Digital watermarking for PDF documents and images : security, robustness and AI-based attack

Makram Hatoum

UBFC
UNIVERSITÉ
BOURGOGNE FRANCHE-COMTÉ

UFC
UNIVERSITÉ
DE FRANCHE-COMTÉ

THÈSE DE DOCTORAT DE L'ÉTABLISSEMENT UNIVERSITÉ BOURGOGNE FRANCHE-COMTÉ

PRÉPARÉE À L'UNIVERSITÉ DE FRANCHE-COMTÉ

École doctorale n°37

Sciences Pour l'Ingénieur et Microtechniques

Doctorat d'Informatique

par

MAKRAM HATOUM

Digital Watermarking for PDF Documents and Images: Security, Robustness and AI-based attack

Tatouage Numérique pour les Documents PDF et Images: Sécurité, Robustesse et Attaque basée sur l'IA

Thèse présentée et soutenue à Belfort, le 23 Septembre 2020

Composition du Jury :

SPITERI PIERRE	Professeur à l'Institut National Polytechnique	Président
MACQ BENOÎT	Professeur à l'École Polytechnique de Louvain	Rapporteur
SPITERI PIERRE	Professeur à l'Institut National Polytechnique	Rapporteur
CONTASSOT-VIVIER SYLVAIN	Professeur à l'Université de Lorraine	Examinateur
COUCHOT JEAN-FRANÇOIS	Professeur à l'Université Bourgogne-Franche Comté	Directeur de thèse
DARAZI RONY	Professeur à l'Université Antonine	Codirecteur de thèse

N° | 8 | 6 | 3 | 1 | 7 |

SPIM

école doctorale **sciences pour l'ingénieur et microtechniques**

UBFC
UNIVERSITÉ
BOURGOGNE FRANCHE-COMTÉ

Université Bourgogne Franche-Comté
32, avenue de l'Observatoire
25000 Besançon, France

ABSTRACT

Digital Watermarking for PDF Documents and Images: Security, Robustness and AI-based attack

Makram Hatoum
University of Bourgogne Franche-Comté, 2020

Supervisors: Jean-François Couchot, Rony Darazi

Technological development has its pros and cons. Nowadays, we can easily share, download, and upload digital content using the Internet. Also, malicious users can illegally change, duplicate, and distribute any kind of information, such as images and documents. Therefore, we should protect such contents and arrest the perpetrator. The goal of this thesis is to protect PDF documents and images using the Spread Transform Dither Modulation (STDM), as a digital watermarking technique, while taking into consideration the main requirements of transparency, robustness, and security.
STDM watermarking scheme achieved a good level of transparency and robustness against noise attacks. The key to this scheme is the projection vector that aims to spreads the embedded message over a set of cover elements. However, such a key vector can be estimated by unauthorized users using the Blind Source Separation (BSS) techniques. In our first contribution, we present our proposed CAR-STDM (Component Analysis Resistant-STDM) watermarking scheme, which guarantees security while preserving the transparency and robustness against noise attacks.
STDM is also affected by the Fixed Gain Attack (FGA). In the second contribution, we present our proposed N-STDM watermarking scheme that resists the FGA attack and enhances the robustness against the Additive White Gaussian Noise (AWGN) attack, JPEG compression attack, and variety of filtering and geometric attacks. Experimentations have been conducted distinctly on PDF documents and images in the spatial domain and frequency domain.
Recently, Deep Learning and Neural Networks achieved noticeable development and improvement, especially in image processing, segmentation, and classification. Diverse models such as Convolutional Neural Network (CNN) are exploited for modeling image priors for denoising. CNN has a suitable denoising performance, and it could be harmful to watermarked images. In the third contribution, we present the effect of a Fully Convolutional Neural Network (FCNN), as a denoising attack, on watermarked images. STDM and Spread Spectrum (SS) are used as watermarking schemes to embed the

watermarks in the images using several scenarios. This evaluation shows that such type of denoising attack preserves the image quality while breaking the robustness of all evaluated watermarked schemes.

Keywords: Digital watermarking, Security, Robustness, Transparency, Deep Learning, Data hiding, STDM, Portable Document Format.

RÉSUMÉ

Tatouage Numérique pour les Documents PDF et Images: Sécurité, Robustesse et Attaque basée sur l'IA

Makram Hatoum
Université de Bourgogne Franche-Comté, 2020

Superviseurs: Jean-François Couchot, Rony Darazi

Le développement technologique a ses avantages et ses inconvénients. Nous pouvons facilement partager et télécharger du contenu numérique en utilisant l'Internet. En outre, les utilisateurs malveillants peuvent aussi modifier, dupliquer et diffuser illégalement tout type d'informations, comme des images et des documents. Par conséquent, nous devons protéger ces contenus et arrêter les pirates. Le but de cette thèse est de protéger les documents PDF et les images en utilisant la technique de tatouage numérique Spread Transform Dither Modulation (STDM), tout en tenant compte des exigences principales de transparence, de robustesse et de sécurité.
La méthode de tatouage STDM a un bon niveau de transparence et de robustesse contre les attaques de bruit. La clé principale dans cette méthode de tatouage est le vecteur de projection qui vise à diffuser le message sur un ensemble d'éléments. Cependant, un tel vecteur clé peut être estimée par des utilisateurs non autorisés en utilisant les techniques de séparation BSS (Blind Source Separation). Dans notre première contribution, nous présentons notre méthode de tatouage proposé CAR-STDM (Component Analysis Resistant-STDM), qui garantit la sécurité tout en préservant la transparence et la robustesse contre les attaques de bruit.
STDM est également affecté par l'attaque FGA (Fixed Gain Attack). Dans la deuxième contribution, nous présentons notre méthode de tatouage proposé N-STDM qui résiste l'attaque FGA et améliore la robustesse contre l'attaque Additive White Gaussian Noise (AWGN), l'attaque de compression JPEG, et diversité d'attaques de filtrage et géométriques. Les expérimentations ont été menées sur des documents PDF et des images dans le domaine spatial et le domaine fréquentiel.
Récemment, Deep Learning et Neural Networks atteints du développement et d'amélioration notable, en particulier dans le traitement d'image, la segmentation et la classification. Des modèles tels que CNN (Convolutional Neural Network) sont utilisés pour la dé-bruitage des images. CNN a une performance adéquate de dé-bruitage, et il pourrait être nocif pour les images tatouées. Dans la troisième contribution,

nous présentons l'effet du FCNN (Fully Convolutional Neural Network), comme une attaque de dé-bruitage, sur les images tatouées. Les méthodes de tatouage STDM et SS (Spread Spectrum) sont utilisés durant les expérimentations pour intégrer les messages dans les images en appliquant plusieurs scénarios. Cette évaluation montre qu'un tel type d'attaque de dé-bruitage préserve la qualité de l'image tout en brisant la robustesse des méthodes de tatouages évalués.

Mots Clés: Tatouage numérique, Sécurité, Robustesse, Transparence, Apprentissage Profond, Masquage des données, STDM, Portable Document Format.

ACKNOWLEDGMENTS

This Ph.D. thesis is the output of the effort and support of several people to whom I am extremely grateful. First and foremost, I thank my supervisors Prof. Jean-François COUCHOT and Prof. Rony DARAZI, for guiding and supporting me over the years in all stages of this work. You offered me valuable suggestions for this study. During the preparation of the thesis, you spent much time improving publications and provide me relevant advice. Their immense knowledge and plentiful experience have encouraged me all the time of my academic research and daily life.

My sincere gratitude also goes to Prof. Raphaël COUTURIER for motivation, valuable ideas, and immense knowledge. You have also given me access to the laboratory and research facilities. Without their precious support, it would not be possible to conduct this Ph.D. thesis. It has been a privilege to work with you all. I hope to be able to work with you again soon.

I would like to thank the jury members Prof. Sylvain CONTASSOT-VIVIER, Prof. Benoît MACQ, and Prof. Pierre SPITERI, for their presence and guidance through this process; Discussion, valuable ideas, and feedback. Thank you for your brilliant questions and comments. I would also like to thank all of my colleagues and friends who supported me during this work of research.

Finally, I would like to express my gratitude to my family, who had a fundamental role in getting me through the Ph.D. process successfully. Without their tremendous support and encouragement in the past few years, it would be impossible for me to complete my study and overcome all the difficulties during this work.

Contents

I

INTRODUCTION

1

INTRODUCTION

1.1/ GENERAL INTRODUCTION

Digital networks are essential communication mechanisms that are used to transfer any sort of information, such as text, audio, and image. With the rapid growth of the volume of exchange data over the internet, access to data sets has become much easier, which produced a significant number of problems, such as illegal distribution, duplication, and malicious tampering of digital data. Authors and data providers are concerned about protecting their data or work to be distributed in a network environment. Among many approaches of protection, digital watermarking has received wide attention and interest, especially for images and videos.
Besides, electronic documents such as Portable Document Format (PDF) are widely exchanged over the internet, and exposed to illegal copying and redistribution. Therefore, protecting PDF documents against malicious users is very important and necessary. PDF is a digital form for representing documents, which is developed and specified by Adobe systems society. PDF is an advanced imaging model based on a structured binary file format and has a lot of functions, including text and images. All around the world, use PDF documents to store, represent, and exchange a variety of content such as graphics and text. Consequently, such content must be protected against malicious users. This goal could be achieved using watermarking methods while taking into account the perceptual similarity, security, and robustness against several types of attacks.
Watermarking is not a new phenomenon. From a thousand years, papers have been visibly watermarked by a logo to identify the copyright. At 1282 in Italy, the papers have been marked by adding thin wire patterns. By the eighteen century, America and Europe were used the watermark as a trademark and as anti-counterfeiting measures on documents and money.
Digital watermarking consists of hiding a watermark, such as an image or text, in digital content like audio, image, video, and text for several purposes, such as copyright protection, broadcast monitoring, authentication, and access control. The watermark could be embedded in the cover work using the spatial domain or the transform domain according to the target requirements. The watermarking techniques are characterized by different properties, such as imperceptibility, payload, robustness, and security. Though, each watermarking technique met a set of the mentioned requirements.
The watermarked signals could be affected by different types of attacks, such as geometric distortions, lossy compression, filtering, and additive noise attacks. Each watermarking scheme survives specific types of attacks based on the target application. Recently, Deep learning and neural networks achieved noticeable development and improvement,

especially in image processing, segmentation, and classification. This model could also be exploited for attacking watermarked signals. Therefore, the study of this topic is very important, so that should highlight the seriousness of such models when used in illegal ways.
In this thesis, we will focus on digital watermarking of PDF documents based on Spread Transform Dither Modulation (STDM), try on the reinforcement of its security and robustness, and evaluation of some attacks based on Deep Learning.

1.2/ CONTRIBUTIONS

The main objective of this dissertation is to protect the PDF documents and images using a blind digital watermarking scheme while taking into account the transparency, security, and robustness against different types of attacks. We summary the main contributions of this work as follow:

- Most of the watermarking schemes have been improved to increase the level of transparency and robustness while the security has received little attention and interest. The security is one of the main requirements, which relies on the power and goals of an adversary, whose aim to modify, copy, or remove the watermark. Blind Source Separation (BSS) techniques such as Principal Component Analysis (PCA) and Independent Component Analysis (ICA) could be used to successfully attack the STDM watermarking scheme. Therefore, we have proposed the CAR-STDM (Component Analysis Resistant-STDM) to improve the security against the aforementioned BSS techniques.

- STDM is robust against additive noise attack, but it is affected by the Fixed Gain Attack (FGA). This type of attack scales the host vector and shifts the watermark vector away from the original quantization cell. In this way, the decoded watermark will be different from the embedded one. Therefore, we have improved the STDM watermarking scheme to resist such type of attack and enhanced the robustness against the Additive White Gaussian Noise (AWGN) attack, and a variety of filtering and geometric attacks. Moreover, this approach is developed to be used as a blind watermarking scheme for images and PDF documents.

- Recently, Deep learning achieved a noticeable improvement in many topics. It provides a perfect solution to many problems, such as language processing, speech recognition, and image recognition. Deep learning models are exploited for image processing and have a suitable denoising performance. Such models could be used as a harmful attack in digital watermarking since the watermark is a sequence of embedded noise. Therefore, we have studied and evaluated this type of denoising against watermarking schemes, and it could be a deleterious attack in digital watermarking.

1.3/ OUTLINE

The overall organization of this thesis is outlined in 4 parts displayed as follows:

The first part outlines the general introduction and Contributions to this thesis.

The second part, entitled PDF Watermarking, is composed of Chapters 2 and 3. Chapter 2 provides an overview of digital watermarking, where we present the different classes and techniques of digital watermarking. Furthermore, we present in this chapter the watermarking attacks, and mainly the image processing attacks, geometric attacks, protocol attacks, and cryptographic attacks.
In Chapter 3, we give an overview of the Portable Document Format (PDF). We detail the main components, the File structure, and the text state parameters in PDF documents. Moreover, we present several proposed methods for digital watermarking and steganography in text and PDF documents, and we highlight a blind digital watermarking technique based on the STDM watermarking scheme.

