and the model-based degradations on the DCT based features are such that the features calculated from

> both the synthetic generated training data and from the real test samples cluster together in the feature space, but cluster separately to other classes. This in turn allows the different character classes to be linearly well separated, which translates to high accuracy by both classifiers.

Both DCT-SVM and DCT-KNN have displayed excellent and very similar accuracy with a slight advantage to DCT-SVM. Furthermore as discussed in Section 2.2, once the model is generated, SVM is more efficient. Therefore SVM is the most reasonable choice for the classifier to pair with the DCT based features.

Since the dimensionality of the features, the size of the training set and different classifier's parameters affect the overall accuracy it is required to further evaluate different combinations of these factors in order to determine the overall optimal configuration.

Conclusion In light of the results obtained in experiments 1 and 2, it is concluded that both GAMI and CDFD are not suitable for robust character recognition in Print-Scan degraded noisy documents. On the other hand, DCT has displayed overall excellent accuracy and an impressive resilience to the substantial Print-Scan noise present in the test data. Moreover, the DCT is relatively straightforward and fast to compute with the FFT algorithm, does not require any special prerequisites, and can be conveniently coupled with all classifiers. Therefore, it is concluded that it is suitable as a shape descriptor for the purpose of accurately classifying characters in Print-Scan degraded documents images.

However, additional tests are required to optimize the performance of DCT-SVM combination. In particular, the effect of the dimensionality of the DCT features and amount of training data on the overall accuracy should be evaluated. Consequently two additional follow-up experiments are now performed.

4.4.2.6 Experiment 3

The aim of this experiment is to determine the optimal DCT feature dimensionality, when in a similar manner to experiment 2-B, confusions in the aforementioned similar classes pairs are not considered as classification errors.

Table 4.5: *The error rates results for the different DCT feature dimensions evaluated in experiment 3. The columns represent the different error rates obtained when changing the selection of coefficients as features.*

5×5 - 25 Dim	6×6 - 36 Dim	7×7 - 49 Dim	8×8 - 64 Dim	10×10 - 100 Dim
0.0053	0.0031	0.0031	0.0031	0.0037

The training data was identical to experiment 2-B, comprised of one ideal and 100 synthetically model-based generated 300 DPI rasterized images per character class, a total of 6,060 images. The SVM classifier was configured with the values specified at Section 4.4.1.3.

Results The error rates resulting from experiment 3 are summarized in Table 4.5. The columns represent the different error rates obtained when changing the selection of coefficients as features. In all cases the features selected are the vertical × horizontal low order coefficients as indicated in the Table. These results show that there is little difference between the error rates of 6×6, 7×7 and 8×8, with 10×10 somewhat reduced in accuracy.

4.4.2.7 Experiment 4

The aim of this experiment is to further explore different combinations of feature dimensionality and training dataset size, when in a similar manner to experiment 3, confusions in the aforementioned similar classes pairs are not considered as classification errors.

In all tests the training data is always comprised of one ideal and additional number of synthetically model-based generated 300 DPI rasterized images per character class as indicated by Table 4.6. The training set is grown through a method of expansion, which means that smaller training sets are proper subsets of larger sets. The SVM classifier was configured with the values specified at Section 4.4.1.3.

Results The error rates resulting from experiment 4 are summarized in Table 4.6. The columns represent the different error rates obtained when changing the selection of coefficients as features. In all cases the features selected are the vertical × horizontal low order coefficients as indicated in the Table.

Table 4.6: *The error rates results for the different DCT feature dimensions evaluated in experiment 4. The columns represent the different error rates obtained when changing the selection of coefficients as features. The rows represent the amount of training data used to train the classifier.*

	6×6 - 36 Dim	8×8 - 64 Dim	10×10 - 100 Dim
1 ideal + 200 synthetic	0.0037	0.0031	0.0040
1 ideal + 300 synthetic	0.0043	0.0028	0.0028

Table 4.7: *A summary of the error rates results obtained in experiments 3 and 4. The columns represent the different error rates obtained when changing the selection of coefficients as features. The rows represent the amount of training data used to train the classifier.*

	5×5	6×6	7×7	8×8	10×10
1 ideal + 100 synth	0.0053	0.0031	0.0031	0.0031	0.0037
1 ideal + 200 synth	N/A	0.0037	N/A	0.0031	0.0040
1 ideal + 300 synth	N/A	0.0043	N/A	**0.0028**	**0.0028**

The results for experiments 3 and 4 are summarized in Table 4.7. The lowest overall error rate was obtained when using either 8×8 or 10×10 low order of DCT coefficients classified with the linear SVM classifier trained with 1 ideal + 300 synthetic observations per class. These results show that if the dimensionality of the features is sufficient, then the inclusion of additional training data improves the overall accuracy, but as the size of the training sets and the dimensionality of the features continue to grow, the improvement becomes marginal, as expected.

4.5 Summary

In this Chapter the accuracy of different combinations of shape descriptors and classifiers was evaluated, with respect to the ideal training observations and with the inclusion of different model-based synthetic training set sizes. Twelve feature-classifier combinations in total were evaluated: the three selected shape descriptors: Geometric Affine Moment Invariants (GAMI), Centroid Distance Function Fourier Descriptors (CDFD) and Discrete Cosine Transform type II (DCT) when coupled with each of the following four classifiers: K-Nearest Neighbor (k-NN), Random Forest (RF), Support Vector Machines (SVM) and Naive Bayes (NB). The evaluation was performed on a uniform test set taken from real scanned documents images.

The experiments show, that both GAMI and CDFD are not suitable for robust character recognition in Print-Scan degraded noisy documents. The results suggest that GAMI features are linearly inseparable in Hilbert Space and therefore attempts to perform generalization of the decision by a perceptron classifier such as linear SVM results in high error rates. CDFD display poor results probably due to the substantial corruption in the boundary of the characters when high levels of Print-Scan corruption is present, in addition to the significant loss of information occurring once the phase information is discarded when calculating the invariants $\{b_n\}$.

Fourier descriptors were suggested to be used as the main shape feature in the content-based verification system suggested by Ming *et al.* (2007). However, due the observed poor performance of CDFD in the experiments performed as part of this thesis, the system suggested by Ming *et al.* (2007) would suffer a high false rejection rate under conditions of significant Print-Scan degradation. This is due to the inherent problems of perimeter-based character recognition in noisy conditions such as the self-joining of and breaking of characters part which result in substantial changes to the perimeter of the shape.

On the other hand, DCT has displayed overall excellent accuracy with all classifiers and an impressive resilience to the substantial Print-Scan noise present in the test data. From all of the evaluated classifiers combinations, both DCT-SVM and DCT-KNN have displayed outstanding and very close accuracy with a slight advantage to DCT-SVM. It is interesting that even when the training set was comprised of only one ideal observation per class, DCT still achieved reasonable accuracy. Furthermore, with the inclusion of additional model-based synthetic generated training data a substantial decrease in the error rate down to 0.0028 was observed. Therefore, it is concluded that *(i)* the synthetic training data reasonably imitates the degradations generated by the Print-Scan Channel, at least in terms of the feature space; *(ii)* the impact of both the real Print-Scan Channel degradations and the model-based degradations on the DCT based features are such that the features calculated from both the synthetic generated training data and from the real test samples cluster together in the feature space, but cluster separately to other classes. This in turn allows the different character classes to be linearly well separated, which translates to high accuracy by both classifiers.

Moreover, the DCT is relatively straightforward and fast to compute with the FFT algorithm, does not require any special prerequisites, and can be conveniently coupled with all classifiers. Therefore, it is concluded that the DCT is suitable as a shape descriptor for the purpose of accurately classifying characters in Print-Scan degraded documents images.

The best overall results were achieved by the DCT-SVM when features were comprised of the 8×8 low order DCT coefficients classified using linear SVM trained with 1 ideal + 300 synthetic observations per class. Therefore it is concluded that this combination is to be used in the proposed framework and taken forward to Chapter 5.

Chapter 5

Verification-Driven Segmentation in Print-Scan Degraded Documents

In the previous Chapter, evaluation of shape features and classifiers suitable for recognition of characters in noisy Print-Scan degraded documents was performed. It was concluded that DCT-based shape features classified by the Support Vector Machines classifier are suitable for successfully recognizing characters in noisy print scan degraded documents.

In this Chapter, a detailed discussion of the methods used for performing verification of noisy documents images is given. This Chapter commences by presenting the typical characteristic phenomenon occurring in degraded document images. Following are the descriptions of the varied methods used for performing verification and for overcoming the characteristic difficulties found in noisy documents images. And finally, the results of an empirical evaluation of the proposed framework are presented and discussed.

5.1 Inherent and Degradation Related Characteristic Phenomenon in Document Images

Accurate classification of individual symbols in the presence of increased degradation is necessary for automatic robust document verification but is not sufficient on its own for performing accurate verification. Real life Print-Scan Channel degraded document images frequently require handling additional complications due to the inherent nature of the documents or due to degradation related characteristic phenomenon such as:

- Transformations.
- Multipart characters.

- Broken characters.
- Joined characters.

The following Sections discuss each of these issues and explore their effects and extent on scanned document images in order to define the problems that must be resolved for achieving a successful verification framework.

Before continuing the discussion, note that throughout the remains of this Chapter, unless otherwise specifically noted, the examples given in the figures were taken from test documents featuring Times New Roman 12 point font. The documents were printed and scanned repeatedly three times in order to produce significant levels of Print-Scan degradation using a HP 7500A model standard inkjet consumer all-in-one printer and scanner on standard general use office paper. The printing was done using the default standard settings, and the scanning was performed using 300 DPI grayscale settings. These examples are 400% magnified binarized characters taken from the aforementioned documents.

In addition, the term “false-positives” is defined as verification errors where characters that were not changed are determined to have been changed. And the term “false-negatives” is defined as inobservance of changes in characters, or in other words, failure to detect tampered characters by the verification system.

5.1.1 Transformations

The printing and scanning of the documents presented for verification are occurring outside of the domain of the system, being performed by a human user with access to printed copies of the original document. Moreover, it is possible that the document has been printed and scanned multiple times before being presented for verification. Hence there are many unknown factors which affect the properties of the Print-Scan Channel and therefore the final channel’s output (e.g. the scanned document image which is presented for verification). One class of degradations that are of particular interest are the global affine transformations occurring in the document image such as translation, scaling, skew and rotation as they can significantly change the shape, location and orientation of the characters in the verified document image. These transformations are caused by a myriad of user related and physical factors which affect the final image produced by the print-scan process (Baird, 2000; Lins, 2009). However, from all of the contributing factors there are a number which are dominant in particular in their contribution to transformations, specifically:

- The number of print-scan repetitions which may result in accumulated transformations.
- Paper positioning.
- The type of equipment used for printing and scanning and the physical properties and/or the operational settings of the printing and scanning equipment.

These factors and their large number of resulting transformations imply that theoretically no two Print-Scan Channel's outcomes are identical (Zhu *et al.*, 2003). Furthermore, some of the prominent global linear conformal transformations translation, scaling and rotation must be estimated and corrected for the following reasons:

- The shape descriptor selected in this thesis, 2-D DCT II, is sensitive to rotation and therefore substantial rotation could potentially result in deterioration of classification accuracy.
- Some character pairs such as ‘d’-‘p’, ‘m’-‘w’, ‘6’-‘9’, ‘\’-‘/’ and ‘(’-‘)’ for example, can be considered very similar but rotated version of one another, therefore if the rotation is substantial enough, these characters change their perceptual meaning.
- Translation, rotation and scaling result in changes of the location and size of the characters. As explained in Section 5.2.5, the expected location and dimensions of the characters stored by the DDM is used to assist in the accurate segmentation of the document image to separate characters. Furthermore, the characters recognized in the document image are compared against the characters expected to be found, and any discrepancies cause the document to be rejected. Therefore, it is important that the characters in the document image would be located as close as possible to their expected ideal locations.

A description of how the linear conformal transformations discussed above are estimated and corrected is given in Section 5.2.2.1.

5.1.2 Multipart Characters

Characters may be comprised of several parts, where some or all of the parts are different in shape. In the English language the following characters are examples of multi-part characters: ‘i’ and ‘j’ from the alphanumeric group; colon, semicolon, question-mark, exclamation and quotation-marks from the punctuation group; and percentage and equal symbols from the special characters group.

Recognizing multipart characters cannot usually be performed by using only basic shape recognition techniques such as the techniques discussed in Section 2.1 because these rely on lower level pre-processing techniques such as connected components analysis to segment and extract symbols. While standard shape recognition methods may be used to recognize the different characters' parts, some higher level logic must be applied to recognize the character as the sum of its parts. This task might be further complicated if any of the parts break down into smaller fragments due to degradation, which is discussed in details in Section 5.1.3 below.

5.1.3 Broken Characters

As shown in Figure 4.4 (which is brought here again for convenience as Figure 5.1), with each repeating printing and scanning cycle, more irregularities are formed in the characters. These irregularities manifest both in the shape's region and at the shape's boundary. Moreover, the contrast between the characters and the paper (foreground and background pixel populations) is diminished with each repeating print-scan cycle. This is expressed in increased overlapping between the intensity levels of the foreground's and background's pixels and increases the separation error or equivalently decreases the separation effectiveness achieved by binarization methods.

To provide a quantitative insight into the effects of degradation on binarization, Table 5.1 shows the optimal threshold value and the separation effectiveness η^* as defined in Equations (1)-(15) of Otsu (1979) for a document image in five levels of degradation, starting at the ideal document image (zero print-scans) and up to four repeating accumulated print-scan cycles. It can be observed that both values decrease steadily as the degradation levels increase. These values confirm the observations previously made in this Section.

Due to this phenomenon, as the level of degradation increases, binarized characters are more often erroneously segmented to several disjoint character fragments. In Figure 5.2, examples of characters broken into two or more fragments are given.

Figure 5.1: *(A repeat of Figure 4.4) The cumulative degradation generated after each additional print-scan cycle is shown in increasing order from (a) to (c).*

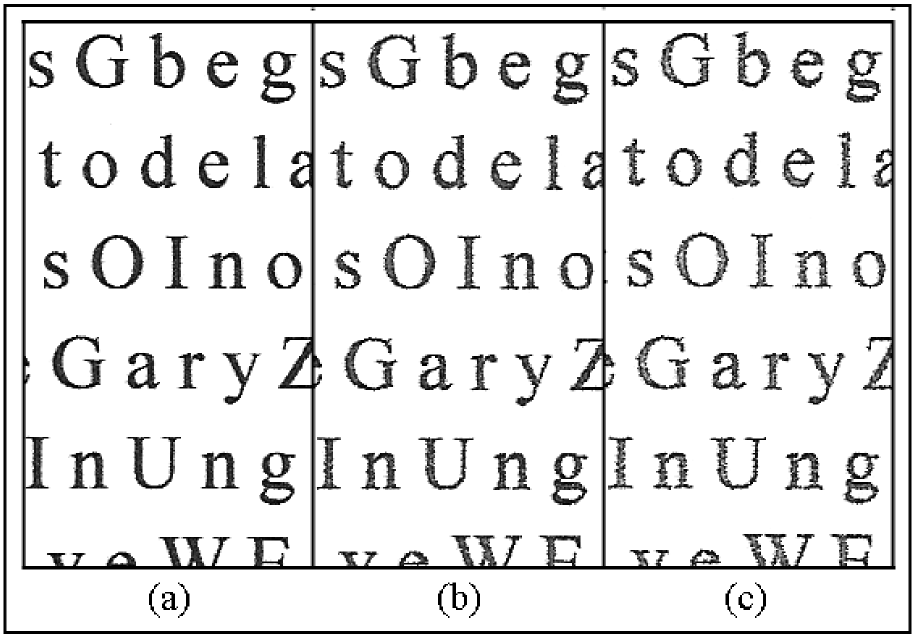
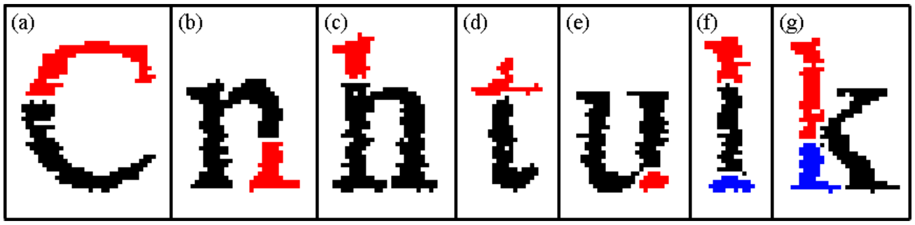


Table 5.1: *The calculated optimal threshold value and the separation effectiveness η^* as defined in Equations (1)-(15) of Otsu (1979) for a document image in five levels of print-scan degradation, starting at the ideal document image (0 print-scan) and up to four repeating accumulated print-scan cycles.*

	0 print-scan	1 print-scan	2 print-scan	3 print-scan	4 print-scan
Threshold	0.4980	0.4294	0.4020	0.3784	0.3725
η^*	0.9705	0.9622	0.9482	0.9310	0.9153

Figure 5.2: *Examples of characters broken into two or more fragments taken from real scanned document image.*



The main difference between broken characters and multipart characters lies in the non-deterministic nature of the breaking process that result from a combination of stochastic irregularities in the brightness levels of groups of pixels and selection of binarization threshold value. Therefore, the resulting character segments may be created in a multitude of combinations of highly varied shapes and forms. This is in contrast to the multipart characters which in most cases can be assumed to follow a known pattern of composition of expected parts.