The third part of this thesis, entitled Contribution, is composed of Chapters 4, 5, and 6 that are devoted to our contributions. In Chapter 4, we present the security weakness of the STDM watermarking scheme against the BSS techniques. After that, we describe our proposed CAR-STDM watermarking scheme that nullifies the effect of such algebraic methods. We also prove the effectiveness of the proposed method in terms of security, transparency, and robustness against noise attacks. This chapter corresponds to an article published in SECRYPT [103], the international conference on Security and Cryptography.
In Chapter 5, we present the robustness weakness of STDM against the Fixed Gain Attack (FGA). We outline our proposed N-STDM watermarking scheme that resists such kind of attack. We prove in this chapter the effectiveness of our proposed method on PDF documents and real images in the spatial domain and frequency domain against the FGA attack and variety of signal processing attacks, geometric attacks, and image filtering attacks. The performance of this method is evaluated by comparing it to other related works. This chapter reformulated many ideas published in the international journal Multimedia Tools and Applications [106].
In chapter 6, we present the Deep Learning and Neural Network that achieved noticeable development and improvement in image processing, segmentation, and classification. Specifically, we evaluate in this chapter the effect of a Fully Convolutional Neural Network Denoising Attack (FCNNDA) on watermarked images in different scenarios. FCNNDA is also compared to other types of attacks to examine the difference in terms of quality and robustness. It is the first time that we propose Deep Learning to attack watermarked images, and it could be a harmful type of attack for watermarked images. This chapter corresponds to an article submitted to an international journal (Signal Processing: Image Communication).

Finally, the fourth part concludes the thesis. In this chapter, we draw the major conclusion and elaborate the perspectives that will be addressed in the continuity of this work.

II

PDF WATERMARKING

2

DIGITAL WATERMARKING

2.1/ INTRODUCTION

Nowadays, digital contents such as texts, images, and videos are easily shared and transferred over the Internet. However, the advancement of Internet technologies also came with its own set of problems. Many online Softwares are available to duplicate digital content without any acknowledgment, which presents security-related problems. Therefore, engineers, artists, scientists, publishers, and all around the world need to protect their digital content against malicious users using copyright authentication and copyright protection. A digital signature could reduce copyright violation and determine the ownership of digital content.
Digital watermarking has been used and proposed to protect and identify owner or creator of audios, videos, documents, or images [47, 100]. In the late 1990s, digital watermarking gained wide popularity and began to mushroom, which have been used by many companies for a wide range of applications such as owner identification, broadcast monitoring, proof of ownership, and content authentication.
Digital watermarking may be seen as steganography, which is the art of concealing digital contents inside any other type of digital content, such as text, image, audio, and video [95]. Digital watermarking and steganography belong to the information hiding category, but they have opposite conditions and objectives. In digital watermarking, the important information is the external data that should be protected using internal data, which is the watermark in this case. Conversely, the important information in steganography is the internal data, and only the intended recipient should be able to detect the hidden information.
A watermarking system is generally composed of two distinct stages, as shown in Figure 2.1, which are the watermark embedding and watermark extraction. The watermark is embedded in the cover work using an embedding function and a watermarking key, which is used during the embedding and extraction process to prevent illegal access to the watermark. The watermark can later be detected and extracted from the watermarked signal. The watermarked content could be subject to different types of attacks. Therefore, the watermark scheme requires a trade-off between transparency, robustness, and other properties based on the target application.

- Transparency: This property is one of the basic requirements in digital watermarking, which refers to the perceptual similarity between the watermarked work and the original one.

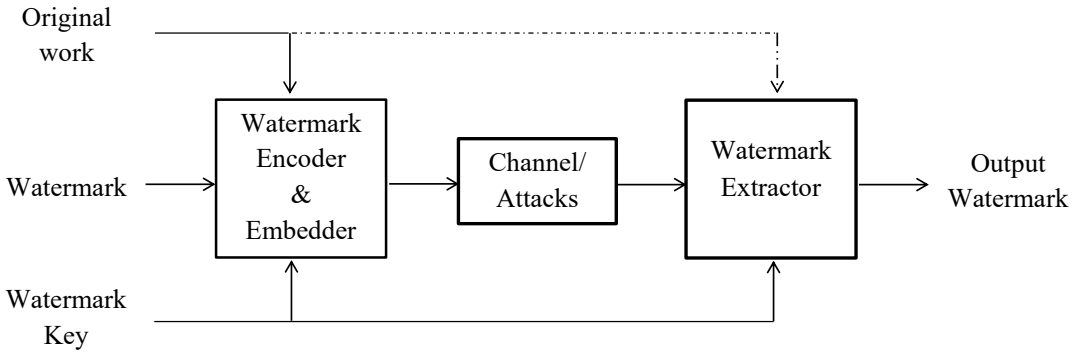


Figure 2.1: General Watermarking System

- Robustness: Watermarked contents are usually subject to different types of attacks prior to the extraction process. Robustness refers to the ability to extract and detect the watermark after applying such types of attacks. Therefore, the watermark should be robust against specific types of attacks that could occur before the detection process.

- Security:The robust watermark should survive normal processing, whereas a secure watermark relies on the goals and the power of an adversary to estimate the secret key used during the embedding and extraction process. And in this way, the adversary will be able to copy, modify, or remove the watermark. Such types of attacks could be concerned as unauthorized embedding, removal, or detection. Unauthorized embedding and removal attacks, also known as active attacks that modify the cover work. Unauthorized detection attack, also known as a passive attack that does not change the cover work.

- Capacity: Refers to the amount of data that could be embedded in the cover work. Specifically, it refers to the amount of information that could be hidden in the cover work without perceptible distortion, while preserving good robustness against several types of attacks.

Each of the above properties has an important role. They are dependent and have a side effect on each other. For example, decreasing the capacity could enhance the quality of the cover work and decrease the robustness, and vice versa. Therefore, a trade-off should be made between the mentioned properties based on the target application. The improvement of one property could affect the other requirements.
The remainder of this chapter is organized as follows. The watermarking classification is presented in Section 2.2. Section 2.3 presents some watermarking techniques. Some important watermarking attacks are presented in Section 2.4. Finally, Section 2.5 concludes the chapter.

2.2/ WATERMARKING CLASSIFICATION

Digital watermarking approaches are categorized as visible and invisible watermarking. For example, television channels use visible watermarks in the form of a logo presented on the corner of the television picture. On the other hand, authors and distributors used invisible watermarks as copyright to prove the ownership of data. The invisible watermarking approaches are classified as blind, semi-blind, and non-blind schemes. Non-blind methods require the original and watermarked contents during the extracting process to extract the watermark. The semi-blind methods require watermark or some side information during the extraction process. The Blind methods require only the watermarked contents during the extraction process to extract the watermark.

2.2.1/ SPATIAL DOMAIN

Digital watermarking methods are also classified into spatial domain and frequency domain. Each method has advantages and disadvantages, and it is selected based on the target application's requirements. In the spatial domain, the watermark is embedded by directly changing the intensity values of the cover work. For example, images are represented in the spatial domain in the form of pixels, and the watermark is embedded by modifying the intensity values of the selected pixels. Least Significant Bit (LSB) is one of the oldest popular methods in the spatial domain, through which the watermark is embedded by substituting the LSB of the selected pixels with the watermark bits [18, 30]. It is a simple method that provides good perceptual transparency, but it is vulnerable to common signal processing attacks, and the watermark can be easily removed and modified.

2.2.2/ FREQUENCY DOMAIN

In the frequency domain, the watermark is distributed over the whole cover work, and the inverse transformation produces the watermarked signal. The most common examples of the frequency domain are the Discrete Fourier Transform (DFT), Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT), and Singular Value Decomposition (SVD).
The Fourier Transform [19] is applied for many applications, such as image filtering, image compression, image analysis, and image watermarking. In this form, the image is decomposed into sine and cosine components, and each point represents a particular frequency. High frequencies contain less information than the lowest ones, which are located in the center of the resulted transform. Many watermarking approaches used the DFT as a transform domain to embed the watermark [14, 23, 24, 71]. DFT is used in digital watermarking because it has good robustness against geometric attacks such as cropping, rotation, and scaling.
The DFT could be applied directly to the host image or the non-overlapping blocks of the host image. Then, the selected coefficients could be modified based on the watermark bits and Pseudo-Noise (PN) sequences using such as the additive watermarking methods. After that, the watermarked image is obtained by applying the inverse DFT. Also, many approaches combined the DFT transform with DWT and DCT to improve the robustness and imperceptibility [69, 28].
DCT [8] is a frequency transform applied in several fields, such as signal processing,

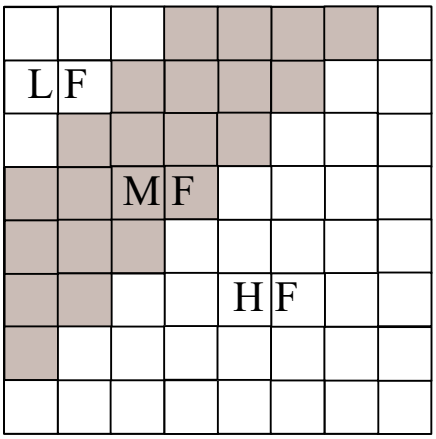


Figure 2.2: DCT Decomposition

data compression, and digital watermarking. DCT converts the image from the spatial domain to the frequency domain to get the cosine coefficients [17, 16, 32, 59]. Those coefficients are classified into low-frequency, middle-frequency, and high-frequency coefficients, as shown in Figure 2.2. Each band presents specific information, such as the low-frequency coefficients, which contain the important information of this image, and the high-frequency coefficients represent the sharp details of the image.
The DCT watermarking methods can be classified as Block based DCT and Global DCT watermarking methods. With the Global DCT, the transform is applied to the whole image as one block. With the Block based DCT, the image will be segmented into 8×8 non-overlapping blocks, and the DCT will be applied to each of these blocks. With this form, the watermark bit is embedded into the selected coefficients of each block.

DWT [9, 12] is a multi-resolution analysis used in various applications, such as signal and image processing, image compression, and image watermarking. DWT decomposes an image into low-frequency and high-frequency components that are presented in 4 sub-bands denoted as LL, LH, HL, and HH. The LL sub-band is known as the approximation part that covers the low-frequency components. LH, HL, and HH cover the high-frequency components that present the horizontal, vertical, and diagonal details of an image. The LL sub-band could be decomposed to another level of 4 sub-bands, as shown in Figure 2.3, and this decomposition could persist until we reach the desired number of levels. We can reconstruct simply the image by applying the inverse DWT.
For image, video, and audio watermarking, one or more sub-bands coefficients can be modified according to the watermark [15, 25, 27, 45, 49, 57].
The decomposition into different sub-bands using the DWT allows the embedding of the watermark with high energy in regions where the human visual system is less sensitive, which intuitively increases the robustness with lower degradation of the quality of the image.
SVD is a useful numerical technique of linear algebra that has demonstrated its feasibility and usefulness in several applications, such as in signal processing, image compression, and image watermarking [96, 77, 70, 56, 58, 31]. SVD decomposes a matrix of size $M \times N$ into three matrices: S, U, and V. The U and V are known as the unitary matrices of size $M \times N$, and S is known as the diagonal matrix or singular matrix of size $M \times N$. The diag-

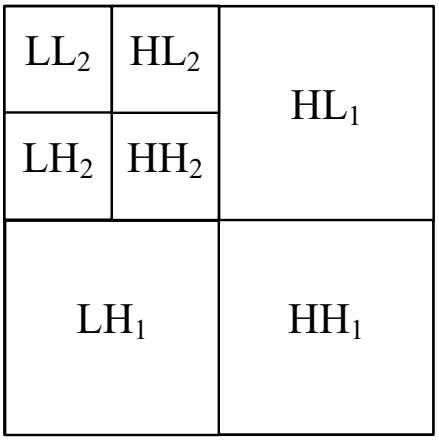


Figure 2.3: Two level DWT Decomposition

onal entries in the matrix S are known as singular values. The columns of matrix V and U are known as the right and left singular vectors, respectively. In image processing, the singular values represent the brightness of an image, and the singular vectors represent the geometry properties of an image [55].
A simple SVD-based watermarking approach consists of applying the SVD on the host signal, and after that, modifies the singular values based on the watermark using a gain factor. After that, we obtained the watermarked signal by multiplying the modified singular matrix by the unitary matrices U and V. In other approaches, the SVD is applied to non-overlapping blocks of the host signal, with a combination with other transforms, such as DWT and DCT.

2.3/ WATERMARKING TECHNIQUES

In the past years, a large number of watermarking embedding methods have been proposed, which are categorized into additive class and substitutive class. The additive class methods, also known as host-interference non-rejecting methods, presume that the host signal is analogous to a source of interference or noise. These methods are valuable when the channel interference is much larger than the host signal interference, or when the host signal is accessible to be used at the decoder.
The substitutive class methods, also known as host-interference rejecting methods, exploit knowledge of the host signal at the encoder. These methods included the Quantization Index Modulation (QIM) methods, which use the quantization function during the embedding process.