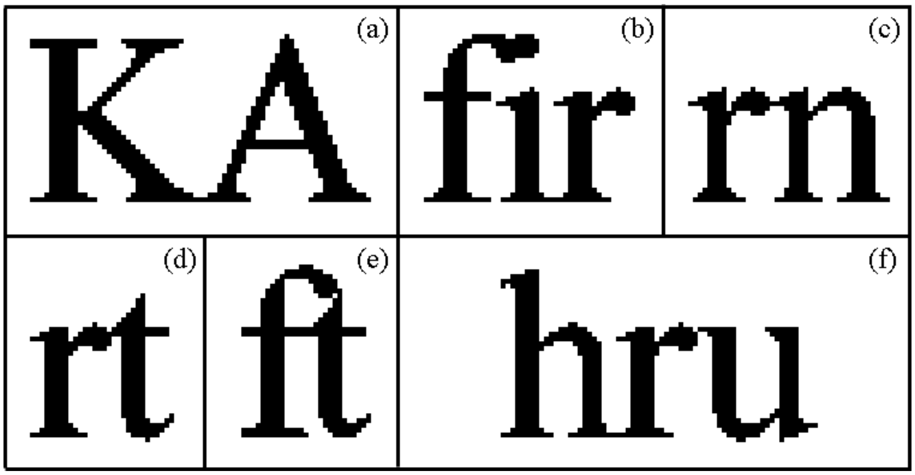
5.1.4 Joined Characters

The joining of characters can occur due to two different main scenarios referred to as Type I Joins and Type II Joins in this thesis.

Type I Joins are the joining of characters due to their placement in the digital document. Depending on the characters' size, shape and location, certain characters are placed in the document in such way they are drawn one on top of the other. In Figure 5.3, examples of 400% magnified and binarized Times New Roman 12 point font characters taken from an ideal noiseless digital image are given. Whilst most of the Type I joins are only due to a narrow "bridge" connecting the neighboring characters, this would be enough to cause connected components labeling to return a single blob that actually represents two or more characters. Type I joins cannot be ignored since they occur in the ideal document image, which is the basis for the printed document image.

In contrast, Type II Joins occur due to the increasing levels of noise and irregularities that are introduced by the repetitive print-scan cycles which occasionally result in the

Figure 5.3: *Examples of 400% magnified and binarized ideal Type I joined characters in the Times New Roman 12 point font.*



thickening of the binarized characters' strokes, especially in places where the distance between neighboring characters is relatively small. As a result the gap between neighboring characters is diminished so that after binarization the two characters are joined to form a single connected component. Examples of joined characters due to Print-Scan Channel degradation are given in Figure 5.4.

To illustrate the phenomenon of joined characters, a measurement was performed as part of this thesis. It has been observed that the occurrence of joined characters increases substantially with each additional print-scan cycle. The number of occurrences of joined characters and the mean number of characters joined per occurrence (chain length), were measured in a document comprised of 974 characters in the Times New Roman 12 point font. The measurement was performed on the ideal noiseless document bitmap image as well as on four scanned document images featuring four different levels of corruption obtained by repetitively printing and scanning the document up to four times. The threshold value for each document was determined globally using Otsu's method (Otsu, 1979). The results of this measurement are summarized in Table 5.2. Note that 0 print-scan refers to the ideal noiseless bitmap representation of the digital document.

These measurements clearly show that under the terms of this experiment, even the ideal noiseless bitmap representation of the digital document suffers from joined characters occurrences due to Type I Joins. Furthermore, there is a rapid increase in the number

Figure 5.4: *Examples of 400% magnified and binarized print-scan degraded Type II joined characters in the Times New Roman 12 point font.*

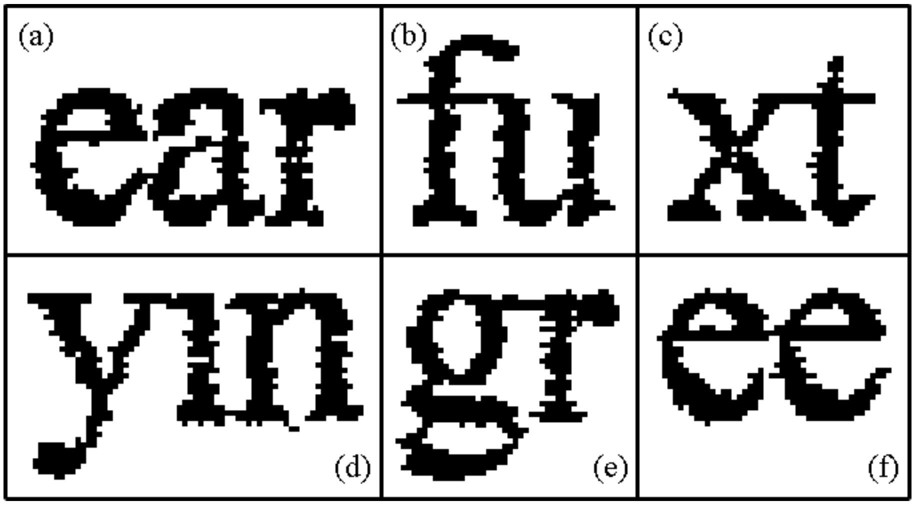


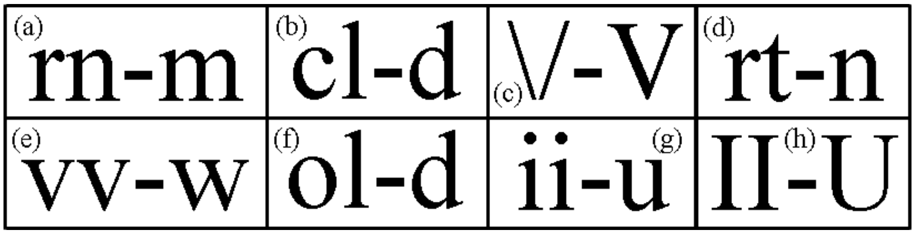
Table 5.2: *The number of joined characters occurrences and the mean length of the joined characters chains in a print-scan degraded document image in five levels of print-scan degradation, starting at the ideal document image (0 print-scan) and up to four repeating accumulated print-scan cycles.*

	0 print-scan	1 print-scan	2 print-scan	3 print-scan	4 print-scan
Occurrences	16	34	94	191	243
Mean chain len	2.0625	2.1176	2.1915	2.2723	2.5679

of Type II Joins and in the average length of the joined characters chains as the level of degradation increases. The test document of 974 characters was grouped in 168 words, with an average word length of 5.7976. These results imply that after as little as three repetitive print-scan cycles, with a very high probability, most, if not all of the words in the document suffer from one or more occurrence of joined characters.

The increased frequency of joined character occurrences, can be explained by the physical properties of the Print-Scan Channel such as defocusing (point spread function) and the way in which the spreading and/or smearing of ink/toner occurs in the paper (Baird, 2000; Eikvil, 1993). It is however important to note that the number of joined characters in a particular document depends on many factors such as the font type used, document formatting, printing and scanning technology and settings, threshold selection method

Figure 5.5: *Illustration of character joins that result in new shapes that resemble other characters.*



and other factors. Therefore, the results obtained as part of these measurements do not necessarily accurately apply to all document images, however it serves to highlight the potential scale of the problem.

Furthermore, another issue of significance resulting from joining of neighboring characters is the phenomenon where the new shape produced by the join perceptually resembles a different character in the "legitimate" character set. This phenomenon depends on factors such as font type, page formatting and the location where the joining occurs. For example consider the following combinations 200% magnified 300 DPI ideal Times new Roman 12 point font given in Figure 5.5.

Such combinations impose additional complexity in the segmentation and classification stages since their presence increases the number of different segmentation and classification possibilities for a give chain of joined characters and therefore are likely to contribute to increased recognition errors.

5.1.5 Change in Perceptual Identity

Change in perceptual identity of characters due to Print-Scan Channel noise is the phenomenon where the character is transformed into a new shape that perceptually resembles a different character in the "legitimate" character set. The following Sections expand upon this phenomenon and discuss two common scenarios that may result in a change to the perceptual identity of a printed and scanned character: self-joining and opening of holes.

5.1.5.1 Self-Joining

Self-joining in characters is the phenomenon where two or more near but separated areas of a character are bridged. Self-joining typically occurs due to the increasing levels of noise and irregularities that are introduced with increasing levels of degradation which occasionally result in the thickening of the binarized characters' strokes, especially in places where the distance between the character's parts is relatively small. Self-joining typically results in the formation of new holes in affected characters. Examples of self-joining can be observed in Figures 5.6 (a), 5.6 (b), 5.6 (d) and 5.6 (e).

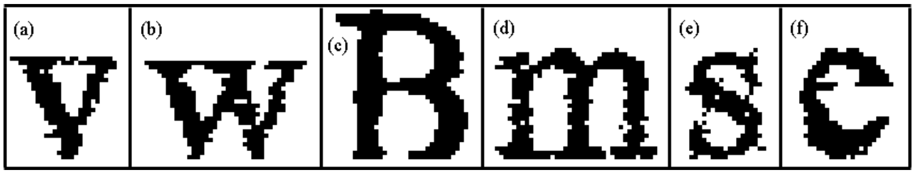
5.1.5.2 Opening of Holes

Another characteristic phenomenon which is essentially the opposite of self-joining is the elimination of holes which occur when a character's region, usually thin, is transformed from foreground to background when binarization is applied.

Even when the change in the affected character is small, these types of defects can have a substantial impact on some of the shape descriptors such as the boundary based Fourier descriptors and on other global features such as Euler number which together are a major part of the recognition methods used in the verification method suggested by Ming *et al.* (2007) (see Section 4.4.2.5). Moreover in cases where the change is sufficiently large this type of defects may potentially perceptually change a character's meaning such as can be observed in the character pairs of Figures 5.6 (c), 5.6 (e) and 5.6 (f), where 'B' become similar to 'R', 's' become similar to '8', and 'e' is almost completely morphed to 'c'. Moreover, these type of defects can also create new invalid symbols as can be observed in Figures 5.6 (a), 5.6 (b) and 5.6 (m).

As can be observed these types of defects can perceptually change the meaning of the character to become a completely invalid symbol, or morph towards a different symbol. In cases where these defects change the shape of the character substantially it usually leads to classification errors that, from a verification point of view, are in fact justified in the cases of substantial character changes. Therefore these false-positive classification errors would translate to local rejection of the verified documents which could then be further inspected by a human operator and be confirmed or rejected. This is preferable over the problem of false-negatives in which intentional tampering or other substantial changes to the document are not detected, which is naturally much harder to detect by a human operator.

Figure 5.6: *Examples of 400% magnified binarized characters from real document images resulting in self-joining, and elimination or formation of new holes.*



In the next Section a detailed discussion of the methods suggested to solve the aforementioned issues is given.

5.2 Proposed Approach

In this Section, the methods used in the framework are discussed. This includes methods for segmentation and classification that are proposed to overcome the aforementioned inherent difficulties and characteristic phenomenon that occur in Print-Scan Channel degraded document images as described in Section 5.1.

5.2.1 Identifying the Verified Document

To perform recognition, the document image must be identified as a printed copy of a digital document as a basis for comparison and the corresponding DDM must be retrieved. As discussed in Section 3.1.2, the information identifying the document and/or the DDM can be stored in a secure manner on the document in the form of a 2-D barcode.

5.2.2 Pre-Processing

Considering the scanned document image is a transformed and noisy variant of the ideal noiseless document image a number of different low-level techniques are employed in order to assist with achieving a number of crucial tasks such as the proper alignment of the scanned image, the separation of the foreground from the background, the suppression of noise and the segmentation of the characters. These methods are commonly cumulatively discussed under the "pre-processing" umbrella term.

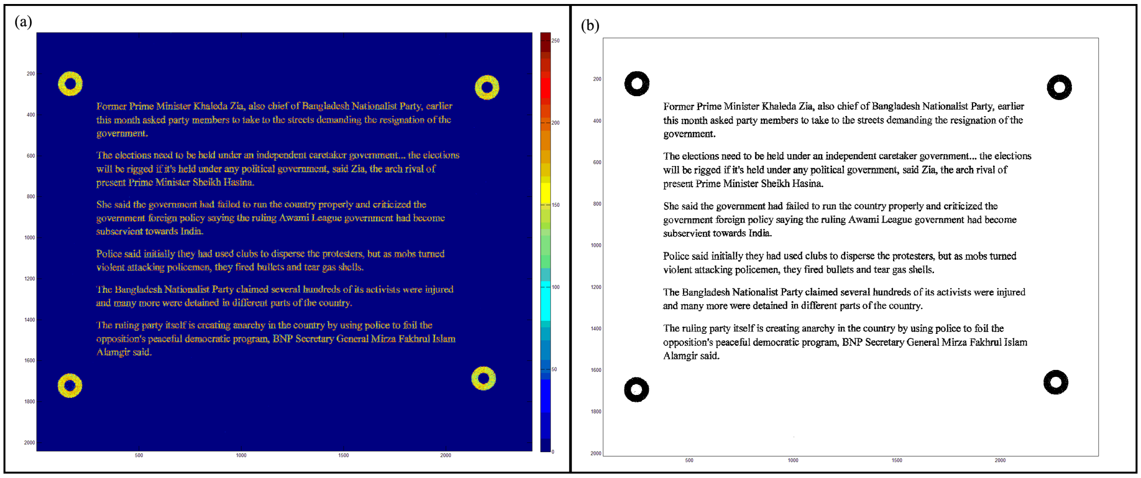
5.2.2.1 Estimation and Correction of Global Linear Conformal Transformations

Accurate image registration is required by the segmentation algorithm and for verifying the perceived symbols in the document by matching against the original digital document contents. In the current implementation, special graphic symbols are embedded on the document (usually at the corners of the document) to assist with the estimation and correction of global document transformations scale, rotation and translation (see Section 3.3.2 for details). The locations and shape(s) of these embedded designated graphic symbols may be unique for each document and as such they may be stored as part of the DDM. However, other methods exist for image registration such as RANSAC (Fischler and Bolles, 1981) which do not necessarily require the aid of any modifications to the document to perform registration of the scanned document image with the ideal document image, but at the cost of computational complexity and possibly lower accuracy in registration.

5.2.2.2 Binarization

In the implemented test system, the global binarization threshold is calculated using Otsu's method (Otsu, 1979), which was found to give satisfactory results. However, other local or global methods such as the methods discussed in Sahoo *et al.* (1988); Pal and Pal (1993); Trier and Taxt (1995); Garain *et al.* (2006) can be used if the input documents images to the system are unevenly illuminated or otherwise suffer from brightness intensity of the characters and/or paper not being uniform. Table 5.1 in Section 5.1.3 provides a quantitative measurement of the effects of degradation on binarization. Table 5.1 shows the optimal threshold value and the separation effectiveness η^* as defined in Equations (1)-(15) of Otsu (1979) for a document image in five levels of degradation, starting at the ideal document image (zero print-scans) and up to four repeating accumulated print-scan cycles. It can be observed that both values decrease steadily as the degradation levels increase. In addition, to give a visual illustration of binarization, Figures 5.7 (a) and 5.6 (b) show the binarization output using Otsu's method on a complete scaled down grayscale document.

Figure 5.7: *Illustration of the effects of binarization using Otsu's method. (a) Entire grayscale document scaled down before binarization. The colors represent different grayscale values. (b) The same document after binarization.*



Former Prime Minister Khaleda Zia, also chief of Bangladesh Nationalist Party, earlier this month asked party members to take to the streets demanding the resignation of the government.

The elections need to be held under an independent caretaker government... the elections will be rigged if it's held under any political government, said Zia, the arch rival of present Prime Minister Sheikh Hasina.

She said the government had failed to run the country properly and criticized the government foreign policy saying the ruling Awami League government had become subservient towards India.

Police said initially they had used clubs to disperse the protesters, but as mobs turned violent attacking policemen, they fired bullets and tear gas shells.

The Bangladesh Nationalist Party claimed several hundreds of its activists were injured and many more were detained in different parts of the country.

The ruling party itself is creating anarchy in the country by using police to foil the opposition's peaceful democratic program, BNP Secretary General Mirza Fakhrul Islam Alamgir said.

Former Prime Minister Khaleda Zia, also chief of Bangladesh Nationalist Party, earlier this month asked party members to take to the streets demanding the resignation of the government.

The elections need to be held under an independent caretaker government... the elections will be rigged if it's held under any political government, said Zia, the arch rival of present Prime Minister Sheikh Hasina.

She said the government had failed to run the country properly and criticized the government foreign policy saying the ruling Awami League government had become subservient towards India.

Police said initially they had used clubs to disperse the protesters, but as mobs turned violent attacking policemen, they fired bullets and tear gas shells.

The Bangladesh Nationalist Party claimed several hundreds of its activists were injured and many more were detained in different parts of the country.

The ruling party itself is creating anarchy in the country by using police to foil the opposition's peaceful democratic program, BNP Secretary General Mirza Fakhrul Islam Alamgir said.