2.3.1/ HOST-INTERFERENCE NON-REJECTING METHODS

The system designs of host-interference non-rejecting methods do not allow an encoder to adequately exploit the host signal knowledge. Spread Spectrum (SS) methods are parts of the linear class of methods that belongs to the host-interference non-rejecting

methods [5]. The additive embedding function of such methods has the form:

$$s(x, m) = x + w(m), \tag{2.1}$$

where x represents the host signal, and $w(m)$ is a pseudo-noise sequence that takes the form:

$$w(m) = a(m)v, \tag{2.2}$$

where $a(m)$ represents the scalar function of the message, and v defines the unit energy. Additive SS is like a perturbation of a projection. By substituting equation (2.2) into equation (2.1) we got:

$$s = x + a(m)v. \tag{2.3}$$

By projecting s into v, we will obtain:

$$\hat{s} = s^T v = \hat{x} + a(m), \tag{2.4}$$

where $\hat{x}$ represents the projection of the host signal x onto v:

$$\hat{x} = x^T v. \tag{2.5}$$

By substituting equation (2.4) into equation (2.3), we got the reconstructed signal from the projection:

$$s = x + (\hat{s} - \hat{x})v \tag{2.6}$$

2.3.2/ Host-Interference Rejecting Methods

The system designs of host-interference rejecting methods allow an encoder to sufficiently exploit the knowledge of the host signal. Low-bit modulation (LBM) [4, 7] is a simple example of host-interference rejecting methods, which is a class of quantizes and replaces systems. These methods consist of replacing the least significant bits of a host sample by the message bits.
Specifically, the LBM method depends on two main steps. The first one consists of projecting the host signal vector x onto a pseudo-random vector v as in equation (2.5).
The second step consists of embedding the message bits using the embedding function presented in equation (2.6), wherein this case $\hat{s}$ has the form:

$$\hat{s} = s^T v = q(\hat{x}) + d(m), \tag{2.7}$$

where $d(m)$ represents the perturbation value, and $q(.)$ denotes the uniform scalar quantization function of step size Δ.

2.3.2.1/ Quantization Index Modulation

Quantization Index Modulation (QIM) are advanced methods introduced by Chen and Wornell [20] and belongs to the host-interference rejecting methods.
Non-linear modifications are performed by the QIM methods, and the received samples are quantized by mapping them to the nearest reconstruction point to detect the embedded message.
The main QIM property is to achieve a good level of robustness while preserving a small embedding induced distortion. To do so, the embedding function $s(x, m)$ and the identity function x must be approximately equal:

$$s(x, m) \approx x, \quad \forall m \tag{2.8}$$

and the points in one function range should be too far from the point of any other function range.These properties are defined as the approximate-identity property and the non-intersection property.
To be more clarified about these properties, let's consider an example where we have a message $m \in \{c_1, c_2\}$ to be embedded in a host signal x. To do so, we need two different quantizers Q_1 and Q_2. We will use Q_1 to embed $m = c_1$, and Q_2 to embed $m = c_2$.
This is shown in Figure 2.4, where ◯ represents the samples quantized by Q_1, and × represents the samples quantized by Q_2.
Let's consider that c_1=0 and c_2=1. In this way, we will embed one-bit message $m \in \{0, 1\}$ in sample x. With scalar-QIM, $Q_m(.)$ is used to quantize the sample x based on the bit value of the message m to get the marked sample s, which is defined as:

$$s = Q_m(x) = Q(x, \Delta_m) = round(\frac{x}{\Delta_m})\Delta_m \quad m \in \{0, 1\}, \tag{2.9}$$

where Δ represents the quantization step size, which determines the watermark strength, and $round(.)$ denotes the rounding operation.
To measure the embedding distortion, we could use such as the mean square error, which computes the means of distortion between the watermarked signal and the original one:

$$D(s, x) = \frac{1}{L}\|s - x\|^2. \tag{2.10}$$

On the other hand, we could also determine the robustness of the embedding watermark by computing the minimum distance between the sets of reconstruction points of different quantizers in the ensemble as:

$$d_{min} \triangleq \min_{(i,j):i\neq j} \min_{x_i, x_j} \|s(x_i; i) - s(x_j; j)\| \tag{2.11}$$

We could detect the watermark in QIM without access to the original watermark or the original signal using the minimum distance decoder:

$$\hat{m} = \arg\min_{m\in\{0,1\}} |y - s(y, m)|, \tag{2.12}$$

where $\hat{m}$ represents the extracted message, and y represents the watermarked signal affected by different types of attacks.

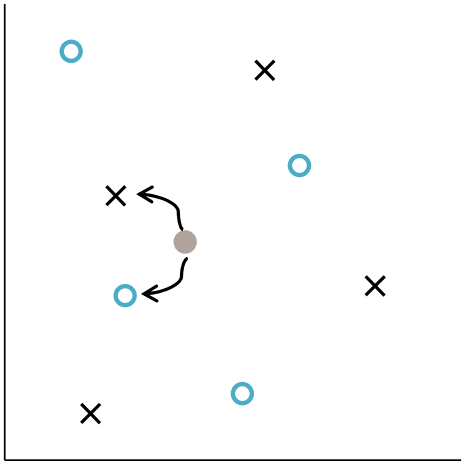


Figure 2.4: Quantization Index Modulation

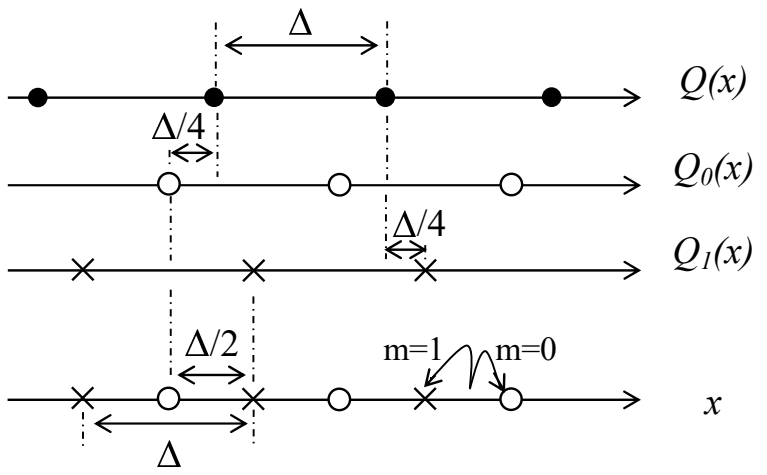


Figure 2.5: Basic Dither Modulation with Scalar-QIM

2.3.2.2/ Dither Modulation with Scalar-QIM

The practical implementation of QIM is based on Dither Modulation. The property of dithered quantizers is that the reconstruction points and quantization cells of a given quantizer are shifted versions of the reconstruction points and quantization cells of any other quantizer. Therefore, to embed a bit message $m \in \{0, 1\}$ in the host signal x, we need two dither values d_0 and d_1 to generate the two quantizers Q_0 and Q_1 based on m. Thus, the marked samples are defined as:

$$s(x, m) = Q_m(x) = Q(x - d_m) + d_m \quad m = \{0, 1\}. \tag{2.13}$$

The dither values are modulated by m, and are usually chosen as:

$$d_0 = -\frac{\Delta}{4}, \quad d_1 = \frac{\Delta}{4}$$

But the dither values could be also chosen pseudo-randomly from $[-\Delta/2, \Delta/2]$:

$$\begin{aligned} &\text{If } d_0 < 0 \quad d_1 = d_0 + \Delta/2 \\ &\text{If } d_0 \geq 0 \quad d_1 = d_0 - \Delta/2 \end{aligned}$$

In this form, d_1 is formed by subtracting or adding $\Delta/2$ to d_0 depending on the sign of d_0. This form is presented in Figure 2.5, and we can notice that the maximum induced error d_{min} is equal to $\Delta/2$.

2.3.2.3/ Spread Transform Dither Modulation

Spread Transform Dither Modulation (STDM) is a special case of QIM where the quantization occurs entirely in the projection of the host signal x onto a normalized projection

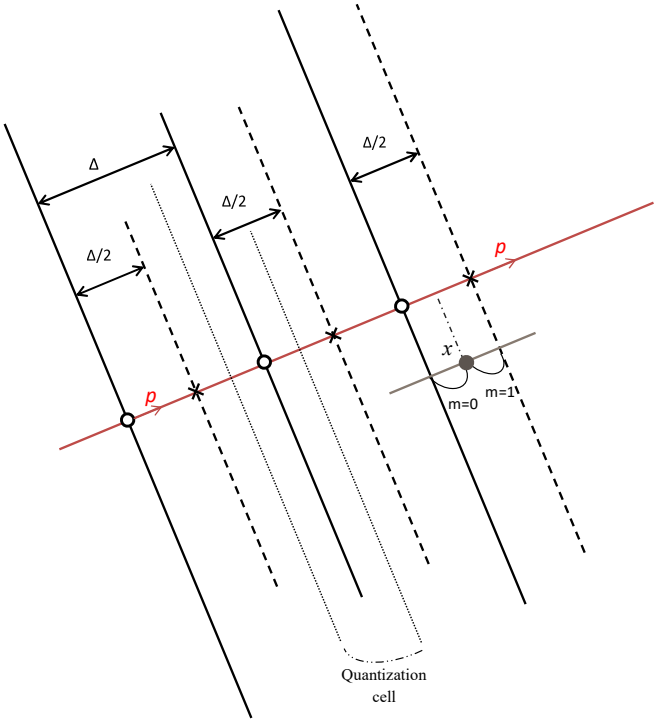


Figure 2.6: Geometrical representation of Spread Transform Dither Modulation

vector p. This way, the embedding-induced distortion spreads into a group of samples rather than one.

For example, if we have a host signal x of length N and watermark bits of length M. First of all, the signal or the cover work will be divided into M segments with equal length L, which is the length of the projection vector p. After that, the i^{th} bit will be embedded in the i^{th} segment. Specifically, the i^{th} bit will be spread in the projection of the i^{th} segment onto the projection vector p.
The embedded function is as follows:

$$\begin{aligned} y &= x + (Q_m(x^T p, \Delta) - x^T p)p \\ &= x + \left(round\left(\frac{x^T p - d_m}{\Delta} \right) \Delta + d_m - x^T p \right) p, \end{aligned} \tag{2.14}$$

where Δ represents the quantization factor, $round()$ is the rounding value to the nearest integer, and d_m denotes the dither level based on the message bit $m \in \{0, 1\}$.
The detection can be performed with a minimum distance decoder to extract the embedded message as follows:

$$\hat{m} = arg \min_{m \in \{0,1\}} \mid y^T p - Q_m(y^T p, \Delta) \mid . \tag{2.15}$$

The decision of the STDM decoder is based on the projection of the channel output y onto p.

Comparing to other QIM methods, STDM has an additional specification. STDM can mix the advantage of LBM and additive SS. By replacing equation (2.7) of LBM:

$$\hat{s} = s^T v = q(\hat{x}) + d(m)$$

into equation (2.6) of SS:

$$s = x + (\hat{s} - \hat{x})v$$

we will get the embedding equation (2.14) of STDM:

$$s = x + ([q(\hat{x}) + d(m)] - \hat{x})v \tag{2.16}$$

Figure 2.6 illustrates the geometrical representation of STDM. The points on dashed-lines represent the embedding for $m = 1$, and the points on solid-lines are for $m = 0$. The host-vector is quantized to the nearest point on an $\times$-line to embed the bit $m = 1$, and on the nearest point on an $\circ$-line to embed the bit $m = 0$. Noting that the quantization is done after the projection of the host-vector onto p. With STDM, the minimum distance d_{min} between quantizers in the ensemble is equal to $\Delta/2$, and the mean distortion D_s is equal to $\Delta^2/12L$ if the quantization error is uniformly distributed over [-Δ/2, Δ/2] [21].
The choice of the projection vector p and the quantization step size Δ make the trade-off between the transparency and robustness of the watermark.

2.4/ Watermarking Attacks

Digital watermarking methods were developed and improved in recent years for various purposes. The watermark could be embedded to be private, public, fragile, or robust. The watermark is defined as a private watermark when the original signal is available during the detection process. In contrast, the watermark is defined as a public watermark when the original signal is not needed during the detection process. The fragile watermark intends to recognize whether the watermarked signal has been adjusted or altered. However, the robust watermark aims to outrun the common image processing attacks or the intentional attempts to alter or remove it [3, 22, 42].
The embedded watermark could be destroyed or altered by an intentional attacker, such as the image forgery, for illegal purposes. Also, the embedded watermark could be affected unintentionally by using the JPEG compression. However, the watermarking attacks could be categorized as image processing attacks, geometric attacks, protocol attacks, and cryptographic attacks.

2.4.1/ Image Processing Attacks

The embedded watermarks in images could be destroyed or affected by image processing attacks that include the filtering, denoising, JPEG compression, and re-modulation attacks.
Median, Average, Wiener, Gaussian, and sharpen filters are part of image filtering attacks that could destroy the watermark embedded in the watermarked images. The median filter is a non-linear digital filtering technique that preserves the edges in the image while removing noise. The average filter reduces the amount of intensity variation between pixels; each pixel value is replaced with the mean value of its neighbors, including itself.

Wiener filter is usually used for the removal of blur in images. The Gaussian filter usually used to blur the image and to reduce contrast and noise. Sharpening filter increases the magnitude of the high frequencies. In a spatial domain, the sharpening filter contains positive values surrounded by negative values.
The re-modulation attack was presented by Langelaar *et al.* [10]. This type of attack predicts the embedded watermark by subtracting median filtered images of the watermarked image. The basis of such an attack consists of denoising the watermarked image, altering the estimated watermark, and adding it to specific locations in the image. This type of attack is quite efficient with the additive watermarking methods.
Joint Photographic Experts Group, known as JPEG, was created in 1986 and approved by the ISO in 1994. JPEG is used for image compression. It can reduce the normal size of an image to about 5%. However, with the JPEG compression, some details of the compressed images are lost.

2.4.2/ Geometric Attacks

Geometric attacks are geometric distortions to an image which include operations such as scaling, rotation, cropping, clipping, flipping, and translation [36, 6]. They are classified basically into local and global geometric attacks. Local geometric attacks affect portions of an image using such as the cropping attack, and the global geometric attacks affect all the pixels of an image using such as the rotation and the scaling attacks.
The scaling attack consists of adjusting the size of an image by down-sampling or up-sampling the width and the length of the image. Rotation attack rotates the watermarked image by different angles by changing the coordinate axes. The cropping attack consists of cutting a specific part in the watermarked image. However, the clipping attack keeps the central quarter of the watermarked image. Translation attack repositions an image by shifting a coordinate location to other coordinate location along a straight line. Flipping or mirroring attack flips an image vertically or horizontally.

2.4.3/ Protocol Attack

The purpose of the protocol attack is to shade an ambiguity regarding the true ownership of the watermarked signal. Copy attack [26] is an example of the protocol attack, that consists of estimating the embedded watermark from the watermarked signal and copy it to another signal, instead of destroying it. Predicting the watermark and replicating it on other signal leads to the ambiguity regarding the true ownership of the watermarked signal.

2.4.4/ Cryptographic Attack

The cryptographic attack aims to retrieve or find the key used during the embedding process of a watermark based on exhaustive key searches. When an adversary finds the key, it could simply overwrite the watermark. Oracle attack belongs to the cryptography attack, which attempts to estimate the original image when the detector is available [11]. The watermark could be removed from the watermarked signal if the public decoder is available. Collusion attack also belongs to the cryptographic attack. This attack is used

when instances of the same image are available. In this case, the modified watermarked image results from the average of several watermarked images.

2.5/ CONCLUSION

In this chapter, an overview of digital watermarking has been provided. We have presented the watermarking methods that are classified into spatial domain and frequency domain. We have categorized them into additive class methods and substitutive class methods. Each method has advantages and disadvantages, and it is selected based on the target application's requirements. Therefore, the watermarking schemes require a trade-off between transparency, robustness, and other properties based on the target application. Furthermore, we have highlighted the watermarking attacks that are categorized as image processing attacks, geometric attacks, protocol attacks, and cryptographic attacks.

3

PORTABLE DOCUMENT FORMAT

3.1/ INTRODUCTION

Portable document format (PDF) was specified and developed by adobe systems in 1993, which was announced as PDF 1.0, and continuing until 2008 when it was released as an open standard and published as ISO 32000-1 PDF 1.7 by the International Organization for Standardization (ISO). After nine years in development, ISO has published the PDF version 2.0 in 2017, which is known as ISO 320000-2 PDF 2.0.
This chapter is inspired by the article of Bitar *et al*. [93] that presents a blind digital watermarking technique for PDF documents.
The remainder of this chapter is organized as follows. The PDF File Structure is presented in Section 3.2. Section 3.3 presents the text state in a PDF document. Related works of PDF watermarking and steganography are presented in Section 3.4. The blind digital watermarking technique for PDF documents is presented in Section 3.5. Finally, Section 3.6 concludes the chapter.

3.2/ PDF FILE STRUCTURE

PDF [51] was derived from the PostScript language as an advanced imaging model to enable users to view, exchange, and print the electronic documents easily on diverse devices and platforms. Unlike the PostScript programming language, PDF has a structured binary file format that includes objects that are useful for document interchange and interactive viewing. A PDF file can contain thousands of objects, different font formats, multiple compression mechanisms, images, graphs, and a variety of metadata. Therefore, over the past years, governments, businesses, libraries, and other institutions around the world have used the PDF as a standard for the electronic exchange of documents.
A PDF file is composed of four parts, as shown in Figure 3.1:

- Objects: Boolean values, strings, integer and real numbers, names, arrays, dictionaries, streams and null object are the basic types of objects that compose the data structure of a PDF document.
- Document structure: The document structure of PDF specifies how the PDF components, such as fonts, pages, and annotations are represented, using the basic

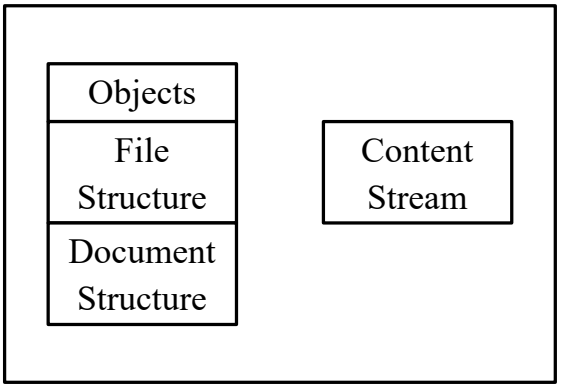


Figure 3.1: PDF Components

object types.

- Content Streams: A PDF page is composed of one or more content streams that describe the appearance of a page or other graphical entity based on a sequence of instructions.
- File structure: Header, body, cross-reference table, and trailer presented in Figure 3.2 are the 4 main components of a file structure that determines how objects are stored, accessed, and updated in a PDF file.

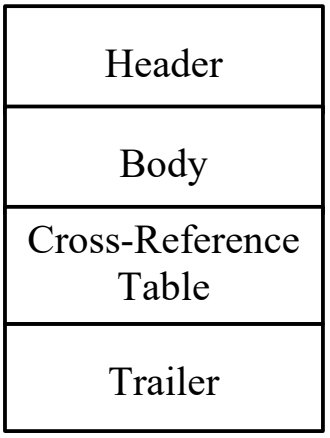


Figure 3.2: PDF and File Structure

The Header contains the version number of a PDF file. The header is located in the first line of the PDF file, which is composed of 5 characters '%PDF' followed by the version number.
The Body of PDF file consists of a series of objects, presented in the previous section, which represents the components of the document, such as pages, fonts, and sampled

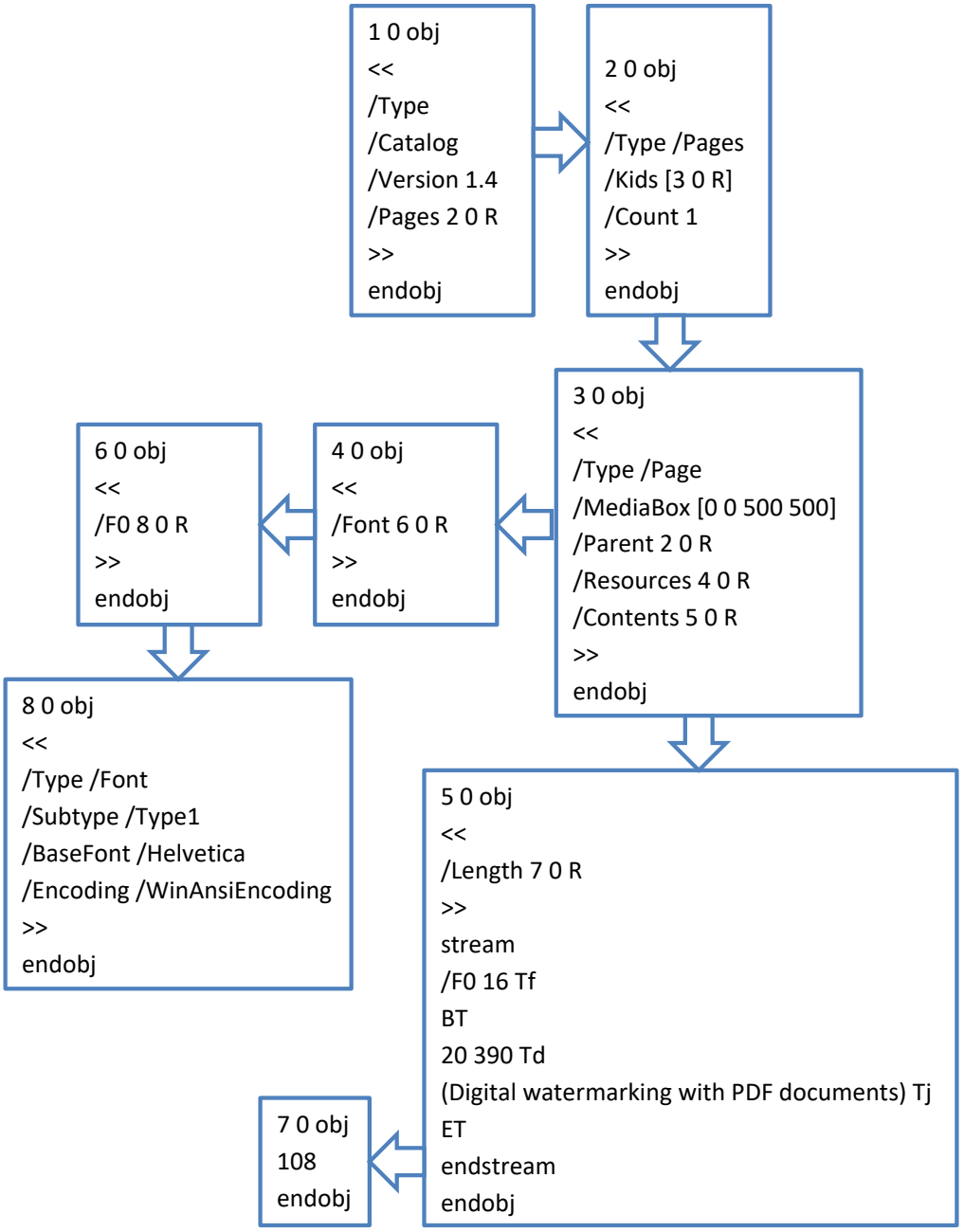


Figure 3.3: Body example of a PDF Document

images. All the objects in the Body element are organized as a linked list to represent the contents of a PDF document.
Let's consider a body example of a PDF file presented in Figure 3.3. This Body example will produce one page in PDF that contains the line "Digital watermarking with PDF documents". This Body begins with the Root object (1 0 obj), which has 0 as a generator and 1 as an identifier or object number. The content for object 1 is located between 1 0 obj and endobj. This object is a Catalog of type dictionary (<< >>), which contains the

version number 1.4, and has a reference to the object number 2 represented by /Pages 2 0 R. Noting that, each object started with a slash has the type 'name'. A second object is a Page object of type dictionary, which begins with 2 0 obj. In this part, we notice that we have only one page in the document, which is denoted by the object /Count. The key object Count is of type Name, and 1 is of type Numeric. This Page object also contains a reference to another object (Kids [3 0 R]), which represents more details about the page. The Page object located in the root 3 0 obj is also of type Dictionary, which contains a MediaBox that presents the length of the page. It contains a reference to the parent object 2, object 4, and object 5.
The 4 0 obj contains a reference to the Font object 6 0 obj, which contains a reference to 8 0 obj, which includes the used base Font Helvetica and encoding type WinAnsiEncoding. The object 5 contains information about the font size, which is presented using the operator Tf. Also, this object contains the position of the string, which is presented using the Td operator. To show the text on the page, the Tj operator is used.
In this example, "Digital watermarking with PDF documents" begin at the position 20 390, which refers to the x-coordinate and y-coordinate of the first character 'D', respectively. The x-coordinate values of the other characters depend on the horizontal spacing that is defined using the Tc operator. The Td and Tj operators are located between Bt and Et, which represents the begin of the text and the end of the text, respectively. Finally the object 5 contains a reference to object 7, which specifies the length of the string.
All these objects are presented in Figure 3.4 as a linked list. Each of the mentioned objects is represented by a node.

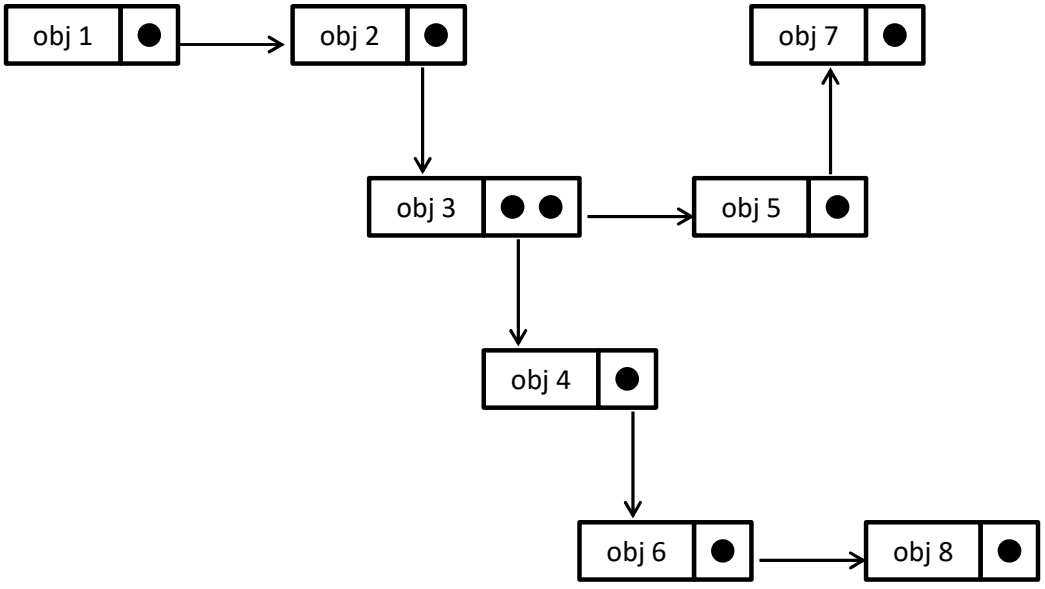


Figure 3.4: Body Linked List

Cross-Reference Table is the third element in the PDF file structure. Cross-Reference Table permits random access to indirect objects within the file. As shown in Figure 3.5,

the Cross-Reference table begins with the keyword xref 0 9. The first number 0 refers to object 0, which represents the head of the linked list. This object is a special sort of entry and doesn't exist in the PDF file. Therefore, the first line in the list ends with the marker "f". The second number in xref counts the existing number of objects in the Cross-Reference Tables. The lines after xref have a long of 20 bytes, and each line ending with the marker "n" indicates the existing object in the body element. In this way, each line in the Cross-Reference table refers to an indirect object, and the offset number represents the location of the object in the body element.