5.2.2.3 Extraction of Connected Components

Extraction of connected components (CCs) from the binarized document image is performed by connected components analysis according to the algorithm outlined in Haralick and Shapiro (1993). The accuracy of connected components analysis for the segmentation is closely dependent on the level of noise in the document. Table 5.2 in Section 5.1.4 shows the results of a measurement of the number of joined characters occurrences and the mean chain length for the same document in different levels of corruption. It has been observed that the occurrence of joined characters increases substantially with each additional print-scan cycle.

After the CCs are extracted, a filtering process based on the area of the CCs is carried where CCs that are smaller than n pixels are discarded. This is performed to remove to remove small noise artifacts which are occasionally created after binarization. In general, if this step is skipped the proposed framework's accuracy does not tend to suffer, however in heavily degraded documents filtering of very small CCs helps to speed up calculations. In the test implementation $n = 5$ was used in practice and satisfactory results were obtained.

5.2.2.4 Morphological Closing

Prior to classification, the morphological closing operation (morphological dilation followed by erosion) is applied to all CCs using a disk shaped structuring element with a radius of two pixels (Milan *et al.*, 2007). The morphological closing operation has been observed in experiments performed as part of this thesis to slightly increase the DCT features based classification accuracy in noisy documents without any observed negative side effects and therefore was applied to all CCs prior to classification.

In order to measure the effects of adding post binarization morphological closing on classification accuracy, an experiment similar to Experiment 1-B Section 4.4.2.2 was performed. This measurement was identical to Experiment 1-B Section 4.4.2.2 except for the following differences: *(i)* this experiment was performed only for the DCT-SVM pair and; *(ii)* morphological closing was performed on the binarized characters with a 2 pixel radius disk shaped structuring element. As a result of applying post binarization morphological closing, the measured error rate was reduced from 0.025 (see Table 4.2 Section 4.4.2.2) to 0.0206.

5.2.3 Affiliation of Connected Components with DDM Characters

The verification of a document image requires comparing the type and location of the perceived characters in the document image with the expected document contents stored in the DDM. One such way of doing so is the generation of a mapping between the two sets. Therefore it follows that a text document image is considered to be verified if and only if a satisfactory bijection exists from the set of symbols perceived in the document image to the set of expected characters in the digital document (as encoded in the DDM) so that the following conditions are maintained:

- The relative or absolute locations of the perceived characters in the document image are in agreement with the locations of their paired characters in the digital document.
- The relative or absolute sizes of the perceived characters in the document image are in agreement with the locations of their paired characters in the digital document.
- The classes of the perceived characters in the document image are in agreement with the classes of their paired characters in the digital document.

The score for a particular connected component denoted as CC to be affiliated with a particular character in the digital document is given by Equation (5.1).

$$p(affiliated) = \frac{NIP}{NCP} * S \tag{5.1}$$

Where NIP is the number of pixels in the intersection of the bounding boxes of CC and the character in the DDM, NCP is the number of pixels contained in the bounding box of the character in the DDM, and $S = 0$ if CC is suspected of being a joined CC comprised of two or more characters (detection of joined characters is discussed in Section 5.2.5.1) or $S = 1$ otherwise. The DDM character which scores the highest positive score is then affiliated to the CC. By letting $S = 0$, a score of zero is calculated for CCs which are suspected of being joined characters since it is of interest to build a bijection that includes only fully segmented characters.

It follows from Equation (5.1) that the score for non-intersecting DDM characters and CCs of being affiliated is zero, so that a maximum affiliation distance threshold between a CC and a DDM character is implicitly defined.

Therefore, to perform the pairing efficiently it is sufficient to perform the score calculation only for characters in the DDM whose bounding boxes intersect with the CC. An efficient search for the intersecting characters can be performed using a quad-tree or by intersecting of the output sets given by two interval trees, one for each axis.

The method of mapping by a greedy maximum shared area requires accurate registration between the scanned document image and the ideal document image to work well. It is not practical to expect the registration between the two images to be pixel level accurate. For this reason the method performs well since it allows for some degree of error in the registration between the two images while at the same time requiring a high level of relative fidelity of the scanned document image to the ideal document image. If the two images are inconsistent with one another, an exact matching between all of the CCs in the scanned document image to the characters in the DDM would fail with a very high probability.

The affiliation between the two sets is updated as necessary when the set of CCs is updated due to the segmentation process. For example, when a CC is separated into two or more smaller CCs, as in the case of segmenting joined characters, the old segmented CC affiliation is deleted, and the new segmented CCs are affiliated with their corresponding pairs if any adequate characters are found in the DDM.

5.2.4 Classification

To classify the CCs extracted from the document, an adaptive hierarchical linear kernel SVM/template matching classifier is used. First an adaptive font-specific SVM is applied, then common mismatches due to the perceptually similar symbols are disambiguated using template matching. The different stages and concepts related to the adaptive hierarchical classifier are explained in more detail in the following Sections.

5.2.4.1 Adaptive Classification

Many OCR algorithms suggested in the literature use a single classifier model approach where a single set of features and classification technique are used for recognizing every test observation (Park *et al.*, 2000; Mori *et al.*, 1995; Nagy, 1992; Mantas, 1986). The single model approach becomes problematic in multi-font and/or multilingual scenarios when the number of classes increases substantially as it leads to lower accuracy and higher computational complexity which may also translate to longer classification time per symbol (Kahan *et al.*, 1987; Öztürk *et al.*, 2001; Nagy, 1992). The problem of decreased accuracy due to the increase in number of classes is of a particular interest in this thesis since achieving sufficient performance for a verification system heavily relies on accurate classification. To address this issue different techniques have been suggested where an adaptive classifier is used to dynamically reduce the number of overall classes to choose from. One such way is attempting to recognize the font and/or language and then use the appropriate model to perform classification of the single recognized font (Öztürk *et al.*, 2001). Another suggested method is by using a varying scale hierarchical adaptive classifier that with each iteration increases the image resolution in areas of interest and reduces the number of classes to consider (Park *et al.*, 2000). Bootstrap methods that use the content of the document itself to improve classification has also been explored (Nagy, 1992; Smith, 2007).

The classifier used in this framework is an adaptive classifier that can dynamically select its decision model as required. The motivation of this approach is that by decreasing the number of overall possible output classes to choose from, higher accuracy and lower computational complexity (faster running time) can be achieved. Since the framework has prior knowledge about the document available in the DDM it can be used to adapt the classifier thus reducing the total number of candidate classes.

In the proposed method, prior to classifying a CC, the classifier checks if the CC is affiliated with any particular character in the DDM. If the CC is affiliated then the classifier switches to the relevant model for the particular font type which is expected to be found according to the DDM (if it is not the active model at that point already). Since the different models for each font type are computed once during the training stage and are stored on the verifying machine, the action of switching between different models is relatively fast as no new model calculations are required. Moreover, text documents usually feature a small number of font types, and do not switch frequently between the different font types. Therefore, active switching of models by the classifier does not impose a significant overhead in common text documents. Alternatively, if the CC is not affiliated with any character in the DDM, then the classifier uses the model representing the neighboring characters' font type or the most frequent font in the document if there are no immediate neighbors (or if the neighbors are of different font types). To perform this process efficiently, the CCs may be grouped according to their expected font type before every classification cycle thus minimizing the number of overall model replacements.

5.2.4.2 Features, Training Data and SVM Configuration

The adaptive linear kernel SVM classifier is trained with the synthetic data generated by the Print-Scan degradation model defined in Baird (1995) and uses the DCT shape description features as defined in Sections 4.3.2 and 4.4.1.2. The training set is comprised of one ideal plus 300 synthetically generated 300 DPI rasterized images per character class, and the DCT coefficients which are used as features are the 8×8 low order coefficients as concluded to be most effective in Section 4.5.

The SVM classifier is configured with the following options:

- Trained using SMO algorithm (Platt, 1998).
- Multi-class classification solved using pair wise classification (Krebel, 1999).
- Standard inner product kernel (linear kernel).
- All components are standard score normalized, $z_i = (x_i - \mu)/\sigma$, where z_i is the normalized component, x_i is the raw non normalized component, μ is the mean value of the component and σ is the standard deviation of the component.

In the implementation of the system used for experiments the SVM classifier implemented by Witten and Frank (1999) was used.

5.2.4.3 Second Tier Classification by Template Matching

Perceptual similarity between different character classes typically occur due to lowercase/uppercase character's similarity and may even occur between two "unrelated" classes as in the case of lowercase el - 'l' and the digit one '1' in the Times New Roman font (see Section 4.4.2.2 and Figures 4.8 and 4.9). In most cases the differences between these perceptually similar characters are very subtle to the extent that in some cases these classes are hardly separable even by humans if no context information is given. In this framework, the character image dimensions are normalized prior to the DCT shape features calculation and moreover do not consider any context information for generating features. Therefore, the DCT features calculated for these perceptually similar characters are very close and these classes are often misclassified with their similar counterparts.

To solve this problem a list of perceptually similar class sets such as such as {'X', 'x'} and {'P', 'p'} is defined and stored for each font. Now consider the event that according to the DDM a particular CC (herein after denoted as CC) is expected to be classified as a certain class $C_x \in G_e$ where G_e is a particular set of perceptually similar classes. However it is instead classified by the SVM classifier as any of the other similarly perceptual classes in its group $C_y \in G_e$ $x \neq y$. In such cases it is reasonable to assume that confusion has occurred due to the similarity between the classes, thus a second tier classification by template matching is performed in order to improve the distinction between the perceptually similar classes.

The template matching classification is performed by calculating the normalized cross-correlation of the misclassified CC (test pattern) with the ideal binary templates h_p representing respectively all $C_p \in G_e$ perceptually similar classes in that particular group which is given by Equation (5.2).

$$R_p(a,b) = \frac{1}{M_a * N_a + M_b * N_b - 1} \sum_{m=0}^{M_a-1} \sum_{n=0}^{N_a-1} \frac{(CC(m,n) - \overline{CC})(h_p(y+a, x+b) - \overline{h_p}}{\sigma_{cc}\sigma_{hp}} \tag{5.2}$$

where $p = 1, 2, \ldots, |G_e|$, CC is $M_a \times N_a$ pixels, h_p is $M_b \times N_b$ pixels, $0 \leq a \leq M_a + M_b - 1$, $0 \leq b \leq N_a + N_b - 1$, $\overline{CC}$ and $\overline{h_p}$ are the mean of CC and h_p respectively, σ_{cc} and σ_{hp} are the standard deviation of CC and h_p respectively.

Following the calculation of all $\{R_p(a,b) \mid p = 1, 2, \ldots, |G_e|\}$, the maximum achieved normalized cross-correlation scalar values between the CC and all other templates $\{h_p\}$ is determined for each template $\{R_p(a,b) \mid p = 1,2,\ldots,|G_e|\}$ which is denoted as $RMAX_p$ where $RMAX_p = \max(R_p)$ for all $\{R_p\}$.

Next, to assist in reducing marginal confusions, the difference d between the two maximum valued $\{RMAX_p\}$, $d = RMAX_{max} - RMAX_{max2}$ is calculated, where $RMAX_{max} = \arg\max(\{RMAX_p\})$ is the largest valued $\{RMAX_p\}$, and $RMAX_{max2} = \arg\max(\{RMAX_p\} \backslash \{RMAX_{max}\})$ is the second largest valued $\{RMAX_p\}$.

Two possible scenarios are distinguished depending on the value of d. Consider the case where $d = RMAX_{max} - RMAX_{max2}$ is a small number. A small d value indicates that at least two ideal binary templates were found to be closely matched to the classified pattern CC in the group and therefore the results could be within the range of statistical error which is caused due to the Print-Scan Channel degradation. On the other hand, if d is sufficiently large it is an indication that the best matching template has substantially more resemblance to the test pattern CC than to all other templates in the perceptually similar group G_e. In the test implementation the value of 0.08 was selected for d based on empirical results.

In addition to d, another important indication is the value of $RMAX_{max}$ in absolute terms. Higher values of $RMAX_{max}$ indicate higher similarity between the test pattern CC and the best matching template, and lower values of $RMAX_{max}$ indicate the opposite.

Therefore, the result of the template matching is considered in light of four parameters, the classification results of the SVM, the expected class in the DDM, the values of d and the value of $RMAX_{max}$. To reach the final conclusion the following logic rules are applied:

- Suppose that the DDM indicates a particular lowercase character is expected, however the SVM classifier has classified the CC as its uppercase perceptually similar counterpart. Then provided that both d and $RMAX_{max}$ are sufficiently large, and the results of the normalized cross-correlation show there is higher cross-correlation with the uppercase character variant, then the decision of the SVM classifier is overruled and the CC is marked as the uppercase version of the character. The exact same logic is applied in the opposite case where according to the DDM a particular uppercase character is expected, however the SVM classifier classifies the CC as its lowercase perceptually similar counterpart.

- For all other cases which are not expected to be a lowercase-uppercase confusion, the following logic is applied. If both d and $RMAX_{max}$ are sufficiently large the results of the cross-correlation overrule the classification, if d is not sufficiently large then the SVM output is taken as the final classification.

5.2.5 Segmentation

As discussed in Section 2.5, most OCR systems work reasonably well in mildly noisy modern documents images. However, OCR in low-quality or degraded documents images is more error prone due to the high frequency of broken and joined characters occurrence as discussed in Section 5.1.

Both Vinicius *et al.* (2007) and Ming *et al.* (2007) who suggested frameworks for verification of document images have not explicitly addressed the segmentation problem. Vinicius *et al.* (2007) suggested using a general OCR engine in order to segment and classify the characters and/or words in the documents images, therefore he implicitly makes the assumption that such general OCR systems can solve this issue with sufficient accuracy as required by a verification system. On the other hand, Ming *et al.* (2007) chose to ignore the issue completely. However, as previously discussed in Sections 5.1.3 and 5.1.4, joined and broken characters are common in document images, and the frequency of such occurrences rises very quickly when the levels of degradation increases. Therefore, it is reasonable to assume that the system proposed by Ming *et al.* (2007) could not possibly achieve reasonable results on the degraded documents studied in this thesis.

Achieving error-free segmentation with no prior information about the document's contents is still an unresolved problem. However, since the proposed system has the advantage that it can obtain information about the documents' contents it is suggested to utilize this knowledge for the purpose of high segmentation accuracy. Nevertheless, it is important to remember that any such usage must still rely in principle on the contents of the document image, so that if any changes are in fact introduced to the document, either by means of natural degradations or malicious tampering, they would still be detected by the system.

As explained in Chapter 3, the suggested verification framework extracts and stores information about the location and font type of the expected characters in the document (which is referred to DDM). The extraction of the DDM is performed at the last authoring stage, once the document is finalized and ready for print.

To solve the problem of broken and joined characters a multi-phase method is suggested in which both the classifier output and prior knowledge of document contents are considered in the different decision making stages, which is described in detail in the following

Sections.

5.2.5.1 Detection of Joined Characters

The aim of this procedure is to detect and flag all CCs which are suspected to be comprised of joined characters and estimate how many and which characters (complete or partial) each suspected joined characters CC is comprised of. To check whether a particular CC is suspected to be comprised of a number of joined characters, a search is performed in the DDM for the ideal characters in the ideal digital document whose bounding boxes intersect with the bounding box of the respective CC. An efficient search of the DDM for the intersecting characters can be performed using a quad-tree or by intersecting of the output sets given by two interval trees, one for each axis.

The existence of intersection between the CC's bounding box and multiple DDM characters bounding boxes is not sufficient by itself as the only criteria. The reason is that the Print-Scan induced global transformations in the document image which are estimated and corrected by the methods presented in Section 3.3.2 are not corrected to pixel accuracy. As the level of degradation grows due to multiple print-scan cycles, the accuracy of transformations estimation will worsen due to the repeated accumulated transformations that occur in each print-scan cycle. Therefore even after correction of estimated transformations, the bounding boxes may still be spatially shifted a small number of pixels in any direction independently for each character relative to their ideal expected location. Therefore, the intersection with multiple DDM bounding boxes is not a sufficient criterion on its own for flagging a CC as suspected of containing joined characters.

To solve this problem, for each character in the DDM whose bounding box is found to intersect the CC's bounding box, the vertical (rows) intersection ratio (denoted as VIR) and the horizontal (columns) intersection ratio (denoted as HIR) are calculated as given by Equations (5.3) and (5.4) .

$$VIR = \frac{NIR}{\min\{T, character\_height\})} \tag{5.3}$$

$$HIR = \frac{NIC}{\min\{T, character\_width\})} \tag{5.4}$$

Where NIR and NIC are the number of intersecting rows and columns respectively between the current intersected character's bounding box and the CC's bounding box. T is a value selected per font type which guarantees a minimum rate of growth for VIR and HIR as functions of NIR and NIC respectively. In other words T increases the sensitivity of Equations (5.3) and (5.4) to wide and/or tall characters.

If both VIR and HIR exceed the vertical and horizontal thresholds r_v and r_h respectively, the character is added to the chain of joined characters.

To illustrate the need for T, consider the Times New Roman 12 point font 'M' and 'i' characters in Figure 5.8, taken from a 300 dpi scan.

The width of 'M' is 43 pixels, and the width of 'i' is only 8 pixels. Therefore a horizontal overlap of the CC's bounding box and the DDM character of 4 pixels would evaluate to 0.5 in 'i' and to 0.093 in 'M' if only the ratio $NIC/character\_width$ was considered. Therefore the inclusion of T as a maximum value for the denominator compensates for the decrease in the value of Equation (5.4) in wider characters. For example, in the case of 'M', with $T = 15$, Equation (5.4) evaluates to 0.266 instead of 0.093.

In Figure 5.9 (a), a CC comprised of the two joined characters 'n' and 't' is shown. This CC was taken from one of the test documents after the estimation and correction of global transformations. To the right of it in Figure 5.9 (b), the word "month" is displayed with the ideal DDM bounding boxes drawn superimposed. It can be observed that the left most pixels of 'n' penetrate its left neighbor's 'o' expected location. However, since the penetration is marginal in the horizontal direction, Equation (5.4) would not exceed the threshold r_h and therefore only 'n' and 't' will be selected as the characters which comprise the CC on the left of the Figure.

Figure 5.8: *Examples of 400% magnified binarized 'M' and 'i' characters in the Times New Roman 12 point font. The characters were extracted from real 300 DPI document image.*

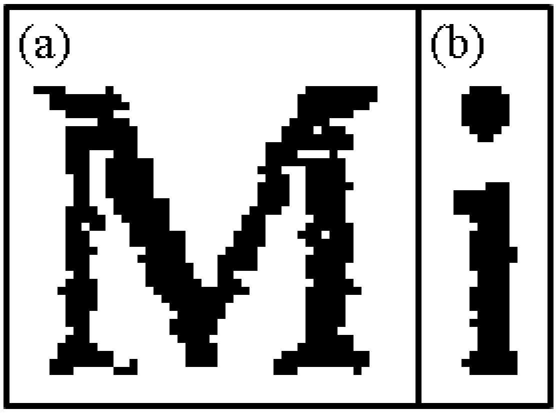
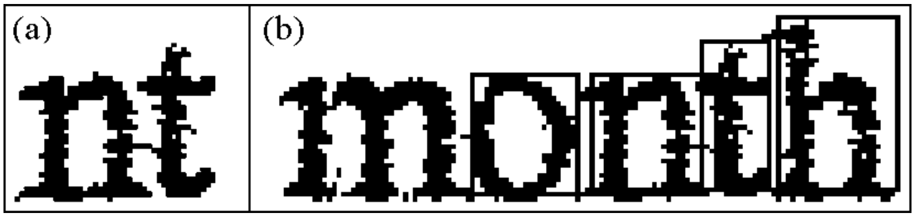


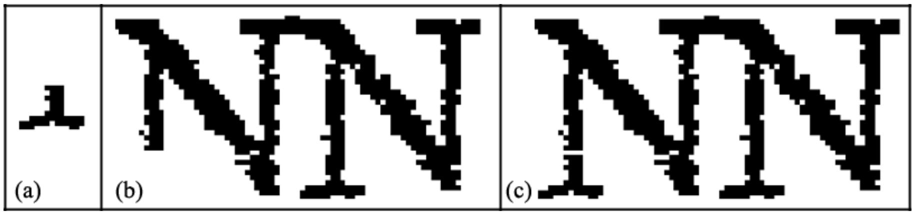
Figure 5.9: *(a) A connected component comprised of the two joined characters 'n' and 't'. (b) The word "month" with the superimposed ideal DDM bounding boxes.*



5.2.5.2 Joining of Multi-Part and Broken Characters

Whilst the previous Section described a method to detect joined characters, a different issue is handling characters that are split. The aim of this procedure is to join all multi-part and broken characters occurrences in the document by using a single method that fits all different scenarios. The suggested approach joins parts of multi-part or broken characters even if it results in the formation or extension of joined characters chains. This approach is taken since as discussed later in Section 5.2.5.3, a method is suggested to perform highly accurate segmentation of joined characters chains. Therefore, it is preferable to have a minimum number of multipart and broken characters to remain disjoined at the end of this procedure, despite the possible trade-off of more joined characters chains and/or longer joined characters chains in average.

Figure 5.10: *(a), (b) Two connected components that are determined to be joined. (c) The compound connected component resulting from the join.*



The DCT shape feature does not suffer from the limitation of classifying multipart shapes since the input to the DCT features are binary images which can contain any number of disjoint connected components. To simplify the discussion, the term “compound CC” is now defined to indicate a union of two or more disjoint CCs, maintaining the relative spatial relationships between them. In the implementation of the framework, a compound CC is represented by a binary image and the bounding box of all CCs in the transforms corrected scanned document image. In Figures 5.10 (a) and 5.10 (b), example of two connected components that are needed to be joined is given. In Figure 5.10 (c) the compound CC resulting from the join is presented.

To check whether a particular CC is suspected to be a part of a DDM character, a search is performed in the DDM for the ideal characters in the ideal digital document whose bounding boxes intersect with the bounding box of the respective CC. An efficient search of the DDM for the intersecting characters can be performed either by using a quad-tree or by intersection of the output sets given by two interval trees, one for each axis. This search is similar to the search performed in Section 5.2.5.1 and could be done using the same mechanism, however it is performed for the purpose of joining of multi-part and broken characters.

For the same reasons as explained in Section 5.2.5.1 due to accumulated transformation and degradation artifacts, the existence of intersection between the CC’s bounding box and the DDM character bounding boxes is not sufficient by itself as the only criteria for deciding whether to join a CC to other CCs which are considered part of the same character.

For example consider Figure 5.10 (b), the word “month” taken from one of the test documents after the estimation and correction of global transformations is displayed with the ideal DDM bounding boxes drawn superimposed. It can be observed that the left most pixels of ‘n’ penetrate its left neighbor’s ‘o’ expected location as defined by its ideal bounding box. Since the overlap is marginal in the horizontal direction, if existence of intersection is the only criteria, the characters ‘o’ and ‘n’ would have been joined.

Clearly the scenario of joining connected components where the overlap of their neighbors’ bounding boxes is marginal should be avoided. To address this issue, whenever a CC’s bounding box is found to intersect with a character in the DDM, a score is calculated as given by Equation (5.5). The score is based on the ratio of actual intersection between the CC’s bounding box and the character’s bounding box, and the suspected number of joined characters of CC is comprised of (if applicable).

$$J = \frac{A}{\min(\{B, C\})} \tag{5.5}$$

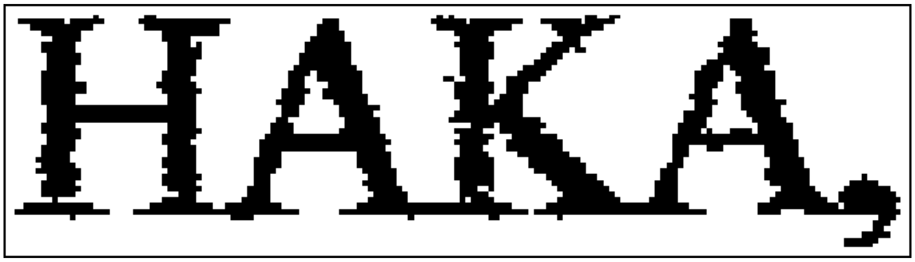
Where A is the number of intersecting pixels between the CC’s bounding box and the intersecting character’s bounding box. B is the number of pixels in the CC’s bounding box given by multiplication of the CC’s bounding boxes dimensions. C is the number of pixels in the intersecting character’s bounding box given by multiplication of the intersecting character’s bounding boxes dimensions. The value C is introduced to the score calculation to account for the scenario where the CC is both part of a broken character and in addition also a chain of joined characters such as the example given in Figure 5.10 (b).

If the value of J is sufficiently large and exceeds a predetermined threshold, the CC is joined with all other CCs that were determined to belong to the same character to create a new compound CC. Following is the deletion of contributing separate CCs and the bounding box of the newly created compound CC is updated based on the bounding boxes of the joined CCs. In the test implementation the value of 0.75 was used as the joining threshold.

5.2.5.3 Separation of Joined Characters

Precise segmentation of the scanned text document image to the atomic characters that it is comprised of is both compulsory and highly significant for classification accuracy since the classifier requires complete character images as inputs. In order to segment a CC which is suspected to be a chain of joined characters to its single complete characters components the following procedure is performed.

Figure 5.11: *Connected component comprised of five joined characters.*



In Section 5.2.5.1 a method is described to analyze the document and flag all CCs which are suspected of being comprised of joined characters. In addition, for each CC which is suspected of being such, the characters in the digital document that are suspected to be part of the joined chain for that particular CC are noted.

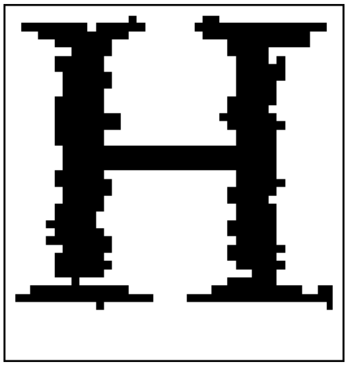
Suppose a particular CC extracted from the document is the target of segmentation (hereinafter denoted as CC). Let $L_{cc} = \{C_i \mid i = 1 \ldots k\}$ be defined as the set of characters in the digital document that are suspected to be part of the joined characters chain comprising CC, where k is the number of suspected characters in the chain and C_i are arranged from left to right as i grows.

Before describing the segmentation process the reader is advised to refer to Figure 5.16 early as it illustrates the segmentation process and therefore may contribute to clarify the remainder of this Section.

The segmentation process is performed by repeating the following procedure for all $C_i \in L_{cc}$. Begin by localizing the best match for the template h_i in the "left-most subsection" of CC, where h_i is the M rows by N columns ideal binary template representing the i-th joined character $C_i \in L_{cc}$. h_i is defined so that the value one corresponds to the foreground pixels while the value zero corresponds to background pixels. The "left-most subsection" is defined as the area starting at the first (left-most) column of CC and ending *(i)* at the end of CC, or *(ii)* after p columns where p is typically slightly larger than N (in the test implementation $p = N + 4$ was used).

In Figure 5.11 a CC comprised of a chain of five joined characters is presented. The first "left-most subsection" extracted from the CC is shown in Figure 5.12. As observed in Figure 5.12, the extracted subsection cover also a small part of the character 'A' to

Figure 5.12: *The first "left-most subsection" extracted from the connected component in Figure 5.11.*



the right, however after the following steps are performed the final segmented image is sufficiently accurate as later experimental results will show.

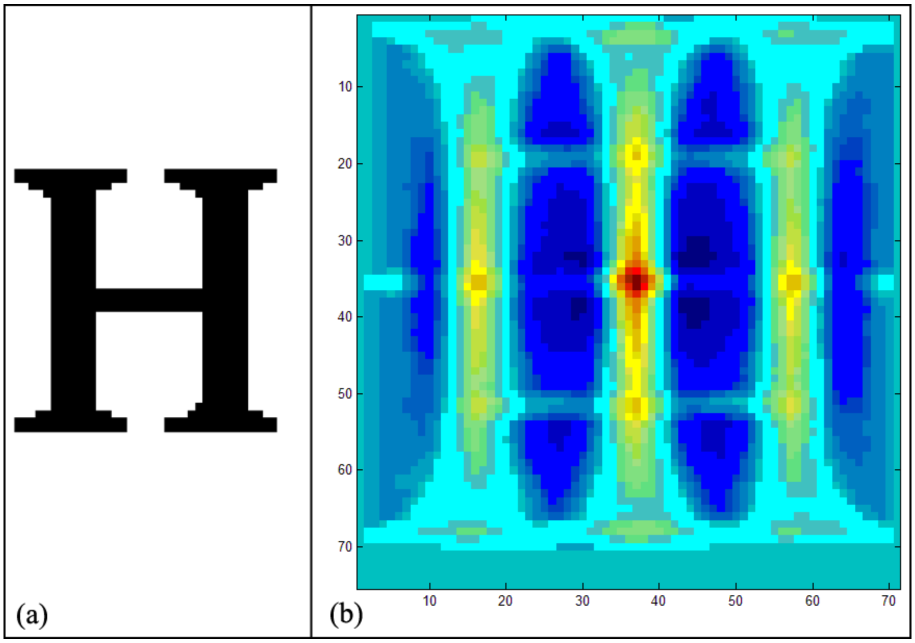
The best match localization for the template h_i in the "left-most subsection" of CC is calculated using the normalized cross-correlation (Equation (5.2)) of the pattern and template (illustrated in Figure 5.13).

Once the best match between the "left-most subsection" of the CC and the template h_i is determined, a rectangle in exactly the same dimensions of the template h_i is cropped from the CC covering the best match location. In the event that CC is not large enough to allow an exact crop, the cropped rectangle is padded with zeroes where required so it has the same dimensions as h_i. In Figure 5.15 (a) the best match rectangle cropped from the CC is presented.

Consider the scenario where some overlapping is expected between the preceding character C_{i-1} and/or the succeeding character C_{i+1} with the currently segmented character C_i. This situation is illustrated in Figure 5.14 featuring binary 400% enlarged Times New Roman characters with the expected bounding boxes taken from an ideal document image.

To take this scenario into account when performing segmentation the number of intersecting columns between the bounding boxes of the previous character C_{i-1} and the current

Figure 5.13: *(a) Graphic representation of the ideal template of the character 'H'. (b) Illustration of the results of the normalized cross-correlation between the first "left-most subsection" shown in Figure 5.12 and the ideal template shown in (a). A strong peak (dark red colour) in the cross-correlation results is observed about the middle area.*



character C_i in the ideal digital document image is determined from the information stored in the DDM, which is denoted as x_{prev}. In addition, the number of intersecting columns between the bounding boxes of the current character C_i and the next character C_{i+1} in the ideal digital document image is determined from the information stored in the DDM, which is denoted as x_{next}.

To finalize this iteration of segmentation all of the pixels in h_i are assigned the value 1 except for the x_{prev} left-most columns (if x_{prev} is positive) and the rightmost x_{next} columns (if x_{next} is positive). Figure 5.15 (c) gives an example of a mask where there is no intersection with any of the left or right neighboring characters, as a result all of the ideal template's pixels values are set to one (representing logical true).

Finally the logical AND operation is applied pixel wise between the "left-most subsection" of the CC and h_i. These steps are intended to guarantee that what is segmented of the CC is the actual contents of the document image and if and only if some overlap between

Figure 5.14: *Overlapping of ideal bounding boxes due to the shapes and placement of characters in the ideal document.*

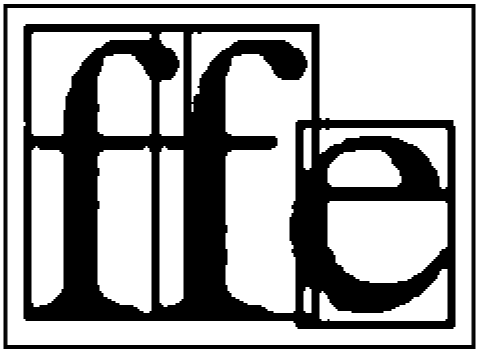
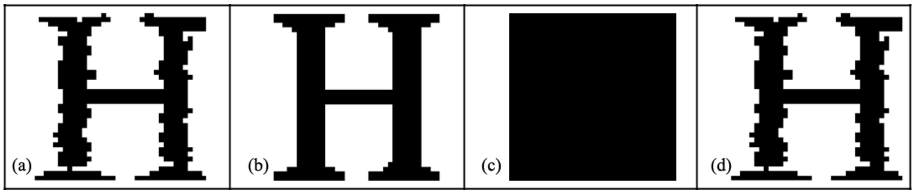


Figure 5.15: *(a) The best match (according to the results of the cross-correlation) rectangle cropped from the connected component. (b) A graphic representation of the ideal template of the 'H' character. (c) The logical mask used for segmenting the character 'H', in this instance all the pixels are set to logical true. (d) The results of the pixel-wise logical AND operation between the ideal template and the connected component.*

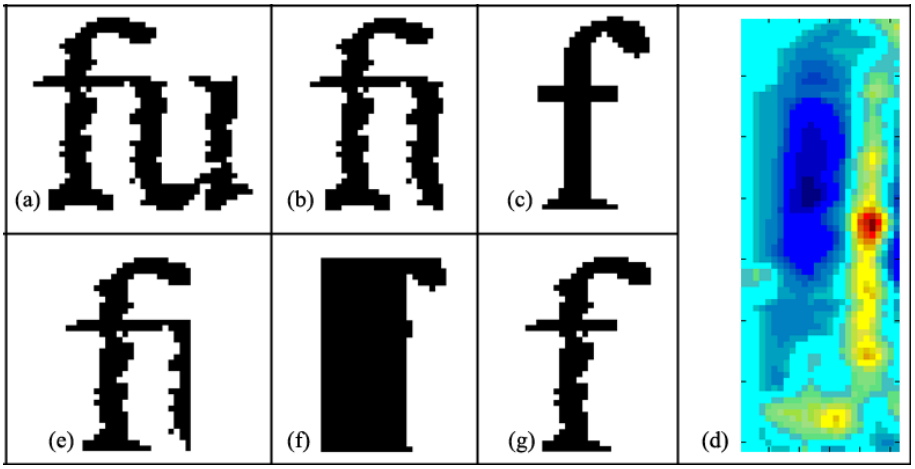


two adjacent characters is expected then the ideal template h_i is used only at the section of actual overlap to help segment the two or more overlapping joined characters. The final segmentation of the character C_i can be observed in Figure 5.15 (d).

The final step before repeating the process for segmenting character C_{i+1} is the removal of all left-most columns of the CC that were: *(i)* the $N - x_{next}$ left-most segmented columns to the character C_i *(ii)* all columns that are positioned left to any column that was segmented to belong to the character C_i.