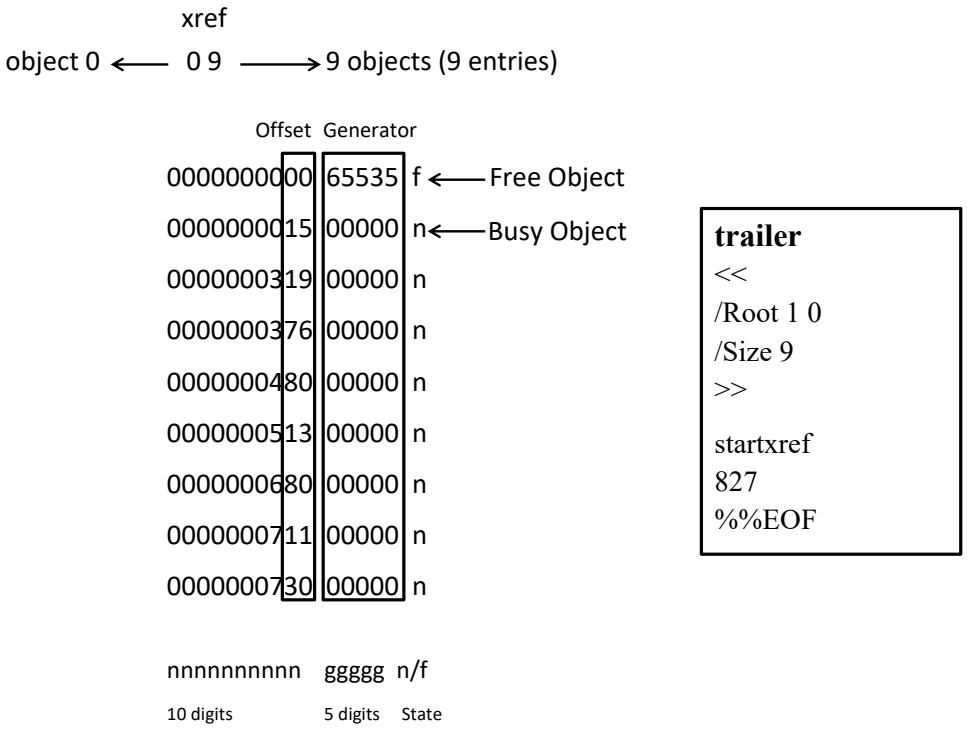


Figure 3.5: Cross-Reference Table and Trailer structures

The Trailer is the last element of the PDF structure. We can find certain special objects and the cross-reference table using the trailer of a PDF file. As shown in Figure 3.5, this trailer of type dictionary contains a link to the object Root 1 0, and also includes the object /Size 9, which represents the total number of objects. The offset of the cross-reference table is located between the keywords startxref and EOF (End Of File). Noting that, the trailer is read backwards from EOF till the trailer keyword.

3.3/ TEXT IN PDF DOCUMENT

Text can be drawn in a PDF file from multiple fonts to represent any language with any popular formats A font is a set of unique drawing instructions known as glyphs, which

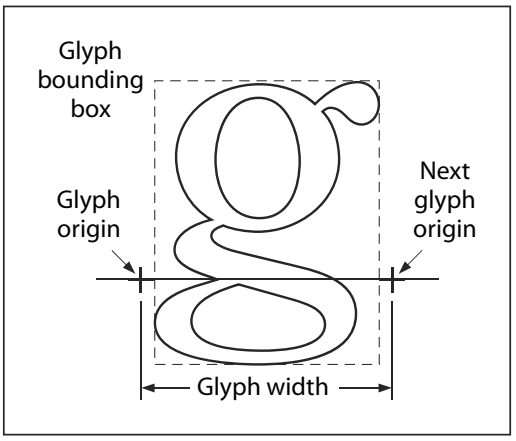


Figure 3.6: Glyph metrics

are the graphical rendering of characters. Each glyph has specific metrics, as shown in Figure 3.6, which present the glyph bounding box, the origin of the glyph, the width of a glyph, and the next origin glyph. The origin coordinate system of the first glyph is the point (0, 0), but the text space could be adjusted to another point, such as in the below example, the origin text space of the first character H is adjusted to the point (30, 60).

```
BT

    30 60 Td
    (HELLO) Tj

ET
```

Specifically, the text-showing operator shows the origin of the first glyph, which is placed at the new adjusted point. Also, the glyph in user space will be altered if the text space is scaled, translated, or rotated. The coordinate systems determine the orientation,

Table 3.1: Text state parameters

Parameter	**Description**
T_c	Character spacing
T_w	Word spacing
T_h	Horizontal spacing
T_l	Leading
T_{mode}	Text rendering
T_{rise}	Text rise
T_f	Text font
T_{fs}	Text font size

position, and size of the text. The real numbers x and y will form the coordinate pair that will locate a character or any point horizontally and vertically within two-dimensional coordinate space. The relationship between the coordinate spaces is also specified by a transformation matrix that will translate, rotate, scale, or transform in any other way the characters.
In a PDF document, the text could be affected based on different text state parameters presented in Table 3.1.
T_c operator is used to varying the space between the characters. The default value of T_c is equal to 0, and increasing this value will expand the distance between glyphs for horizontal writing from left to right. Some Asian writing has a vertical displacement. In this case, the distance between glyphs will be expanded using a negative value of the T_c operator. A simple example is shown in Figure 3.7, which shows the character spacing in horizontal writing.

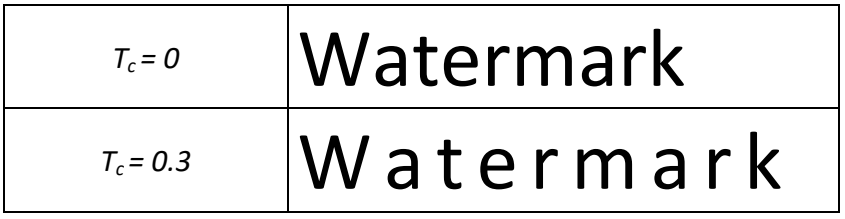


Figure 3.7: Character spacing

The space between words could also be varied using the T_w parameter. The default value of T_w is equal to 0. A positive value of T_w will increase the space between words for horizontal writing, and it will decrease the space for vertical writing. Figure 3.8 shows an example of word-spacing in horizontal writing.

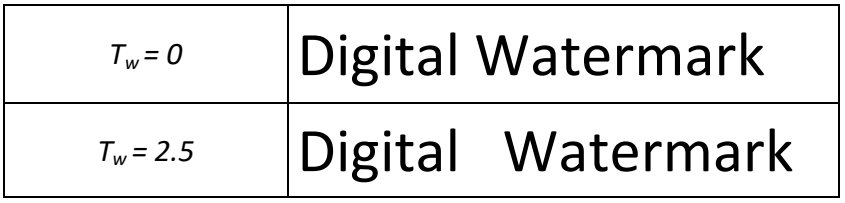


Figure 3.8: Word spacing

The T_h parameter will scale the glyphs in a horizontal direction. The default value of T_h is equal to 100. Increasing this value will stretch the glyphs, and decreasing it will compress the glyphs. Figure 3.9 shows an example of compressing the glyphs.

The T_l parameter is a leading parameter that specifies the vertical distance between the adjacent lines in a text. Whenever is the writing mode, this parameter is applied to the

$T_h = 100$	Watermark
$T_h = 50$	WatermarkWatermark

Figure 3.9: Word scaling

vertical coordinate. By default, the initial value is equal to 0.
Also, the outlines of a glyph could be filled, stroked, or used as a clipping boundary based on the text rendering mode (T_{mode}). Figure 3.10 shows an example of text rendering modes.

Mode	Example	Description
0	D	Fill
1	D	Stroke
2	D	Fill and Stroke

Figure 3.10: Example of text rendering modes

```
(Digital) Tj
-5 Ts
(Watermarking) Tj
5 Ts
(for) Tj
0 Ts
(PDF) Tj
-5 Ts
(Documents) Tj
```

Digital Watermarking for PDF Documents

Figure 3.11: Text rising

The T_{rise} parameter moves the baseline of the text up or down. Regardless of the writing mode, this parameter could be applied to the vertical coordinate, and the default value is

equal to 0. A negative value will move the baseline down, and a positive value will move it up. An example of text rise is shown in Figure 3.11.
The font is also an important attribute in the text state. We can specify a font and text size using the T_f operator. The first operand of the T_f parameter specifies the font, and the second one specifies the text size. Below is a simple example.

```
BT                                          % Begin the text

    /F13 16 Tf                              % Set The font and font size
    250 733 Td                              % Specifiy the starting position of the first character
    (HELLO) Tj                              % Paint the characters on the page

ET                                          % End the text
```

3.4/ PDF WATERMARKING AND STEGANOGRAPHY

Several methods of Digital watermarking and Steganography in Text and PDF documents have been proposed. Por and Delina [52] proposed a hybrid method in information hiding for text steganography using inter-paragraph and inter-word spacing. This method provides a large capacity to embed the message bits, but the embedded message could be destroyed by deleting or modifying some spaces between the paragraphs or the words in a text.
Wang and Tsai [53] proposed a data hiding technique in PDF documents based on different objects for covert communication. Hiding data is performed in the parameter of pages and text matrices. The properties of a page are described using page objects, such as the media box that describes the visible area of the page. This media box is composed of 4 numbers that specify the upper left corner and lower right corner of the page. The text matrix is composed of six numbers, which are used to describe the position, scale, and orientation of a text. The secret message is embedded by modifying such numbers. This technique achieves a good level of transparency, but it has a low embedding capacity.
Lee and Tsai [61] proposed a covert communication method for PDF documents. In this method, a secret message is embedded at the between-character and between-word locations in a text. The ASCII codes 20 and A0 could be used as white spaces between words. Therefore, the message bit could be encoded in a PDF text by placing the ASCII code A0 between two words when the bit value is equal to 0, and 20 when the bit value is equal to 0. The main drawback of this method is that an opponent could easily detect the embedded message by extracting the invisible ASCII codes from the text and decoding the secret message.
Alizadeh *et al.* [68] proposed alternative algorithms for the Tj method to hide messages inside PDF documents. In a PDF document, the text strings are displayed using the Tj operator, which contains an array of string and space values used between characters. A positive value decreases the space between the characters, and a negative value increases the space. The embedding capacity of this alternative method is less than the original one. Lin *et al.* [75] proposed copyright protection for PDF documents using the quadratic residue, which is usually used in cryptography. A proposed encryption technique combines the quadratic residue and information hiding technique in PDF as a basis to be applied to copyright protection. With this method, an opponent could easily remove

the hidden message.
Mehta *et al*. [89] used image watermarking algorithms for watermarking PDF files. In this method, the image text document is divided into texture and non-texture blocks. After this classification, the watermark is embedded in the texture blocks in a content-adaptive manner. DCT is used to compute the energy of each block to vary the watermark embedding strength between the blocks. The proposed approach is integrated with known image watermarking methods based on DWT, DCT, and SVD. This method could only be applied for scanned PDF documents.
Bitar *et al*. [93] proposed a blind digital watermarking scheme for PDF documents. This method consists of embedding the watermark in the x-coordinates of a group of characters, taking into consideration the transparency and robustness trade-off. The x-coordinates values are non-constant, and are exploited as watermarking space to embed the watermark. In this work, two distinct thresholds are computed by exploiting the robustness and transparency to get the acceptable distortion.
Kuribayashi *et al*. [104] proposed an approach that embeds a watermark in a PDF file into the spaces of characters in a text. In this method, the space lengths are grouped in a host-vector, and the DCT frequency transform is applied to the resulted vector. After that, the watermark is embedded into the AC components using the Dither Modulation Quantization Index Modulation (DM-QIM). The proposed data hiding method achieved a good embedding capacity with low distortions.
Nursiah *et al*. [107] proposed a data hiding technique in a PDF document that does not cause any visual distortion. The message is hidden by modifying the positioning coordinate values of the glyph based on the leading zero strategies. Specifically, adding a leading zero to the glyph positioning value (e.g., 031.2) will indicate the bit “1”, and the absence of the leading zero (e.g., 31.2) will indicate the bit “0”. The main drawback of this method is that an opponent could easily modify or remove the embedded message by modifying the leading zero in the glyph positioning coordinate values.