Another example of a segmentation process of a character that has a shared area with its right neighbor is given in Figure 5.16. In Figure 5.16 (a) the CC comprised of two joined characters is presented. In Figure 5.16 (b) the "left-most subsection" extracted from the CC is shown. Figure 5.16 (c) shows the ideal noiseless template representing the expected character 'f'. In Figure 5.16 (d) the results of the normalized cross-correlation between the "left-most subsection" and the ideal noiseless template are graphically illustrated where

Figure 5.16: *Illustration of the different stages of the segmentation of two joined characters. (a) Connected component comprised of two joined characters. (b) The "left-most subsection" extracted from the connected component. (c) Ideal template of the character 'f'. (d) Illustration of the results of the normalized cross-correlation between the first "left-most subsection" shown in (b) and the ideal template shown in (c). (e) Refined bounding box estimate and crop based on template matching results. (f) Illustration of the logical AND mask. (g) Final segmentation obtained by performing the logical pixel-wise AND operation between the image in (e) and the mask in (f).*



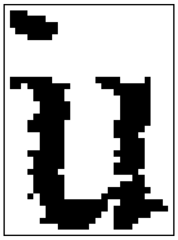
the peak is represented by the dark red area. Figure 5.16 (e) displays the rectangle cropped from the CC covering the best match location in exactly the same dimensions of the ideal template of 'f'. Figure 5.16 (f) shows the ideal template after all of its left-most columns which are not shared by the neighboring character as expected by the DDM are transformed to foreground value (numerically 1 or equivalently logically true). Finally Figure 5.16 (g) displays the segmented character after applying the pixel wise logical and operation between the images in Figure 5.16 (e) and Figure 5.16 (f).

In Figure 5.17 the remainder and yet unsegmented part of the CC is presented after removal of the $N - x_{next}$ left-most already segmented columns.

To complete the segmentation for the entire CC the same segmentation process is repeated for the entire CC until all of the characters which were suspected to comprise the joined character chain are segmented, or alternatively until the CC is completely processed.

As the experimental results in the next Sections show, this process is very effective in providing accurate segmentation when the presented document image for verification is true to the contents of the digital document as encoded in the DDM. Additionally, in the case where the document's content is different than what is expected (e.g. due to

Figure 5.17: *The remainder and yet unsegmented part of the connected component segmented in Figure 5.16 after segmentation of the character 'f'.*



tampering), there is a very high probability that the segmentation would result in meaningless symbols thus increasing the chance of classification error and therefore the successful detection of tempering.

5.2.5.4 Further Processing of Rejected Symbols

The processes described in the preceding Sections of segmentation and classification of the symbols perceived in the document results in the agreement or disagreement of the segmentation and classification results with the expected contents of the document as encoded in the DDM. In the case of such disagreement, further logic is applied through local binarization, segmentation, noise removal and reclassification to those particular symbols that have failed the verification process so far. This method aims at improving segmentation and classification results for "problematic" symbols by utilizing prior knowledge of the document's content encoded in the DDM and by applying additional and different low level operations. However, the emphasis remains on relying solely on the symbols appearing in the scanned document image.

In the case where a CC (possibly compound) has failed to comply with the expected characters in the DDM, variants on the basic approach are performed to handle different types of noise that often occurs. The first variant is to binarize the CC using a local threshold in order to search for a match under conditions where the local characters are poorly scanned. The second variant is to apply median filtering (Milan *et al.*, 2007) to the CC before classifying, in an attempt to repair the deformations caused by the print-scan process.

Local Threshold The CC (possibly compound) is regenerated by cropping the relevant sub-image (based on the bounding box of the rejected CC) from the input grayscale document image and then calculating the optimal threshold for the cropped sub image method using Otsu's method. This is equivalent to performing local binarization in the area of the CC when the threshold calculation is performed using Otsu's method (Otsu, 1979).

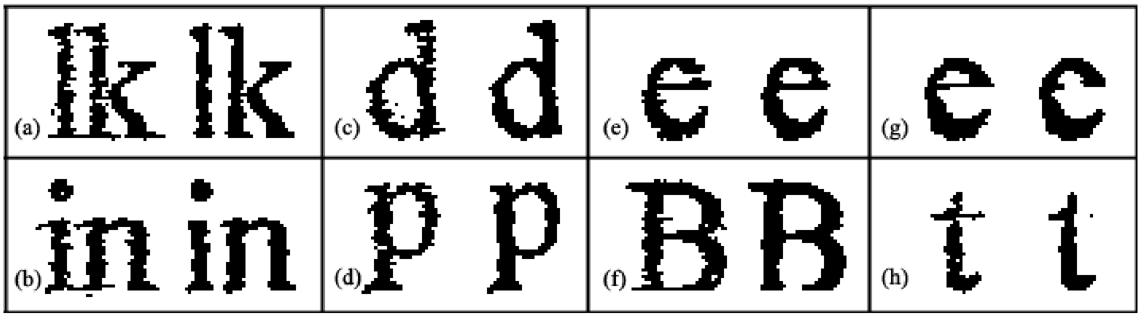
Next segmentation and classification is repeated for the newly extracted CC. If the results agree with the expected character in the DDM the mapping between the CCs and the DDM is updated and no further action is taken. However, if the classification results still do not agree with the DDM then further processing is applied to the CC as follows.

Median Filtering Again the process of cropping the relevant sub-image from the input grayscale document image and then calculating the optimal threshold for the cropped sub image method using Otsu's method is repeated.

Next a 3×3 two-dimensional median filtering (Milan *et al.*, 2007) is applied to the newly extracted CC (possibly compound) to remove noise. From experimental results performed as part of this thesis, the 3×3 two-dimensional median filtering has been observed in the majority of cases to result in excellent removal of many of the typical deformations caused by the Print-Scan Channel noise. In Figure 5.18 a number of examples are given for characters before (left hand) and after (right hand) median filtering. Examples of the typical effect of median filtering on noisy characters is shown in Figures 5.18 (a) to 5.18 (e). However in some cases, as can be observed in Figures 5.18 (f) to 5.18 (h) median filtering result in higher degradation, typically causing the disappearance of thin strokes. This is a serious problem which in some scenarios actually changes the perceived meaning of a character such in the case of the example shown in Figure 5.18 (g) where the character 'e' was transformed into the character 'c'.

Following the median filtering, segmentation and classification is repeated for the newly extracted CC. If the results agree with the expected character in the DDM the mapping between the CCs and the DDM is updated and no further action is taken. However, if the classification results still do not agree with the DDM then no further processing is applied to the CC, resulting in the CC marked as conflicting with the expected content of the document.

Figure 5.18: *Examples of the effects on median filtering on print-scan degraded characters. Left hand characters are before median filtering and right hand characters are after median filtering.*



5.3 Evaluation

Ideally, a comparative baseline for the accuracy achieved by the suggested verification framework would be a similar commercial or literature proposed verification systems such as the system proposed in Vinicius *et al.* (2007) and in Ming *et al.* (2007).

In consideration of the discussion in Section 2.4.2, the system suggested in Ming *et al.* (2007) has three significant shortcomings:

- A substantial security flaw which allows an attacker to replace all of the occurrences of an existing character in the document with another non existing character without being detected.
- The system suggested in Ming *et al.* (2007) ignores joining of characters, a phenomenon typical to highly degraded text documents as discussed in Section 5.1.4. Ignoring joining of characters due to Print-Scan noise would undoubtedly result in a rapid and substantial decrease in the accuracy of the proposed verification system as the level of noise increases.
- The system suggested in Ming *et al.* (2007) utilizes contour-based Fourier descriptors as its main shape descriptors. Therefore, the maximal theoretical accuracy achieved by this system is closely related to the accuracy of the suggested descriptors. As discussed in Section 4.4.2.5, the contour-based Fourier descriptors are unsuitable for performing accurate character classification in noisy text documents.

Therefore, in consideration of the arguments above, it seems that the system proposed in Ming *et al.* (2007) is unsuitable for accurate verification of noisy document images and therefore would not provide an adequate comparative baseline.

On the other hand, as described in Section 2.4.1, Vinicius *et al.* (2007) suggested a content-based verification method that uses a commercial OCR engine to perform segmentation and classification of the characters in the verified document. Therefore the theoretical accuracy limit of the verification system proposed in Vinicius *et al.* (2007) is bounded by the accuracy of the OCR engine used. Hence, comparing the results of the system proposed in this thesis against a selected OCR system is in fact equivalent to evaluation of the proposed system against the maximal theoretical accuracy of the system proposed in Vinicius *et al.* (2007).

Consequently, it was decided to compare the results obtained by an implementation of the framework suggested in this thesis against the accuracy of Tesseract, Google's state-of-the-art OCR system (Google, 2012b).

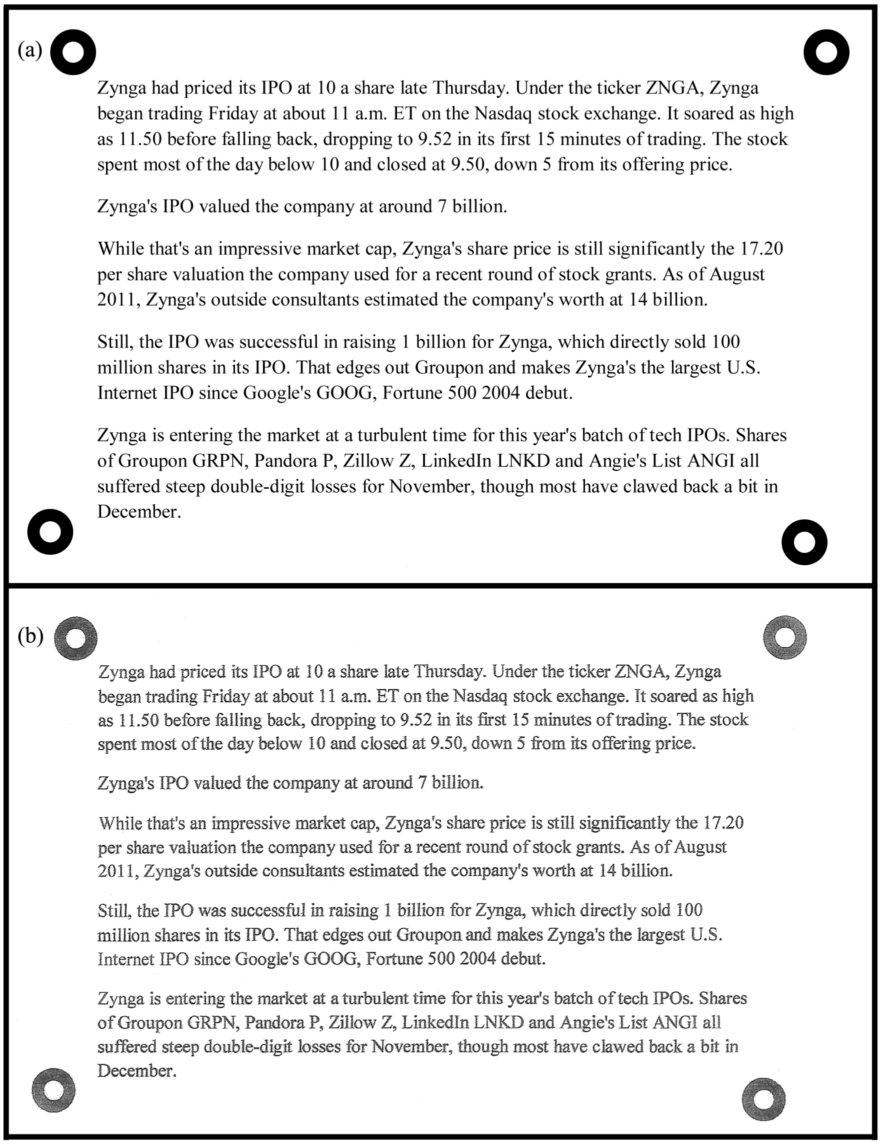
5.3.1 Test Data

Since the suggested framework requires both the digital file in PDF format and the scanned images of the same documents to perform verification, no public document images database was found suitable to be used for evaluation of the suggested framework. Therefore, it was required to self-produce the test documents, and the documents images scans.

The documents used for the evaluation in all of the following tests are self-produced PDF text documents that contain the entire English language alphanumeric character set and the period, comma, apostrophe and dash punctuation marks, a total of 66 character classes. The test documents' contents are based on short paragraphs copied from various news articles. Therefore, the occurrence frequency of the different test character classes is not uniform, but is closely related to real world distribution.

Given that this thesis aims to handle verification under conditions where the document image is significantly degraded from printing and scanning noise, the test documents were printed and scanned repetitively up to three times in order to simulate three increasing levels of Print-Scan degradation. However, unlike Chapter 4, every print-scan cycle output was tested, providing an indication of the tested systems' accuracy when processing document images in different levels of degradation.

Figure 5.19: *A scaled down example of one of the test documents used for the experiments in Section 5.3. (a) The ideal noiseless document. (b) Same document image after three print-scan cycles.*



(a)

Zynga had priced its IPO at 10 a share late Thursday. Under the ticker ZNGA, Zynga began trading Friday at about 11 a.m. ET on the Nasdaq stock exchange. It soared as high as 11.50 before falling back, dropping to 9.52 in its first 15 minutes of trading. The stock spent most of the day below 10 and closed at 9.50, down 5 from its offering price.

Zynga's IPO valued the company at around 7 billion.

While that's an impressive market cap, Zynga's share price is still significantly the 17.20 per share valuation the company used for a recent round of stock grants. As of August 2011, Zynga's outside consultants estimated the company's worth at 14 billion.

Still, the IPO was successful in raising 1 billion for Zynga, which directly sold 100 million shares in its IPO. That edges out Groupon and makes Zynga's the largest U.S. Internet IPO since Google's GOOG, Fortune 500 2004 debut.

Zynga is entering the market at a turbulent time for this year's batch of tech IPOs. Shares of Groupon GRPN, Pandora P, Zillow Z, LinkedIn LNKD and Angie's List ANGI all suffered steep double-digit losses for November, though most have clawed back a bit in December.

(b)

Zynga had priced its IPO at 10 a share late Thursday. Under the ticker ZNGA, Zynga began trading Friday at about 11 a.m. ET on the Nasdaq stock exchange. It soared as high as 11.50 before falling back, dropping to 9.52 in its first 15 minutes of trading. The stock spent most of the day below 10 and closed at 9.50, down 5 from its offering price.

Zynga's IPO valued the company at around 7 billion.

While that's an impressive market cap, Zynga's share price is still significantly the 17.20 per share valuation the company used for a recent round of stock grants. As of August 2011, Zynga's outside consultants estimated the company's worth at 14 billion.

Still, the IPO was successful in raising 1 billion for Zynga, which directly sold 100 million shares in its IPO. That edges out Groupon and makes Zynga's the largest U.S. Internet IPO since Google's GOOG, Fortune 500 2004 debut.

Zynga is entering the market at a turbulent time for this year's batch of tech IPOs. Shares of Groupon GRPN, Pandora P, Zillow Z, LinkedIn LNKD and Angie's List ANGI all suffered steep double-digit losses for November, though most have clawed back a bit in December.

The documents were printed and scanned using a HP 7500A model standard inkjet consumer all in one printer and scanner on standard general use office paper. The printing was done using the default standard settings and the scanning was done using 300 DPI grayscale settings. The evaluation was performed on all document images from the three different levels of print-scan degradation. In Figures 5.19 (a) and (b) a scaled example of one of the test documents is given in its ideal noiseless form as well as after three print-scan cycles.

5.3.2 Experiments

Three experiments are performed for evaluating the accuracy of the suggested verification framework.

Experiment 1 aims to evaluate the false rejection (false positives) rate for documents featuring a single font type. The false positives rate is evaluated for the verification system and is compared against the recognition error rate of a state-of-the-art OCR system on the same documents as a baseline for comparison.

Experiment 2 aims at evaluating the accuracy of the proposed framework in detecting inconsistencies present in the document images with the digital document's contents, or in other words the true positives accuracy in detecting tampering by the proposed framework.

Experiment 3 aims to evaluate the false rejection (false positives) rate for documents featuring multiple font types. This experiment is performed to obtain a measure of the effect on the accuracy when the number of fonts increases.

5.3.2.1 Experiment 1

In this experiment, the false rejection (false positives) rate of the verification system was evaluated for documents featuring a single font type - Times New Roman 12 point. The experiment was performed on 20 different documents where each of the documents was analyzed in three different levels of degradation generated by repetitively printing and scanning the document images up to three times. Therefore the total number of images verified by the system was 60, representing 20 different documents in three different increasing levels of Print-Scan Channel degradation containing a total of 61,284 characters. To obtain a baseline for comparison, the same document images were processed for recognition by Tesseract, Google's state-of-the-art OCR system (Google, 2012b), and the number of errors in the recognized output document was recorded.

It is important to note, that the proposed framework also verifies the location of the perceived characters against their expected locations as encoded in the DDM. Therefore, in the case of poor registration of the scanned input document image with the ideal document image the system may reject some characters even if the recognition was accurate. This is not the case with Tesseract OCR which is only evaluated for its recognition accuracy. Therefore this experiment is slightly biased in favour of Tesseract OCR.

Table 5.3: *Results observed in experiment 1 for the proposed framework.*

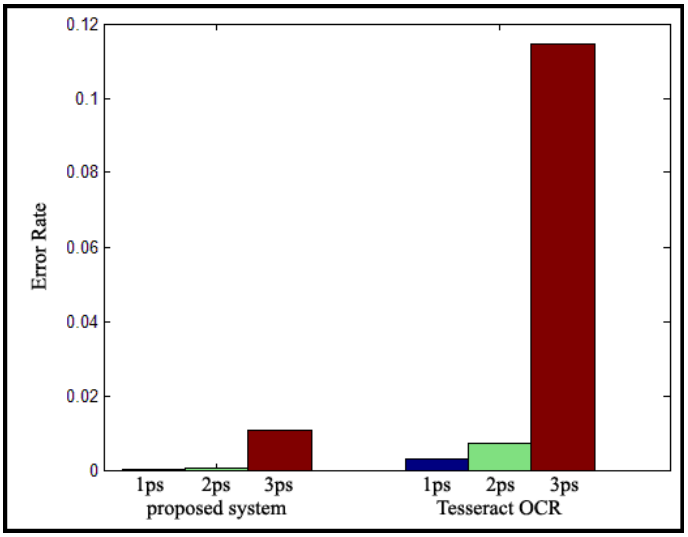
Degradation Level	Proposed Err Rate	OCR Err Rate	Proposed Docs No Err	OCR Docs No Err	Performance Index
1 print-scan	0.00014	0.00308	17 (85%)	3 (15%)	17-2-1
2 print-scan	0.00063	0.00729	9 (45%)	0 (0%)	20-0-0
3 print-scan	0.01076	0.11445	0 (0%)	0 (0%)	20-0-0
All docs	0.00385	0.04160	26 (43.3%)	3 (5%)	57-2-1

Results The results observed in experiment 1 for the proposed framework are summarized in Table 5.3 as follows. The Proposed Err Rate is the proposed system's average false rejection rates defined as (*rejections/total characters*) are given. OCR Err Rate is the average error rate obtained by Tesseract OCR across all documents calculated as (*recognition errors/total characters*). Proposed Docs No Err is the absolute number and percentage of documents that were verified successfully by the system without any false rejection occurring. OCR Docs No Err shows the number and percentage of documents that were processed by Tesseract OCR without any recognition errors. Performance Index aims at summarizing which system has outperformed in the per document comparison level. The Performance Index comparison is given in the x-y-z format where x represents the number of documents in which the suggested framework has outperformed Tesseract OCR in terms of errors per document, y represents the number of draws between the two systems, and z represents the number of times in which the Tesseract OCR system outperformed the suggested framework in terms of errors per document. The three top rows of the Table represent the different levels of corruption in increasing order and the bottom row of the Table represents the overall average values across all tested document images.

A graphical representation of the results summarized in Table 5.3 is given in Figure 5.20. Figure 5.20 shows clearly the effect of degradation on classification accuracy, with both systems failing more frequently with higher degradation. However, in comparison to the suggested framework, Tesseract basically collapses at the highest level of degradation achieving on average just over one error per 10 characters.

In addition to the results summarized in Table 5.3 it is interesting to note the following statistical observations. The worst result obtained by the system for a particular document was 29 false positives out of 1245 characters or equivalently a 0.025 rejection rate. On the other hand, the worst result obtained by Tesseract OCR for a single document was 213 errors out of 1117 characters or equivalently a 0.1907 error rate. Both results were

Figure 5.20: *A graphical representation of experiment 1 results. The error rates of the proposed system and of the Tesseract OCR system are shown grouped by system. The blue bars represent the level of degradation achieved after one print-scan cycle, the green bars represent the level of degradation accumulated after two repeating print-scan cycles, and the maroon bars represent the level of degradation accumulated after three repeating print-scan cycles.*



achieved for different documents images that were printed and scanned three repetitive times.

Furthermore, there were only three documents images that obtained results worse than 0.02 by the proposed system, all three are documents that were printed and scanned three repetitive times. Tesseract OCR on the other hand, has obtained results worse than 0.02 for 20 out of the 60 document images, 19 of these were documents that were printed and scanned three times, and one document which was printed and scanned two times.

The smallest absolute performance ratio between the system and Tesseract OCR for a particular document image was three false positives by the system compared to 19 recognition errors by Tesseract which is equivalent to a ratio of 6.33 times the error rate in favour of the proposed framework. The largest absolute performance ratio between the system and Tesseract OCR for a particular document image was two false positives by the system compared to 79 recognition errors by Tesseract which is equivalent to a ratio of 39.5 times the error rate in favour of the proposed framework. Both aforementioned document images were printed and scanned repetitively three times therefore containing the highest levels of noise tested.

Table 5.4: *Additional statistical information on experiment 1 results.*

Measurement	System	OCR
Best result 2 print-scan	0 (9 docs)	0.0031
Worst result 2 print-scan	0.0019	0.026
Best result 3 print-scan	0.0022 (2 docs)	0.0196
Worst result 3 print-scan	0.025	0.1907
Variance of results for three print-scan	0.000049	0.00262
Variance of results for all docs	0.000038	0.003524

Some more statistical information on experiment 1 results is given in Table 5.4. The test results reveal that the proposed framework consistently outperforms the Tesseract OCR system. Furthermore as the levels of Print-Scan Channel noise in the tested documents images increases, the accuracy of the OCR system drops much more quickly than the accuracy drop observed in the proposed system, and the latter outperforms Tesseract in every case. This is demonstrated both by the overall higher accuracy and by the substantially lower variance in accuracy obtained by the proposed system.

As discussed in previous sections of this Chapter, the large decrease observed in OCR performance is very likely due to both classification errors and the substantial increase in occurrence of broken characters and of joined characters chains which raises the difficulty and complexity of achieving accurate segmentation.

Further inspection of the rejections reported by the system shows that most false positives were due to confusion between the characters ‘l’ and ‘I’ in the most severely degraded of images. As the level of degradation grows these two characters become more and more alike and therefore are more likely to be confused in the classification stage. Since the two classes are very much similar to one another this is indeed a hard problem to solve in noisy text documents without context information. Furthermore, this shows that the overall performance of the system in highly degraded document images is in the general case very accurate and stable when the classes are reasonably separate from another in terms of perceptual similarity. The observed accuracy achieved by the proposed system implies that the segmentation and classification methods both are sufficiently robust and accurate to be useful for practical implementation of a robust verification system.

5.3.2.2 Experiment 2

Experiment 2 aims at evaluating the accuracy of the proposed framework in detecting inconsistencies present in the document images with the digital document's contents, or in other words the true positives accuracy in detecting tampering by the proposed framework. This experiment is not applicable to OCR therefore this experiment was not evaluated on any other system as for a comparative baseline.

When performing this evaluation, the focus is shifted from evaluating accuracy in the context of complete documents (e.g. measuring rejection rate of tampered documents) towards focusing on evaluating the accuracy in detecting different categories and types of tampering. The reason is that the rejection rate of tampered documents depends also on the number, types, and scope of the changes made to the document. Documents featuring a single change have higher probability of escaping detection over documents featuring a larger number of changes. Similarly, changes of smaller scope are also harder to detect than changes of larger scope. Moreover, as discussed later in this Section, some types of modifications are harder to detect than others, and furthermore a small number of modifications are found to exist that the system failed to detect. Therefore, the measurement of successful rejection rate for complete documents does not reveal much about the system's performance.

Establishing this, to evaluate the accuracy of the proposed framework, this experiment defines several categories of modifications and measures the accuracy of the framework in detecting changes in these particular categories across different levels of print-scan degradation. It is important to note that since later scans are based on earlier scans, failures to detect specific modifications (false negatives) will tend to accumulate. Any undetected modification in an early scan will likely continue to produce the same results in subsequent scans.

Modification Categories

Extended Scope Modifications Modifications under this category include the following:

- Replacement of a number of adjacent characters or a word with characters that are not perceptually similar, but without causing a substantial shift of the subsequent characters. For example “boat” changed into “shop”.

- Any modification that results in substantial shift of the subsequent characters in the same and/or following lines. For example modifying the word “SHOPPING” to “SHIPPING” may result in the second ‘I’ in the word to be sufficiently displaced so it would end completely out of its expected location.
- The addition or deletion of an entire word or more.

Small Scope Prominent Modifications Modifications under this category include modifications that do not generate a substantial shift of the following characters, hence the changes to the document are local or in other words having a small scope:

- The replacement of a character or two adjacent characters with characters that are not perceptually similar, but without causing a substantial shift of the following characters. For example “made” changed into “make”.
- The deletion or addition of one or two adjacent characters, but without causing a substantial shift of the following characters if any.

Small Scope Subtle Modifications Modifications under this category include the replacement of a character or two adjacent characters with characters that are perceptually similar, but without causing a substantial shift of the following characters. Examples of some of the modifications evaluated as part of this experiment are given in Table 5.5.

For the purpose of conducting the experiment, 274 modifications were performed in nine documents containing a total of 12,071 characters so that almost every line in the documents contained at least one modification. Each of the documents was analyzed in three different levels of degradation generated by repetitively printing and scanning the document images up to three times. Therefore the verified document images contained a total number of 822 modifications embedded in a total of 36,213 characters.

Table 5.5: *Examples of modifications in the small scope subtle category evaluated in experiment 2.*

Changed From	Changed To	Changed From	Changed To
first	flrst	Zynga	7ynga
10	1o	list	lisf
beef	beet	Have	Haue
comments	connnemts	will	wili
suck	svck	now	how
85	8S	November	Novcmber
It's	lt's	creator	creat0r
right	righl	Rosenblatt	Rosenhlatt
Italy's	Italv's	Romano	Komano
distinguished	distinguisbed	DHAKA	DIIAKA
elections	elecfions	demonstrations	demonsttations
recent	reeent	IPO	1PO
million	mlllion	2009	20O9
something	scmething	started.	started,
perfect	perrect	expense	expenSe
SUV	SVV	churned	Churned
government	govemment	rigged	ripped
common	coininon	FORTUNE	FORTVNE

Results The results observed in experiment 2 are summarized in Table 5.6. The data in the cells of Table 5.6 are formatted in the X/Y form where X represents the number of detected modifications in the category and Y represents the total number of changes in the category..

The results show that all extended scope modifications were detected regardless of the level of degradation present in the document images.

The small scope prominent modifications were all detected except for a replacement of the character ‘U’ with the characters “II”. This occurred probably due to the way the two characters were joined and segmented. These results were consistent across all three levels of degradation.

The absolute majority of all small scope subtle modifications were detected in all three

Table 5.6: *Results observed in experiment 2 for the proposed framework in verifying different modification categories.*

Degradation Level	Extended Scope	Small Scope Prominent	Small Scope Subtle
1 print-scan	39/39	79/80	138/155
2 print-scan	39/39	79/80	137/155
3 print-scan	39/39	79/80	131/155

levels of degradation. However there was a small decrease in accuracy observed as the level of degradation increased. Analysis of the modification detection failures reveal that the failures can be divided into two main categories.

The first category applies to changes that were usually detected but were occasionally missed. The modifications in this category include mostly changes in the case of a character such as ‘s’ changed into ‘S’, ‘w’ changed into ‘W’ and ‘v’ changed into ‘V’. Also contained in this category is a single failure of detecting “ti” modified into “fi” and a single failure of detecting “fr” changed into “ft”, modifications that are both very subtle in terms of perceptual similarity in noisy documents.

The second category contains modifications that were repeatedly and consistently undetected by the system. In particular the system consistently failed to detect changes in words containing the following modifications:

- ‘m’ changed into “rn”, such as in the case of “government” changed into “govemment”.
- “mm” changed into “inin”, such as in the case of “common” changed into “coininon”.
- “mm” changed into “nnn”, such as in the case of “comments” changed into “connnents”.
- ‘H’ changed into “II”, such as in the case of “DHAKA” changed into “DIIAKA”.

In the case of ‘m’ changed into “rn”, the joining of the two characters results in a new shape which is very similar to the original character and therefore the failure to detect this type of modification is expected.

In the case of “mm” changed into “inin” the system verified the lower parts of the ‘i’ coupled with the ‘n’ as the character ‘m’, however the top part of the ‘i’ (the dot) was detected as not belonging to the document (an addition).

The two other cases fail only in the event where the characters are joined. The resulting shapes from the segmentation stage are similar enough to the expected characters so that the classifier recognizes these as the shapes that are expected and therefore the modifications are wrongly verified.

In all other subtle modifications cases tested, the system exhibited stable and consistent rejection rates across all levels of degradation tested.

5.3.2.3 Experiment 3

In this experiment, the false positives rate of the proposed verification system was evaluated for documents containing multiple font types. Content wise (characters), the documents used in this experiment are a subset of the documents used in experiment 1, however the documents were edited to be rendered in a mixture of the following font types - Times New Roman 12 point, Arial 10 point and Boopee 11 point (see Figure 5.21). The change in the font types used in the documents also changed the layout and position of the text in the documents (in comparison to experiment 1). The change in the fonts was performed in varied places such as beginning of paragraph, beginning of sentence, per words, in the middle of words, and per character.

The font type Boopee 11 point was chosen in addition to the Arial and Times New Roman fonts due to its characteristics and the difficulties imposed on the system:

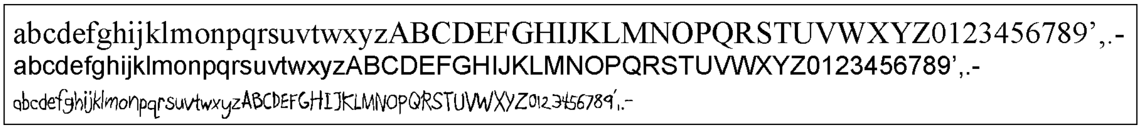
- Boopee 11 point font is different to the other two fonts and appears to be a somewhat randomly shaped cursive like script.
- Boopee 11 point font is relatively small which makes binarization and classification more sensitive to noise.
- It is very thin which makes it more fragile and results in a larger number of broken characters with higher level of erosion and therefore larger spatial separation between the parts in comparison to the other two font types as the level of degradation increases. To visually illustrate this, in Figure 5.22 character ‘l’ is displayed in magnification for the three fonts discussed here.
- The characters are relatively very close to one another and therefore the Boopee 11 point font tends to suffer more from the phenomena of joined characters as the degradation level increases.

- Since some of the characters in the font are very thin (e.g. lower case ‘l’ and ‘i’) the estimation and correction of the linear conformal transformations must be very accurate or else the algorithm would suffer reduced accuracy since the mapping between the connected components found in the document and the DDM depends on intersection of the bounding boxes in the ideal document and the scanned image document.

The experiment was performed on 10 different documents where each of the documents was analyzed in three different levels of degradation generated by repetitively printing and scanning the document images up to three times. Therefore the total number of images verified by the system was 30, representing 10 different documents in three different increasing levels of Print-Scan Channel degradation containing a total of 29,172 characters.

In Figure 5.23 an excerpt of one of the test documents after three repeating print-scans and before binarization is given to illustrate the appearance of the three fonts in high level of print-scan degradation. And in Figure 5.24 a section of the same document excerpt (before binarization) is shown in magnification. Substantial joining of characters and print-scan induced degradation of the characters is observed.

Figure 5.21: *The three fonts used in experiment 3, Times New Roman 12 point (top), Arial 10 point and Boopee 11 point (bottom). The absolute dimensions of the characters in the image are not accurate however the relative sizes and spacing are correct.*



abcdefghijklmonpqrsuvtwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789',.-
abcdefghijklmonpqrsuvtwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789',.-
abcdefghijklmonpqrsuvtwxyzABCDEFGHIJKLMNOPQRSTUVWXYZ0123456789',.-

Figure 5.22: *The character lowercase el - 'l' is displayed in magnification for the three font types: (a) Boopee 11 point, (b) Arial 10 point, (c) Times New Roman 12 point.*

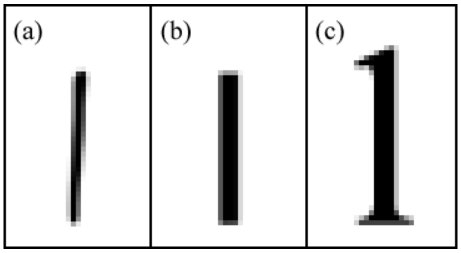
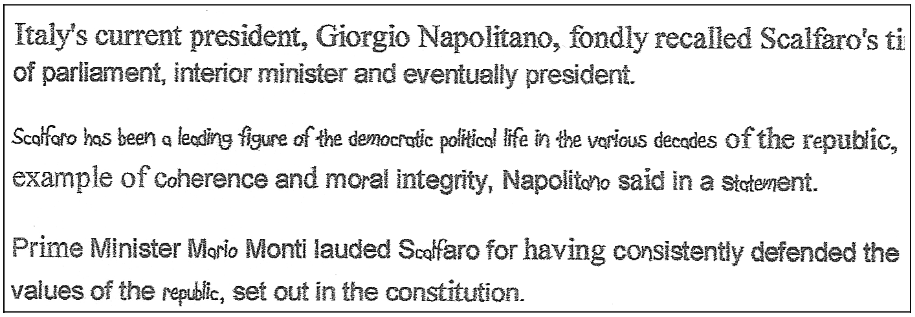


Figure 5.23: *Excerpt of one of the test documents used in experiment 3 after three repeating print-scans and before binarization.*

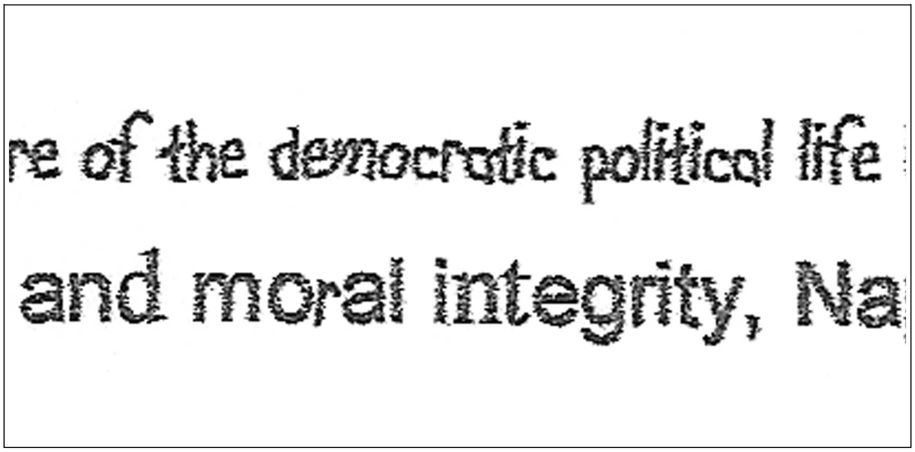


Italy's current president, Giorgio Napolitano, fondly recalled Scalfaro's ti
of parliament, interior minister and eventually president.