3.5/ Blind Digital Watermarking in PDF Documents

This section presents the blind digital watermarking technique for PDF documents based on the Spread Transform Dither Modulation (STDM) method [93]. The main challenge for digital watermarking in PDF document is to find the appropriate watermarking space to embed the watermark with an appropriate tradeoff between the Transparency and Robustness constraints.
In a PDF file, coordinate systems determine the position, orientation, and size of the text, images, and graphics that appear on a page. Each character has a coordinate pair x and y that will locate the character horizontally and vertically within two-dimensional coordinate space. The x-coordinates values are non-constant, as shown in Figure 3.12. Therefore, they can be exploited as the watermarking space to embed the watermark. We can also use the y-coordinates values as watermarking space, but that will decrease the embedding capacity. For example, if we take a line from a text, which could contain around 100 characters. All the characters in this line have the same y-coordinate value, but 100 different x-coordinates values. The capacity level in this case of the x-coordinates increased 100 times compared to the y-coordinate per line.

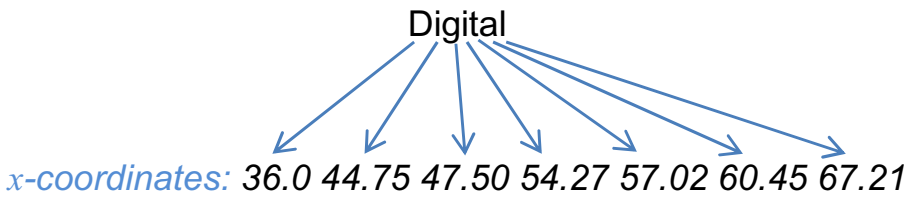


Figure 3.12: An example of the x-coordinates values of the word Digital

3.5.1/ Embedding and Decoding Concepts

All the necessary resources such as x-coordinate, y-coordinate, width, and height of each character should be extracted from the original document. After that, the x-coordinates values of the selected characters would be grouped in a vector to form the host signal through which the watermark will be embedded. After that, each bit of the watermark will be embedded into the samples of the host signal using the STDM watermarking scheme by quantizing the signal as presented in equation (2.14).
Let's suppose that k is the length of the watermark bits. In this case, we need $k \times L$ x-coordinates values to embed the whole watermark. Each bit of the watermark will be spread into L different x-coordinates values.
Assume that b_0=0 and b_1=1 are two bits of the watermark to be embedded in the x-coordinates values shown in Figure 3.13, where the length of projection vector L is equal to 8. Thus, b_0 would be embedded using the dither level d_0 and quantizer Q_0, and b_1 would be embedded using the dither level d_1 and quantizer Q_1. As a result, 8×2 characters are needed to embed 2 bits of the watermark. Each bit is inserted into 8 x-coordinates values of 8 characters that have the position (x_0, y_0), (x_1, y_1), (x_2, y_2), (x_3, y_3), (x_4, y_4), (x_5, y_5), (x_6, y_6), and (x_7, y_7). After that, we will get the modified positions of the characters (x'_0, y_0), (x'_1, y_1), (x'_2, y_2), (x'_3, y_3), (x'_4, y_4), (x'_5, y_5), (x'_6, y_6), (x'_7, y_7), and so on.

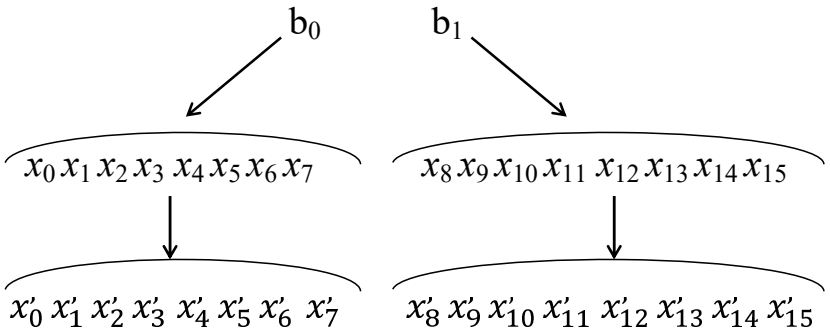


Figure 3.13: A basic example of spreading two bits into the x-coordinates values

The detection is performed with a minimum distance decoder presented in equation (2.15) to extract the embedded watermark.

3.5.2/ EXPERIMENTS

The STDM watermarking scheme have an average expected distortion $D_s = \Delta^2/12L$ [21]. Based on D_s, we could know the acceptable value of the watermarking strength Δ that would lead to a good level of robustness with sufficient transparency.
The robustness and transparency constraints are two opposite objectives. Therefore two threshold levels are considered during the experimental tests, namely t and r. The threshold t is computed in the transparency experiments, and the threshold r is computed in the robustness experiments. Thus, when the distortion D_s is greater than r, the STDM method ensures a good level of robustness, and when D_s is inferior to t, the STDM method ensures a good level of transparency. Therefore, sufficient robustness and transparency are achieved with an interval acceptable distortion $r \leq D_s \leq t$.
For example, suppose that r=0.3 and t=0.4. In this case, if D_s is inferior to 0.3, we will get sufficient transparency with weak robustness. Otherwise, if D_s is greater than 0.4, we will get sufficient robustness with weak transparency. But if D_s belongs to the integral [0.3 0.4], we will get sufficient transparency with sufficient robustness.
Robustness and transparency experiments are performed to deduce the best approximation values of r and t. The proposed method is tested on a paragraph in a PDF document that contains a number of characters n equal to 952. The watermark "LAW" to be embedded in the x-coordinate values of the characters is encoded into 8 bit to form a total of k=24 bits. The length of the projection vector used during the embedding and extraction process is equal to 39 (n/k=952/24≈39).

3.5.2.1/ TESTS OF TRANSPARENCY

The transparency of the proposed method is tested in different kinds of experiments under several values of Δ: 0.1, 0.5, 1, 1.5, 2, 2.5, 3, 5, 8, and 10. The error measurements between the original and watermarked documents are presented in Table 3.2 using dif-

Table 3.2: Error computations between the original and modified documents in terms of their x-coordinate values

Quantization step (Distortion)	MSE	RSE	MAE
Δ =0.1 (D_s=0.00002)	2.2444×10^{-5}	0.0047	0.0036
Δ =0.5 (D_s=0.00053)	6.3574×10^{-4}	0.0252	0.0175
Δ =1 (D_s=0.00214)	0.0023	0.0484	0.0365
Δ =1.5 (D_s=0.00481)	0.0058	0.0765	0.0615
Δ =2 (D_s=0.0855)	0.0090	0.0950	0.0744
Δ =2.5 (D_s=0.01335)	0.0144	0.1202	0.0924
Δ =3 (D_s=0.01923)	0.0198	0.1407	0.0983
Δ =5 (D_s=0.05342)	0.0687	0.2660	0.1909
Δ =8 (D_s=0.13675)	0.1612	0.4015	0.2663
Δ =10 (D_s=0.21367)	0.1886	0.4343	0.3304

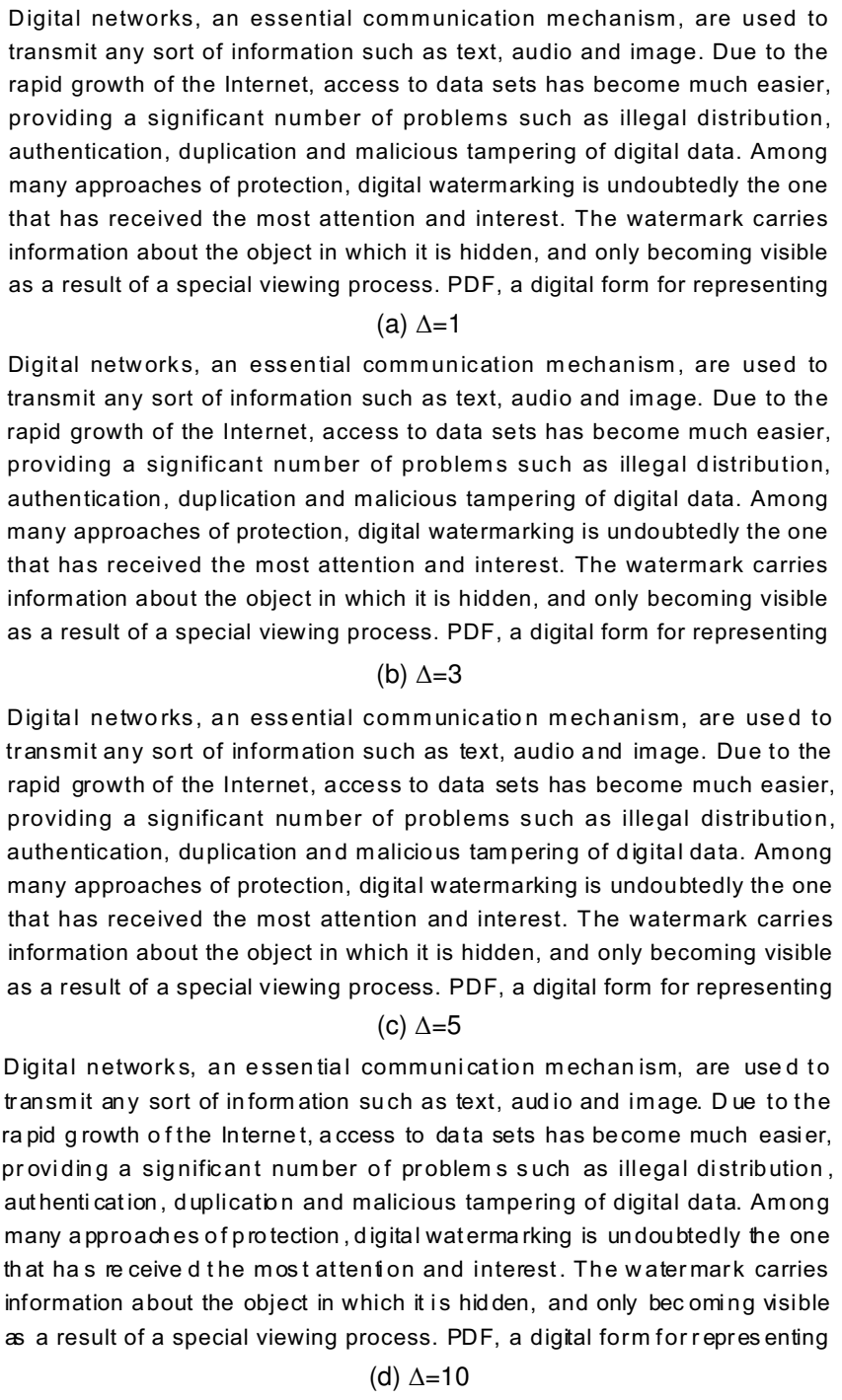


Figure 3.14: Perceptual visualization of the watermarked document using the STDM for Δ=1, Δ=3, Δ=5 and Δ=10 gradually from top to bottom

ferent metrics: Mean Absolute Error (MAE), Root Square Error (RSE), and Mean Square Error (MSE). The watermarked documents under different values of Δ are presented in Figure 3.14. As shown in Table 3.2 and Figure 3.14, a small value of Δ gives a slight

modification in the position of the characters, and a notable modification with a higher value of Δ that decreases the transparency level.
Based on the experiments, a perceptual difference between the watermarked and original documents can be noticed for an average distortion Ds greater than 0.01923. Thus the transparency threshold t is equal to 0.01923.

3.5.2.2/ TESTS OF ROBUSTNESS

The robustness experiments were conducted by applying the Gaussian and Salt&Pepper watermarking attacks to the x-coordinates of the characters in the watermarked docu-

Table 3.3: Tests of robustness against Gaussian attack

Δ (Distortion)	Variance	BER	Corr
Δ= 0.1 (D_s=0.00002)	0.1	11.584	0.0267
	0.25	11.520	0.0300
Δ= 0.5 (D_s=0.00053)	0.1	11.420	0.0163
	0.25	11.598	0.0260
Δ=1 (D_s=0.0241)	0.1	8.3820	0.2899
	0.25	10.530	0.1139
Δ= 1.5 (D_s=0.00481)	0.1	4.3080	0.6351
	0.25	7.1760	0.3944
Δ= 2 (D_s=0.00855)	0.1	**1.9200**	0.8376
	0.25	4.2860	0.6821
Δ= 2.5 (D_s=0.01335)	0.1	1.1320	0.9045
	0.25	**2.9160**	0.7585
Δ= 3 (D_s=0.01923)	0.1	0.3420	0.9710
	0.25	1.2680	0.8927
Δ= 4 (D_s=0.03419)	0.1	0.0680	0.9943
	0.25	0.4100	0.9657
Δ= 5 (D_s=0.05342)	0.1	0.0060	0.9995
	0.25	0.0640	0.9946
Δ= 6 (D_s=0.07692)	0.1	0	1
	0.25	0.0080	0.9993
Δ= 7 (D_s=0.10470)	0.1	0	1
	0.25	0.0040	0.9997
Δ= 8 (D_s=0.13675)	0.1	0	1
	0.25	0	1
Δ= 9 (D_s=0.17307)	0.1	0	1
	0.25	0	1
Δ= 10 (D_s=0.21367)	0.1	0	1
	0.25	0	1

ment, under two variances and densities (0.1 and 0.25). Therefore, different robustness threshold levels are computed respectively from the experiments of each type of attacks. The thresholds r_1 and r_2 are computed from the Gaussian experiments under variances equal to 0.1 and 0.25, respectively. The thresholds r_3 and r_4 are computed from the Salt&Pepper experiments under densities equal to 0.1 and 0.25. The robustness threshold level r corresponds to the best robustness level under the mentioned watermarking attacks. Noting that, the digits after the decimal point are modified under such types of attacks.
The Bit Error Rate (BER) and Pearson's linear correlation coefficients (Corr) were used to evaluate the robustness level, by comparing the original watermark to the extracted one,

Table 3.4: Tests of robustness against Salt&Pepper attack

Δ (Distortion)	Variance	BER	Corr
Δ= 0.1 (D_s=0.00002)	0.1	11.184	0.0583
	0.25	11.468	0.0312
Δ= 0.5 (D_s=0.00053)	0.1	9.3900	0.2074
	0.25	11.434	0.0375
Δ=1 (D_s=0.0241)	0.1	3.6120	0.6932
	0.25	8.5360	0.2802
Δ= 1.5 (D_s=0.00481)	0.1	**1.0480**	0.9109
	0.25	4.2480	0.6389
Δ= 2 (D_s=0.00855)	0.1	0.2880	0.9756
	0.25	**1.9880**	0.8314
Δ= 2.5 (D_s=0.01335)	0.1	0.0640	0.9946
	0.25	0.7120	0.9394
Δ= 3 (D_s=0.01923)	0.1	0.0100	0.9992
	0.25	0.3180	0.9729
Δ= 4 (D_s=0.03419)	0.1	0	1
	0.25	0.0340	0.9971
Δ= 5 (D_s=0.05342)	0.1	0	1
	0.25	0.0020	0.9998
Δ= 6 (D_s=0.07692)	0.1	0	1
	0.25	0	1
Δ= 7 (D_s=0.10470)	0.1	0	1
	0.25	0	1
Δ= 8 (D_s=0.13675)	0.1	0	1
	0.25	0	1
Δ= 9 (D_s=0.17307)	0.1	0	1
	0.25	0	1
Δ= 10 (D_s=0.21367)	0.1	0	1
	0.25	0	1

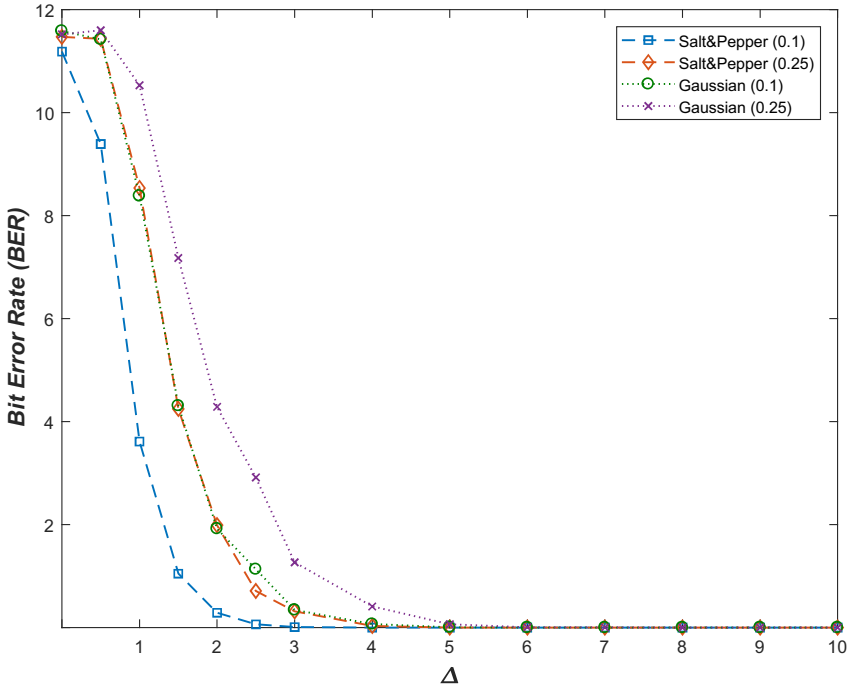


Figure 3.15: Comparisons between Salt&Pepper and Gaussian noise attacks in terms of BER

to make the objective performance comparison. In this test, a total of k=24 bits are embedded into the x-coordinates values, and the simulations were repeated 500 times. Let's consider that 12.5% of BER can be authorized to deduce the threshold values. That's means a total of 3 error bits from 24 can be corrected by the majority of error-correcting codes. Therefore, the threshold level would be equal to the distortion value D_s from which the error bits are inferior or equal to 3.
The robustness tests against Gaussian and Salt&Pepper noise attacks are presented in Tables 3.3 and 3.4, respectively. We notice from Table 3.3 that for D_s≥0.00855, the average BER is less than 3 when the variance equal is to 0.1, and for D_s≥0.01335, the average BER is less than 3 when the variance is equal to 0.25. Therefore, r_1 and r_2 are equal to 0.00855 and 0.01335, respectively. We notice from Table 3.4 that for D_s≥ 0.00481, the average BER is less than 3 when the density is equal to 0.1, and for D_s≥0.00855, the average BER is less than 3 when the variance equal to 0.25. Therefore, r_3 and r_4 are equal to 0.00481 and 0.00855, respectively. The mentioned thresholds are highlighted in Bold in Tables 3.3 and 3.4.
The BER and correlation results are also presented in Figures 3.15 and 3.16, by the function of Δ. We can notice that the embedded bits are more robust against the Salt$Pepper noise attack comparing to the Gaussian noise attack. The robustness level increases with a high value of Δ, and vice versa.

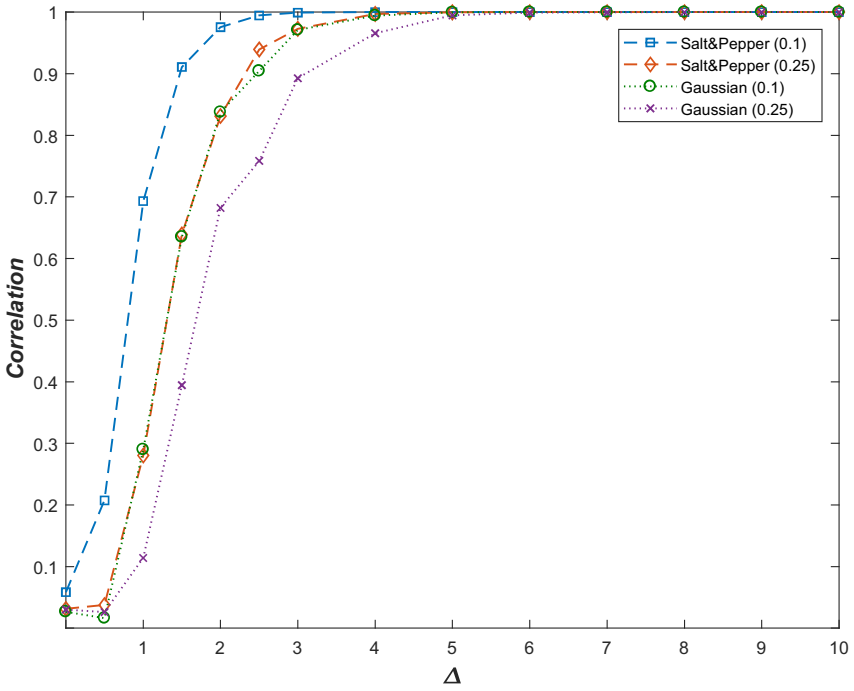


Figure 3.16: Comparisons between Salt&Pepper and Gaussian noise attacks in terms of Correlation

3.5.2.3/ TRANSPARENCY WITH ROBUSTNESS

Robustness and transparency experiments were performed to deduce the best approximation values of the threshold levels r and t. Therefore, sufficient robustness and transparency are achieved with an interval acceptable distortion $r \leq D_s \leq t$.
In the transparency tests, we have found that for $D_s \leq 0.01923$, the perceptual distortion between the original and watermarked documents cannot be noticed. For that, $t = D_s = 0.01923$ ($\Delta = 3$).
In the robustness tests, we have noticed that the acceptable distortion with Salt&Pepper noise attack is $D_s \geq 0.00855$, and $D_s \geq 0.01335$ with Gaussian noise attack. Therefore $r = D_s = 0.01335$ ($\Delta = 2.5$).
The final acceptable distortion that belongs to the interval [0.01335 0.01923] shows sufficient transparency and robustness against the Salt&Pepper and Gaussian noise attacks.

3.6/ CONCLUSION

In this chapter, we have provided an overview of the Portable Document Format (PDF). We have shown in detail the PDF components, the PDF File structure, and the text state parameters in a PDF document. After that, we have presented several proposed methods for Digital watermarking and steganography in Text and PDF documents.
The x-coordinates values of the characters in a PDF document are non-constant and could be exploited to embed the watermark. Therefore, we have presented a blind digi-

tal watermarking technique for PDF document that is based on the STDM watermarking scheme. The experiment results show that an acceptable distortion value provides sufficient robustness against noise attacks while preserving a good level of transparency.

III

CONTRIBUTION

4

SECURITY AGAINST PCA AND ICA ATTACKS

4.1/ INTRODUCTION

In the past chapter, we have presented a blind digital watermarking scheme for PDF documents that consist of embedding the watermark in the x-coordinates values of a group of characters using the STDM method. This method is applied with a trade-off between robustness and transparency, which are the main properties of digital watermarking in addition to payload and security.
STDM is a blind watermarking scheme that achieved high robustness against requantization and random noise attacks. It has been applied mainly to images, videos, audios, and PDF documents. STDM watermarking scheme spread the watermark bits into sample vectors using a projection vector, which is a secret key used during the embedding and decoding process. Knowing the projection vector by an adversary leads to a security problem. The observation of several watermarked signals can provide sufficient information for an adversary to estimate the projection vector using Blind Source Separation (BSS) techniques.
Bass and Hurri [41] show that STDM can be attacked successfully using a BSS technique called Independent Component Analysis (ICA), by estimating the projection vector p of STDM used during the embedding process. Cao [83] proved that an attacker can estimate the projection vector p using another BSS technique called Principal Component Analysis (PCA).
This chapter corresponds to an article published in SECRYPT [103] that presents the CAR-STDM (Component Analysis Resistant-STDM) for blind PDF watermarking that increases the security against PCA and ICA attacks. The following of the chapter is organized as follows. Section 4.2 recalls some backgrounds on BSS attacks. The proposed method is presented in Section 4.3. The evaluation of the proposed approach is presented in section 4.4. Finally, Section 4.5 concludes the chapter.

4.2/ BLIND SOURCE SEPARATION TECHNIQUES

Besides imperceptibility and robustness, security is an important requirement for watermarking schemes. While increasing the security, we guarantee that the embedded watermark is safe against the opponent, whose aim is to estimate the private key.

4.2.1/ PCA ATTACK

A BSS technique called Principal Component Analysis (PCA) could be used by an attacker to estimate the projection vector p as done in [83]. PCA identifies a smaller number of uncorrelated variables known as principal components from a complex data set. It is a variable reduction procedure, which is useful to be applied to redundant variables. Reducing the observed variables into a smaller number of principal components will account for most of the variance in the observed variables. Therefore, we could extract the relevant information from the complex data, and this is achieved by computing the eigenvector with the highest eigenvalue. A helpful strategy for an opponent is the Known Original Attack (KOA). In such a case, the original signal is known, and the opponent's goal is to gain information about the structure of the secret key, to hack later on several watermarked signal. In this way, the attacker will get the quantization error q' presented in equation (4.5) and will try to estimate the projection vector p using PCA.
PCA provides a roadmap to extract relevant information from a complex data set. The spread between numbers in a data set X is measured using the variance, which is the original statistical measure of the deviation of variables from its mean $\bar{X}$ in one dimension.

$$\sigma^2 = \frac{\sum_{i=1}^{n}(X_i - \bar{X})^2}{n} \tag{4.1}$$

The covariance is a measure of how much two-dimensional data set X and Y change together, to see if there is a relationship between those sets of variables.

$$cov(X,Y) = \frac{\sum_{i=1}^{n}(X_i - \bar{X})(Y_i - \bar{Y})}{n} \tag{4.2}$$

The value and the sign of the covariance are important during the estimation, noting that $cov(X,Y)$ is equal to $cov(Y,X)$. When the covariance is zero, it indicates that the two sets of variables X and Y are independent. If the covariance is negative, it indicates that one of the sets of variables increases while the other one decreases, or vice-versa. In the opposite case, a positive covariance would indicate that both sets of variables increase or decrease together.
The covariance calculation is used to find the relationships between high dimensional data sets. m-dimensional data sets will result in a $m \times m$ covariance matrix. For example, suppose that we have 3 data sets X, Y, and Z, hence the covariance matrix C is calculated as follow:

$$C = \begin{bmatrix} cov(X,X) & cov(X,Y) & cov(X,Z) \\ cov(Y,X) & cov(Y,Y) & cov(Y,Z) \\ cov(Z,X) & cov(X,X) & cov(Z,Z) \end{bmatrix} \tag{4.3}$$

A scalar value λ is called an eigenvalue of the squared matrix C if:

$$\begin{aligned} C \cdot e &= \lambda \cdot e \\ C \cdot e - \lambda \cdot I \cdot e &= 0 \\ (C - \lambda I) \cdot e &= 0 \end{aligned} \tag{4.4}$$

Where e is the eigenvector corresponding to the eigenvalue λ, and I is the identity matrix. Where for each eigenvector e there is only one associated eigenvalue λ, therefore λ is calculated such that the determinant $|C - \lambda I|$ is equal to zero.
By substituting the eigenvalue in equation (4.4), we will find the eigenvector.