Scalfaro has been a leading figure of the democratic political life in the various decades of the republic,
example of coherence and moral integrity, Napolitano said in a statement.

Prime Minister Mario Monti lauded Scalfaro for having consistently defended the
values of the republic, set out in the constitution.

Figure 5.24: *Magnification of a section of the document excerpt shown in Figure 5.23.*



re of the democratic political life

and moral integrity, Na

Results The results observed in experiment 3 for the proposed framework are summarized in Table 5.7 as follows. Column A shows the system's average false rejection rates defined as (*rejections/total characters*). Column B represents the absolute number and percentage of documents that were verified successfully without any false rejection occurring. Column C shows the respective error rates change in **experiment 1** for the same document subset. The three top rows of the Table represent the different levels of corruption in increasing order and the bottom row of the Table represents the overall average values across all tested document images.

The test results reveal that while achieving good accuracy the proposed framework has displayed a drop in the accuracy in comparison to experiment 1. For one print-scan documents the decrease in accuracy was by a factor of 6.93, for two print-scan documents the decrease in accuracy was by a factor of 13.2 and for three print-scan documents the decrease in accuracy was by a factor of 7.89, all in comparison to experiment 1 results. Despite testing multi-font documents the results achieved by the proposed framework in this experiment are still significantly better than the results obtained by Tesseract OCR in experiment 1 for single type documents.

Further inspection of the rejections reported by the system in this experiment show that in documents that were printed and scanned once the rejections are distributed between the fonts as follows: Boopee - 66.67%, Arial - 25% and Times New Roman - 8.33%. Moreover, the errors occur exclusively in the characters 'i' - 62.5% and 'l' - 37.5%. The errors occur mainly due to shifting of the relative locations of the bounding boxes and the relative thinness of these characters that affects their segmentation. This result is interesting in

Table 5.7: *The results observed in experiment 3 for the proposed framework.*

	A	B	C
1 print-scan	0.0014	3 (30%)	0.00021
2 print-scan	0.0067	1 (10%)	0.00051
3 print-scan	0.0412	0 (0%)	0.00523
All docs	0.0164	4 (13.3%)	0.001984

light of the expectation that the classification of other characters in the Boopee 11 point font would results in errors since the increased difficulty inherent to the font as it is much smaller, thinner, and significantly spatially less spaced then the other two fonts.

In documents that were printed and scanned more than once the Boopee 11 point font has contributed substantially more to errors than the two other fonts. As the level of degradation increases, the characters in the Boopee font are much more likely to join and break than the two other fonts. When combining the higher frequency of broken and joined characters with increased shifts in the bounding boxes due to the global and local transformations, the segmentation task becomes much more difficult and prone to errors.

To summarize the results of experiment 3, the verification of characters in the Times New Roman 12 point font remained comparatively the same in comparison to documents featuring a single font. Characters in the Boopee 11 point and Arial 10 point generated more false rejections (false positives) mainly in the 'l' and 'i' characters as they are not easily distinguishable due to lack of serif in these fonts. The overall recognition of Boopee 11 point was very good in documents scanned and printed up to two times despite the additional challenges due to its thin strokes being particularly vulnerable to degradation. All in all, the results show good robustness to scanning degradation, exceeding 99% for all but document images generated after three print-scan cycles.

Additional research is required to improve accuracy further when faced with higher level degradation. One such potential approach is first estimating the level of degradation in order to further fine-tune the adaptive classifier's models. Another approach is improving the flexibility of the bounding boxes mapping in conjunction with the DDM data when performing segmentation as many of them verification errors in the 'l' and 'i' characters were caused due to the relative shift of the characters in respect to the expected locations in the ideal document images.

5.4 Summary

This Chapter discussed the typical characteristic phenomenon occurring in degraded document images and presented the varied methods suggested for overcoming the difficulties that arise in verifying noisy documents images. Finally, three experiments were performed to evaluate the suggested approach in practice and the results of the experiments were presented and discussed.

The experiments show that the proposed approach achieves substantially higher accuracy than observed by a state-of-the-art OCR (Tesseract) system to which it was compared against, in particular in noisy print-scan degraded document images. Moreover, the experiments show that the system is capable to detect tampering and other changes in text documents with high accuracy and sensitivity to subtle changes.

Chapter 6

Conclusion

This Chapter contains two main sections. The first is a summary of the conclusions made as part of the work on this thesis, while the second Section highlights some of interesting questions that rose during this work but presently still remained unanswered. These Sections combined, suggest to the reader some of the possible future research possibilities for yet improving the accuracy of automated text document verification frameworks.

6.1 Concluding Remarks

Fraudulent documents cause significant financial overhead and impose security breaches to civil and government organizations. While many robust methods have been suggested for the verification of digital documents such as digital signatures and digital watermarks, authentication of printed documents still relies on physical security mechanisms. Unfortunately, existing approaches to authentication of digital documents do not adapt well to printed documents. The geometric transformations and added noise that are part of the printing and scanning process ultimately do not allow the authentication of printed and scanned documents by pure mathematical comparison to the digital document. Consequently, the content perceived in the document image is the only reasonable basis that may be used for automated verification of Print-Scan degraded documents.

In this thesis a method for robust content-based verification of print-scan degraded documents using standard off-the-shelf printing and scanning devices was investigated. The objective of the system is to analyze printed and scanned documents (possibly multiple times) and verify that they have not been tampered with in comparison to the original digital documents. The Document Analysis Component discussed in Chapter 3 analyzes the original digital documents stored in the popular PDF format and extracts descriptive information about the document contents (referred to as "DDM", see Section 3.1) that is stored and used later in the document image verification stage.

The DDM is used as a baseline to which the perceived content and its placement in the document image is compared against. Moreover, since the system performs verification rather than only recognition, it is not feasible to use n-gram statistics or dictionaries to infer knowledge about the verified document thus improving recognition rates. In the absence of such mechanisms for improving recognition in highly degraded document images, an alternative method is applied. By relying on prior document knowledge encoded in the DDM the system can segment the document images more accurately than systems that solely rely on statistical methods such as general OCR systems for example.

Accurate classification of the characters present in the documents is one of the most important steps in content-based verification of document images as they are the fundamental building blocks of text documents. In Chapter 4, the recognition accuracy of individual characters in noisy Print-Scan degraded documents was evaluated for different shape descriptors and classifiers combinations utilizing differently sized synthetically generated training sets. It was concluded that region-based DCT features give excellent accuracy with all tested classifier combinations and resilience to the substantial Print-Scan noise present in the test data. From all of the evaluated classifiers combinations, both DCT-SVM and DCT-KNN have displayed outstanding and very close accuracy with a slight advantage to DCT-SVM.

It is interesting that even when the training set was comprised only of one ideal observation per class, DCT provided reasonable recognition accuracy. Furthermore, with the inclusion of additional model-based synthetic training data generated according to the degradation model suggested in Baird (1995) a substantial decrease in the error rate was observed. Therefore, it was concluded that *(i)* the synthetic training data generated by the model reasonably imitates the degradations generated by the Print-Scan Channel, at least in terms of the feature space; *(ii)* the effect of both the real Print-Scan Channel degradations and the model-based degradations on the DCT based features are such that the features calculated from both the synthetic generated training data and from the real test samples cluster together in the feature space, but in separate to other classes. As a result the different character classes are linearly well separated in the feature space, which translates to high accuracy by both k-NN and SVM classifiers. The best overall results were achieved by the DCT-SVM when features were comprised of the 8×8 low order DCT coefficients classified using linear SVM trained with 1 ideal + 300 synthetic observations per class.

On the other hand, it was concluded, that both Geometric Affine Moment Invariants (GAMI) and the Centroid Distance Function Fourier Descriptors (CDFD) are not applicable for robust character recognition in Print-Scan degraded noisy documents. It was concluded that GAMI features are linearly inseparable in Hilbert Space and therefore generalization of the training data by a perceptron classifier such as linear SVM does not provide accurate classification. CDFD also failed to deliver good results most likely due to substantial corruption in the boundary of the characters in the presence of high levels of Print-Scan corruption.

In Chapter 5, a detailed discussion of the methods used for performing verification of noisy documents images is given. In particular, Chapter 5 discusses the problem of effective and accurate segmentation with respect to the typical characteristic phenomenon occurring in degraded document images such as the joining, morphing and breaking of characters. It is concluded, that by using the DDM and by incorporating template matching, it is possible to perform accurate segmentation in the presence of high levels of print-scan degradation, while still being sensitive to detecting fine levels of tampering of the documents.

To evaluate the effectiveness of the approach three experiments were performed for evaluating the false-positive and false-negative detection rates of the system in varying levels of print-scan degradation and for single and multi-font documents. It was concluded that the proposed approach is substantially more accurate than a state-of-the-art OCR system (Tesseract) to which it was compared against, in particular in noisy print-scan degraded document images. Moreover, it was concluded that the proposed approach is capable of detecting tampering in text documents with high accuracy and sensitivity to subtle changes.

6.2 Future Work

There are many ideas that were left out of this thesis due to time and scope constraints. In the following list, some of the many suggestions and ideas to extend this research are brought.

- Estimation of degradation levels in a particular document image would allow to further optimize the scope of the hierarchal adaptive classifier. Classifier models representing distinct levels of degradation for a particular font type may be generated in advance and loaded as required depending on the actual levels of degradation detected in the verified document image.
- Automatic consolidation of classes sharing very high perceptual similarity such as lowercase el - ‘l’ and the digit ‘1’, and automatic generation of perceptually similar classes sets based on auto generated similarity metrics. This would also enable the system to break compound classes such as ‘i’, ‘j’ and ‘:’to shared primitive building

blocks and would simplify the segmentation of the document.

- Improving classification accuracy by generation of real-time ad hoc classification models utilizing the characters present in the verified document, a specialized degradation model, and prior document knowledge. The degraded characters already verified in the document contain information about the degradation that can be expected in the reminder of the document and therefore may provide the most accurate basis for the generation of additional training samples.

- Despite giving satisfactory results in this thesis, the output of the degradation model defined in Baird (1995) is quite different to the characters obtained from real life scanned images in perceptual terms, especially as the level of degradation grows. It is this author's opinion that there is much work that can be done on improving the available degradation models such that the output would be perceptually more similar to characters extracted from real life scanned document images.

- Verification of documents containing embedded images in various levels of Print-Scan degradation. Many documents feature embedded images and other graphic elements such as logos and handwritten signatures which would require verification but have not been dealt with at all in this thesis.

- Improving upon the generation of the bijection between the document image and the DDM contents to allow for varying levels of flexibility based on different constraints (e.g. character dimensions) when mapping the two.

Bibliography

Abul Hasnat, Md.and Habib, S. M. M. and Khan, M. (2007). Segmentation free Bangla OCR using HMM: Training and recognition. In *Proceedings of the 1st Conference on Digital Communications and Computer Applications 2007*.

Adobe Systems, I. (2009). Document management - portable document format. http://wwwimages.adobe.com/www.adobe.com/content/dam/Adobe/en/devnet/pdf/pdfs/adobe_supplement_iso32000_1.pdf.

Ahmed, N., Natarajan, T., and Rao, K. R. (1974). Discrete cosine transform. *IEEE Transactions on Computers*, **23**(1), 90–93.

Alpert, N. M., Bradshaw, J. F., Kennedy, D., and Correia, J. A. (1990). The principal axes transformation - a method for image registration. *J Nucl Med*, **31**(10), 1717–1722.

Amanatiadis, A., Kaburlasos, V., Gasteratos, A., and Papadakis, S. (2009). A comparative study of invariant descriptors for shape retrieval. In *Imaging Systems and Techniques, 2009. IST '09. IEEE International Workshop on*, pages 391–394.

Amanatiadis, A., Kaburlasos, V., Gasteratos, A., and Papadakis, S. (2011). Evaluation of shape descriptors for shape-based image retrieval. *Image Processing, IET*, **5**(5), 493 –499.

Bai, X., Yang, X., Latecki, L. J., Liu, W., and Tu, Z. (2010). Learning context-sensitive shape similarity by graph transduction. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **32**(5), 861–874.

Baird, H. S. (1995). Document image defect models. In L. O'Gorman and R. Kasturi, editors, *Document image analysis*, pages 315–325. IEEE Computer Society Press, Los Alamitos, CA, USA.

Baird, H. S. (2000). The state of the art of document image degradation modeling. In *Proceedings of 4 th IAPR International Workshop on Document Analysis Systems, Rio de Janeiro*, pages 1–16.

Bellman, R. E. (1957). *Dynamic Programming.* Princeton University Press.

Bentley, J. L. (1975). Multidimensional binary search trees used for associative searching. *Communications of the ACM*, **18**(9), 509–517.

Blum, H. (1973). Biological shape and visual science. *Journal of Theoretical Biology*, **38**(2), 205–287.

Breiman, L. (2001). Random forests. *Machine Learning*, **45**(1), 5–32.

Breiman, L. and Cutler, A. (Last Accessed: Nov 2012). Random forests. http://www.stat.berkeley.edu/ breiman/RandomForests/cc_home.htm.

Buch, P. (2011). Svm max sep hyperplane with margin.png. http://en.wikipedia.org/wiki/File:Svm_max_sep_hyperplane_with_margin.png.

Burges, C. J. C. (1998). A tutorial on support vector machines for pattern recognition. *Data Min. Knowl. Discov.*, **2**(2), 121–167.

Caruana, R. and Niculescu-Mizil, A. (2006). An empirical comparison of supervised learning algorithms. In *Proceedings of the 23rd international conference on Machine learning*, ICML '06, pages 161–168, New York, NY, USA. ACM.

Caruana, R., Karampatziakis, N., and Yessenalina, A. (2008). An empirical evaluation of supervised learning in high dimensions. In *Proceedings of the 25th international conference on Machine learning*, ICML '08, pages 96–103, New York, NY, USA. ACM.

Casey, R. and Lecolinet, E. (1996). A survey of methods and strategies in character segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, **18**(7), 690–706.

Charan, S. K. (2006). A block DCT based printed character recognition system. Master's thesis, Department of Mathematics and Computer Science, Sri Sathya Sai Institute of Higher Learning, Deemed University.

Cheriet, M. and Moghaddam, R. F. (2008). Diar: Advances in degradation modeling and processing. In *ICIAR*, pages 1–10.

Coomans, D. and Massart, D. L. (1982). Alternative k-nearest neighbour rules in supervised pattern recognition: Part 1. k-nearest neighbour classification by using alternative voting rules. *Analytica Chimica Acta*, **136**, 15–27.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Mach. Learn.*, **20**(3), 273–297.

Cox, I. (2008). *Digital Watermarking and Steganography*. The Morgan Kaufmann Series in Multimedia Information and Systems Series. Elsevier Science Limited.

Customs, U. and Protection, B. (2009). Fraudulent document detection and traveler identification security measures. http://www.cbp.gov/linkhandler/cgov/newsroom/fact_sheets/travel/ frad_doc_detect.ctt/frad_doc_detect.doc.

Cybenko, G. (1989). Approximation by superpositions of a sigmoidal function. *Mathematics of Control, Signals, and Systems (MCSS)*, **2**(4), 303–314.

Cyc (2008). Svm separating hyperplanes.png. http://en.wikipedia.org/wiki/File:Svm-separating-hyperplanes.png.

Deng, M. and Dodson, C. (1994). *Paper: an engineered stochastic structure*. Tappi Press.

Devcore (2008). Dctjpeg.png. http://en.wikipedia.org/wiki/File:Dctjpeg.png.

Devijver, P. A. and Kittler, J. (1982). *Pattern recognition: A statistical approach*. Prentice Hall.

Devroye, L., Gyorfi, L., Krzyzak, A., and Lugosi, G. (1994). On the strong universal consistency of nearest neighbor regression function estimates. *The Annals of Statistics*, **22**(3), 1371–1385.

Dhanya, D. and Ramakrishnan, A. G. (2002). Optimal feature extraction for bilingual OCR. In *Proceedings of the 5th International Workshop on Document Analysis Systems V*, DAS '02, pages 25–36, London, UK, UK. Springer-Verlag.

Dietterich, T. G. and Bakiri, G. (1995). Solving multiclass learning problems via error-correcting output codes. *Journal of Artificial Intelligence Research*, **2**(1), 263–286.

Droettboom, M. (2003). Correcting broken characters in the recognition of historical printed documents. In *Proceedings of the 3rd ACM/IEEE-CS Joint Conference on Digital Libraries*, JCDL '03, pages 364–366, Washington, DC, USA. IEEE Computer Society.

Duan, K.-B. and Keerthi, S. S. (2005). Which is the best multiclass SVM method? an empirical study. In *Proceedings of the Sixth International Workshop on Multiple Classifier Systems*, pages 278–285.

Eichmann, G., Lu, C., Jankowski, M., and Tolimieri, R. (1990). Shape representation by Gabor expansion. *Proceedings of SPIE*, pages 86–94.

Eikvil, L. (1993). OCR - optical character recognition.

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

Elkan, C. (2011). Nearest neighbor classification.

Fischler, M. A. and Bolles, R. C. (1981). Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, **24**(6), 381–395.

Fletcher, T. (2009). Support vector machines explained.

Flusser, J. and Suk, T. (1993). Pattern recognition by affine moment invariants. *Pattern Recognition*, **26**(1), 167 – 174.

Flusser, J. and Suk, T. (1994). Affine moment invariants: A new tool for character recognition. *Pattern Recognition Letters*, **15**(4), 433–436.

Fournel, T., Becker, J. M., and Boutant, Y. (2007). Self-encryption for paper document authentication. *Journal of Physics: Conference Series*, **77**(1), 7–12.

Garain, U., Paquet, T., and Heutte, L. (2006). On foreground-background separation in low quality document images. *Int. J. Doc. Anal. Recognit.*, **8**(1), 47–63.

Goldwasser, S., Micali, S., and Rivest, R. L. (1988). A digital signature scheme secure against adaptive chosen-message attacks. *SIAM Journal on Computing*, **17**(2), 281–308.

Google (Last Accessed: Nov 2012a). ocropus. http://code.google.com/p/ocropus/.

Google (Last Accessed: Nov 2012b). tesseract-ocr. http://code.google.com/p/tesseract-ocr/.

Govindan, V. K. and P., S. A. (1990). Character recognition - a review. *Pattern Recognition*, **23**(7), 671–683.

Granlund, G. H. (1972). Fourier preprocessing for hand print character recognition. *IEEE Transactions on Computers*, **21**(2), 195–201.

Hamidi, M. and Pearl, J. (1976). Comparison of the cosine and Fourier transforms of Markov-1 signals. *IEEE Transactions on Acoustics, Speech and Signal Processing*, **24**(5), 428–429.

Haralick, R. and Shapiro, L. (1993). *Computer and robot vision*. Number v. 2 in Computer and Robot Vision. Addison-Wesley Pub. Co.

Hastie, T. and Tibshirani, R. (1998). Classification by pairwise coupling. In *Proceedings of the 1997 Conference on Advances in Neural Information Processing Systems*, NIPS '97, pages 507–513, Cambridge, MA, USA. MIT Press.

Haykin, S. (1999). *Neural Networks: A Comprehensive Foundation*. International edition. Prentice Hall International.

Headlessplatter (2011). Random forest model space.png. http://en.wikipedia.org/wiki/File:Random-forest-model-space.png.

Hesse, J. (1999). Counterfeiting and misuse of the social security card and state and local identity documents. Testimony before U.S. House Judiciary Committee, Subcommittee on Immigration and Claims.

Ho, T. K. (1995). Random decision forests. In *Proceedings of the Third International Conference on Document Analysis and Recognition - Volume 1*, ICDAR '95, pages 278–282, Washington, DC, USA. IEEE Computer Society.

Hosny, K. M. (2007). Exact Legendre moment computation for gray level images. *Pattern Recognition*, **40**(12), 3597–3605.

Hu, M.-K. (1962). Visual pattern recognition by moment invariants. *IRE Transactions on Information Theory*, **8**(2), 179–187.

Kahan, S., Pavlidis, T., and Baird, H. S. (1987). On the recognition of printed characters of any font and size. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **9**(2), 274–288.

Kanungo, T., M. Haralick, R., and Phillips, I. (1993). Global and local document degradation models. In *Proceedings of the International Conference on Document Analysis and Recognition*, pages 730–734.

Kanungo, T., Haralick, R. M., Stuezle, W., Baird, H. S., and Madigan, D. (2000). A statistical, nonparametric methodology for document degradation model validation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**(11), 1209–1223.

Kauppinen, H., Seppanen, T., and Pietikainen, M. (1995). An experimental comparison of autoregressive and Fourier-based descriptors in 2D shape classification. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **17**(2), 201–207.

Khan, A. and Mirza, A. M. (2007). Genetic perceptual shaping: Utilizing cover image and conceivable attack information during watermark embedding. *Information Fusion*, **8**(4), 354–365.

Khayam, S. A. (2003). The discrete cosine transform (DCT): Theory and application. Technical report, Wireless and Video Communications (WAVES) Lab at Michigan State University.

Kibriya, A. M. and Frank, E. (2007). An empirical comparison of exact nearest neighbour algorithms. In *Proceedings of the 11th European Conference on Principles and Practice of Knowledge Discovery in Databases*, PKDD 2007, pages 140–151, Berlin, Heidelberg. Springer-Verlag.

Krebel, U. H.-G. (1999). Pairwise classification and support vector machines. In B. Schölkopf, C. J. C. Burges, and A. J. Smola, editors, *Advances in kernel methods*, pages 255–268. MIT Press, Cambridge, MA, USA.

Krzyzak, A., Leung, S. Y., and Suen, C. (1988). Fourier descriptors of two dimensional shapes- reconstruction and accuracy. In *IAPR Workshop on CV, 1988, Tokyo*, pages 199–202.

Kuhl, F. P. and Giardina, C. R. (1982). Elliptic Fourier features of a closed contour. *Computer Graphics and Image Processing*, **18**(3), 236–258.

Kuklinski, T. (2004). Automated authentication of current identity documents. Technical report, 2004 IEEE Conference on Technologies for Homeland Security, Cambridge, MA, April 21-22, 2004.

Kunttu, I., Lepistö, L., Rauhamaa, J., and Visa, A. (2003). Multiscale Fourier descriptor for shape classification. In *Proceedings of the 12th International Conference on Image Analysis and Processing*, ICIAP 03, pages 536–541, Washington, DC, USA. IEEE Computer Society.

Latecki, L. J. (2005). Retrieval results for shape similarity on the MPEG-7 data set. Technical report, Dept. of Co,puter and Information Sciences, Temple University.

Latecki, L. J., Lakmper, R., and Eckhardt, U. (2000). Shape descriptors for non-rigid shapes with a single closed contour. In *Proceedings of the IEEE Conference Computer Vision and Pattern Recognition*, pages 424–429.

Li, Y. (1992). Reforming the theory of invariant moments for pattern recognition. *Pattern Recognition*, **25**(7), 723–730.

Lin, C.-S. and Hwang, C.-L. (1987). New forms of shape invariants from elliptic Fourier descriptors. *Pattern Recognition*, **20**(5), 535–545.

Lin, H.-t. and Lin, C.-J. (2003). A study on sigmoid kernels for svm and the training of non-PSD kernels by SMO-type methods. Technical report.

Ling, H. and W. Jacobs, D. (2007). Shape classification using the inner-distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **29**, 286–299.

Lins, R. D. (2009). A taxonomy for noise in images of paper documents - the physical noises. In *Proceedings of the 6th International Conference on Image Analysis and Recognition*, ICIAR '09, pages 844–854, Berlin, Heidelberg. Springer-Verlag.

Lyman, P. and Varian, H. R. (2003). How much information? Technical report, School of Information Management and Systems.

Machajdik, J. and Hanbury, A. (2010). Affective image classification using features inspired by psychology and art theory. In *Proceedings of the international conference on Multimedia*, MM '10, pages 83–92.

Maitra, S. (1979). Moment invariants. *Proceedings of the IEEE*, **67**(4), 697–699.

Mantas, J. (1986). An overview of character recognition methodologies. *Pattern Recognition*, pages 425–430.

Mao, W. (2003). *Modern Cryptography: Theory and Practice*. Prentice Hall Professional Technical Reference.

Maragos, P. (1989). Pattern spectrum and multiscale shape representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **11**(7), 701–716.

Mehta, R. and Kaur, R. (2013). Neural Network Classifier for Isolated Character Recognition. *International Journal of Application or Innovation in Engineering and Management (IJAIEM)*, **2**(2), 285–293.

Mehtre, B. M., Kankanhalli, M. S., and Lee, W. F. (1997). Shape measures for content based image retrieval: A comparison. *Information Processing and Management*, **33**(3), 319–337.

Meyer, D., Leisch, F., and Hornik, K. (2003). The support vector machine under test. *Neurocomputing*, **55**(1-2), 169–186.

Milan, S., Vaclav, H., and Roger, B. (2007). *Image Processing, Analysis, and Machine Vision*. CL-Engineering, 3rd edition.

Ming, J., Wong, E. K., and Memon, N. D. (2007). Robust document image authentication. In *ICME*, pages 1131–1134.

Moghaddam, R. F. and Cheriet, M. (2009). Low quality document image modeling and enhancement. *Int. J. Doc. Anal. Recognit.*, **11**(4), 183–201.

Mokhtarian, F. (1995). Silhouette-based isolated object recognition through curvature scale space. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **17**(5), 539–544.

Mori, S., Suen, C. Y., and Yamamoto, K. (1995). Historical review of OCR research and development. In L. O'Gorman and R. Kasturi, editors, *Document image analysis*, pages 244–273. IEEE Computer Society Press, Los Alamitos, CA, USA.

Nagy, G. (1992). At the frontiers of OCR. In *Proceedings of the IEEE*, volume 80:7, pages 1093–1100.

Nonnemaker, J. and Baird, H. S. (2009). Using synthetic data safely in classification. In *Document Recognition and Retrieval XVI*, volume 7247.

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

Öztürk, S., Sankur, B., and Abak, A. T. (2001). Font clustering and cluster identification in document images. *J. Electronic Imaging*, **10**(2), 418–430.

Pal, N. R. and Pal, S. K. (1993). A review on image segmentation techniques. *Pattern Recognition*, **26**(9), 1277–1294.

Park, J., Govindaraju, V., and Srihari, S. N. (2000). OCR in a hierarchical feature space. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **22**(4), 400–407.

Persoon, E. and Fu, K. S. (1986). Shape discrimination using Fourier descriptors. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **8**(3), 388–397.

Pezeshk, A. and Tutwiler, R. (2010). Extended character defect model for recognition of text from maps. In *Image Analysis Interpretation (SSIAI), 2010 IEEE Southwest Symposium on*, pages 85–88.

Platt, J. C. (1998). Sequential minimal optimization: A fast algorithm for training support vector machines. Technical report, Microsoft Research.

Platt, J. C., Cristianini, N., and Shawe-Taylor, J. (2000). Large margin dags for multiclass classification. In *Advances in Neural Information Processing Systems 12*, pages 547–553.

Reiss, T. H. (1991). The revised fundamental theorem of moment invariants. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **13**(8), 830–834.

Rice, S., Kanai, J., and Nartker, T. (1992). *A Report on the Accuracy of OCR Devices.* University of Nevada, Information Science Research Institute.

Rice, S., Jenkins, F., and Nartker, T. (1995). The fourth annual test of OCR accuracy. Technical report, Information Science Research Institute.

Ronhjones (2012). Gaussian training data.png. http://en.wikipedia.org/wiki/File:Gaussian-training-data.png.

Rose, H. (2009). How good can it get? analysing and improving OCR accuracy in large scale historic newspaper digitisation programs. *D-Lib Magazine*, **15**(3/4).

Sahoo, P. K., Soltani, S., Wong, A. K., and Chen, Y. C. (1988). A survey of thresholding techniques. *Comput. Vision Graph. Image Process.*, **41**(2), 233–260.

Schulenburg, J. (Last Accessed: Nov 2012). GOCR. http://jocr.sourceforge.net/.

Segal, M. R. (2004). Machine learning benchmarks and random forest regression. Technical report, Center for Bioinformatics and Molecular Biostatistics, UC San Francisco.

Sellen, A. J. and Harper, R. H. (2003). *The Myth of the Paperless Office*. MIT Press, Cambridge, MA, USA.

Sima, C. and Dougherty, E. R. (2008). The peaking phenomenon in the presence of feature-selection. *Pattern Recogn. Lett.*, **29**(11), 1667–1674.

Smith, R. (2007). An overview of the Tesseract OCR engine. In *Document Analysis and Recognition, 2007. ICDAR 2007. Ninth International Conference on*, volume 2, pages 629–633.

Stevenj (2009). DCT-symmetries.svg. http://en.wikipedia.org/wiki/File:DCT-symmetries.svg.

Taxt, T., Ólafsdóttir, J. B., and Daehlen, M. (1990). Recognition of handwritten symbols. *Pattern Recognition*, **23**(11), 1155–1166.

Teague, M. R. (1980). Image analysis via the general theory of moments∗. *J. Opt. Soc. Am.*, **70**(8), 920–930.

Trier, O. D. and Taxt, T. (1995). Evaluation of binarization methods for document images. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **17**(3), 312–315.

Trier, Ø. D., Jain, A. K., and Taxt, T. (1996a). Feature extraction methods for character recognition - a survey. *Pattern Recognition*, **29**(4), 641–662.

Trier, Ø. D., Jain, A. K., and Taxt, T. (1996b). Feature extraction methods for character recognition - a survey. *Pattern Recognition*, **29**(4), 641–662.

Van Otterloo, P. J. (1991). *A Contour-Oriented Approach to Shape Analysis*. Prentice Hall International (UK) Ltd., Hertfordshire, UK, UK.

Veltkamp, R. C. (1998). Hierarchical approximation and localization. *The Visual Computer*, **14**(10), 471–487.

Veltkamp, R. C. and Hagedoorn, M. (1999). State-of-the-art in shape matching. Technical report, Principles of Visual Information Retrieval.

Verikas, A., Gelzinis, A., and Bacauskiene, M. (2011). Mining data with random forests: A survey and results of new tests. *Pattern Recogn.*, **44**(2), 330–349.

Vinicius, P., Borges, K., Mayer, J., and Izquierdo, E. (2007). A practical protocol for digital and printed document authentication. In *EUSIPCO 2007*, pages 2529–2533.

Wang, B. (2011). Shape retrieval using combined Fourier features. *Optics Communications*, **284**(14), 3504–3508.

Wang, B. and Shi, C. (2006). A novel Fourier descriptor for shape retrieval. In *Proceedings of the Third international conference on Fuzzy Systems and Knowledge Discovery*, pages 822–825, Berlin, Heidelberg. Springer-Verlag.

Wikipedia (Last Accessed: Nov 2012). Digital watermarking. http://en.wikipedia.org/wiki/Digital_watermarking.

Witten, I. H. and Frank, E. (1999). *Data Mining: Practical Machine Learning Tools and Techniques with Java Implementations*. Morgan Kaufmann Publishers, Inc.

Wright, M. and Fallside, F. (1993). Skeletonisation as model-based feature detection. *Communications, Speech and Vision, IEE Proceedings I*, **140**(1), 7 –11.

Yazici, A. and Sener, C., editors (2003). *Computer and Information Sciences - ISCIS 2003, 18th International Symposium, Antalya, Turkey, November 3-5, 2003, Proceedings*, volume 2869 of *Lecture Notes in Computer Science*. Springer.

Zahn, C. T. and Roskies, R. Z. (1972). Fourier descriptors for plane closed curves. *IEEE Transactions on Computers*, **21**(3), 269–281.

Zhang, D. and Lu, G. (2001). A comparison of shape retrieval using Fourier descriptors and short-time Fourier descriptors. In *Proceedings of the Second IEEE Pacific Rim Conference on Multimedia: Advances in Multimedia Information Processing*, PCM '01, pages 855–860, London, UK, UK. Springer-Verlag.

Zhang, D. and Lu, G. (2004). Review of shape representation and description techniques. *Pattern Recognition*, **37**, 1–19.

Zhang, D. and Lu, G. (2005). Study and evaluation of different Fourier methods for image retrieval. *Image and Vision Computing*, **23**(1), 33–49.

Zhang, G. P. (2000). Neural networks for classification: a survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C*, **30**(4), 451–462.

Zhang, H. (2004). The optimality of naive bayes. In V. Barr and Z. Markov, editors, *FLAIRS Conference*. AAAI Press.

Zhou, J. and Chen, P. (2009). Generalized discrete cosine transform. In *Circuits, Communications and Systems, 2009. PACCS '09. Pacific-Asia Conference on*, pages 449–452.

Zhu, B., Wu, J., and Kankanhalli, M. S. (2003). Print signatures for document authentication. In *Proceedings of the 10th ACM conference on Computer and communications security*, CCS '03, pages 145–154, New York, NY, USA. ACM.

Zi, G. and Doermann, D. S. (2004). Document image ground truth generation from electronic text. In *ICPR (2)*, pages 663–666.