The principal component of the data sets is the eigenvector with the highest eigenvalue. And in our case, this principal component will be the estimated projection vector $\hat{p}$.
The following are the main steps of the PCA attack in the proposed method:

1. Extract the quantization error q' from the watermarked document and rearrange it into a matrix J of size $L \times N$, where N represents the number of embedded bits, and L represents the length of the projection vector p.
2. Calculate the covariance of J, through which we will get the eigenvectors E. The last column of these eigenvectors is associated with the largest eigenvalue, which corresponds to the estimated projection vector $\hat{p}$.
   By doing so, we obtain the estimated projection vector of length L, denoted $\hat{p}$. Algorithm 1 details how this projection vector is recomputed.

Algorithm 1 PCA Algorithm

```
Matrix J = (Matrix)(new Matrix(q'))
      .covariance
      .eig()
      .getV();
for (idx = 0; idx < N; idx ++) do
   p̂[idx] = (B.getArrayCopy())[idx][N − 1];
end for
```

STDM spreads the distortion into a sample vector x based on the projection vector p, as shown in equation (2.14). In this way, the watermark energy is focused on the projection vector after the STDM embedding. Equation (2.14) can be seen as the host signal x augmented with the quantization error q':

$$q' = \left(round\left(\frac{x^T p - d_m}{\Delta} \right) \Delta + d_m - x^T p \right) p \tag{4.5}$$

The quantization error q' makes the variables correlated because the same projection vector p is used during the embedding process. Using PCA, we could be able to estimate the projection vector p. In this way, the attacker will be able to estimate the projection vector and will get the possibility to copy, modify, or remove the watermark.

4.2.2/ ICA ATTACK

An attacker may also estimate the projection vector p of STDM using another blind source separation technique called Independent Component Analysis (ICA), which is a computational and statistical technique for recovering a set of independent signals from sets of random variables or signals by some simple assumptions of their statistical properties [13, 33]. What distinguishes ICA from PCA is that it recover the signals such that they are non-Gaussian and statistically independent. The individual signals will be independent if we break the Gaussian observation down into a set of non-Gaussian mixtures, each with distributions that are non-Gaussian as possible.
ICA technique could be used to estimate a mixing matrix A and the independent sources X from a set of watermarked contents $\hat{Y}$ using the matrix formulation:

$$\hat{Y} = AX \tag{4.6}$$

While using FastICA, which is a popular ICA algorithm that achieves a fast operation and reliability of the extracted basis vector [41], we are able to compute the matrices A and X and extract the estimated projection vector $\hat{p}$ located in one of the columns of the matrix A.
A limitation of the ICA technique is the fact that the secret carrier could be estimated up to sign, but this will be solved by multiplying the independent components by -1 without affecting the model. Another ambiguity of ICA is that the estimated independent components may appear in an arbitrary (column) order, but one of them would be the proper estimated secret carrier.
We could present the principle of ICA using a simple example. Let's consider a situation where there are 3 peoples in the same room speaking simultaneously. Assume further that there exist 3 microphones in different locations. Denote by $x_1(t)$, $x_2(t)$ and $x_3(t)$ the original signals, and by $o_1(t)$, $o_2(t)$ and $o_3(t)$ the observed signals. We will get linear equations:

$$\begin{aligned} o_1(t) &= a_{11}x_1(t) + a_{12}x_2(t) + a_{13}x_3(t) \\ o_2(t) &= a_{21}x_1(t) + a_{22}x_2(t) + a_{23}x_3(t) \\ o_3(t) &= a_{31}x_1(t) + a_{32}x_2(t) + a_{33}x_3(t) \end{aligned} \tag{4.7}$$

Which can be represented as:

$$\begin{pmatrix} o_1(t) \\ o_2(t) \\ o_3(t) \end{pmatrix} = \begin{pmatrix} a_{11} & a_{12} & a_{13} \\ a_{21} & a_{22} & a_{23} \\ a_{31} & a_{32} & a_{33} \end{pmatrix} \begin{pmatrix} x_1(t) \\ x_2(t) \\ x_3(t) \end{pmatrix} \tag{4.8}$$

Where, A is the unkown mixing matrix.
Using ICA, we will recover the original signals from the mixtures, while finding the unmixing matrix W, that will give the best possible approximation of X.

$$\begin{aligned} x'_1(t) &= w_{11}o_1(t) + w_{12}o_2(t) + w_{13}o_3(t) \\ x'_2(t) &= w_{21}o_1(t) + w_{22}o_2(t) + w_{23}o_3(t) \\ x'_3(t) &= w_{31}o_1(t) + w_{32}o_2(t) + w_{33}o_3(t) \end{aligned} \tag{4.9}$$

To do so, The main steps of ICA are:

1. Centering the variables.
2. Whitening.
3. Measuring non-gaussianity.

To simplify and improve the algorithm of ICA, it is interesting to make a preprocessing of data. The first operation is the centering of the observable variables by subtracting their sample mean:

$$O = O - E\{O\} \tag{4.10}$$

Whitening is the second operation of preprocessing, which will reduce the dimensionality of the data and improves the convergence of the algorithms. It consists of de-correlating the mixtures. Practically, to pass from O to F, we introduce the whitening matrix R, such that:

$$F = RO \tag{4.11}$$

Algorithm 2 ICA Algorithm

Choose ϵ;
Choose a random weight vector $W(0)$;
$W(1) = E(Fg(W^T(0)F))–E(g'(W^T(0)F))W(0)$;
$W(1) = W(1)/\|W(1)\|$;

while $(1 - |W^T(k)W(k-1)| \geq \epsilon)$ **do**
  k=k+1;
  $W(k) = E(Fg(W^T(k-1)F))–E(g'(W^T(k-1)F))W(k-1)$;
  $W(k) = W(k)/\|W(k)\|$;
end while

F is obtained by PCA, presented in Section 4.2.1.
After that, we have to search the whitening separating matrix W^T to get the independent components $X' = W^T F$ while measuring the non-Gaussianity. In practice, we could use the negentropy, which is based on the information-theoretic quantity of entropy. The advantage of using negentropy as a measure of non-Gaussianity is that it is well justified by statistical theory[13]. The approximation is based on the maximum-entropy principle:

$$W = E(Fg(W^T F))–E(g'(W^T F))W \tag{4.12}$$

Where $g(y)$= tanh(ay) and $g'(y)$=a(1-tanh$^2(ay)$). $1 \leq a \leq 2$.
Algorithm 2 details how to get one independent component using the FastICA algorithm. The boucle is repeated until we get the convergence between the old and the new value of W, *i.e.* until we get a dot-product close or equal to 1.
As we will see in Section 4.4, the opponent can estimate the projection vector using PCA or ICA, therefore we applied a simple modification to the STDM watermarking scheme to overcome such kind of attacks.

4.3/ Proposed CAR-STDM Method

The projection vector p is an essential part of the STDM watermarking method, that is used as a secret key to embed and extract the watermark. The observation of several watermarked signals can provide sufficient information for an attacker to estimate the projection vector, by using the PCA or ICA attacks. The proposed scheme takes into account the security constraint and tries to overcome such kind of BSS attacks.

4.3.1/ Embedding Process

The embedding process is divided into 4 main steps as shown in Figure 4.1. The main idea is to have a number of projection vectors equal to the number of the bits to be embedded.
The first step consists of reading the original document, extracting the x-coordinates c_i of the existing characters and rearranging them into a matrix X of size $N \times L$.
For the second step, we use one secret key k shared between the embedder and decoder

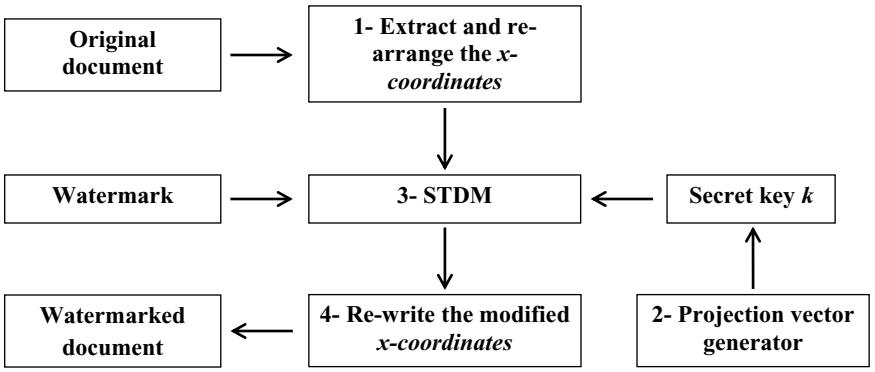


Figure 4.1: Embedding Block diagram

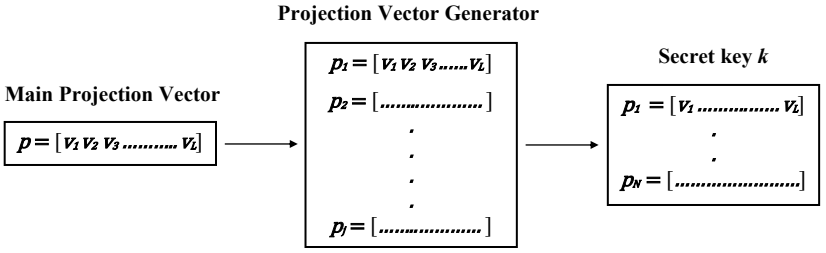


Figure 4.2: Projection Vector Generator

as a seed of a cryptographically secure pseudorandom number generator (PRNG) to produce L projection vectors. This step is further denoted as "Projection vector generator" and is illustrated in Figure 4.2.
The third step consists of embedding each bit of the watermark into the x-coordinates values using the STDM embedding function presented in equation (2.14). We will use one projection vector per embedded bit, for that we only have to replace p with p_i in equation (2.14). The last step consists of rearranging and re-writing the modified x-coordinate of each character into the watermarked document.
The embedding process is summarised in Figure 4.3. Where b is the binary representation of the secret message that contains N bits. Each bit of b will be embedded into the host signal X using a projection vector p_i of length L.
For instance, suppose that we have a PDF file containing 800 characters through which we will embed a watermark of length N=24 bits. If we consider that the length of projection vector L=8; in this case, we need 192 characters (24×8) to embed the watermark. Hence 192 characters will be selected from the original document, and the x-coordinates will be rearranged into a matrix X. 24 projection vectors will be extracted from the projection vector generator using the secret key k, and each bit will be projected into 8 x-coordinates using different projection vector.

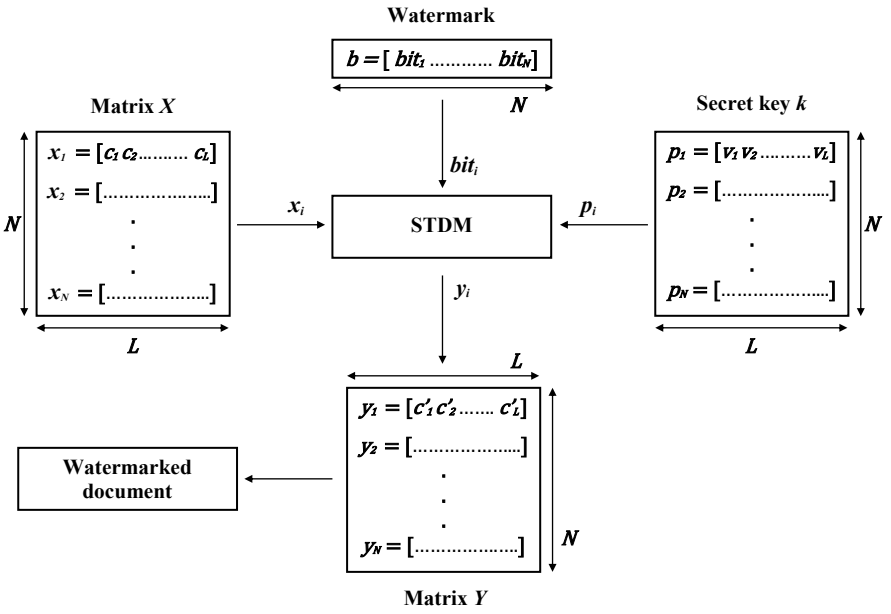


Figure 4.3: Embedding the ith bit of the secret message into a set of x-coordinates x_i using a projection vector p_i

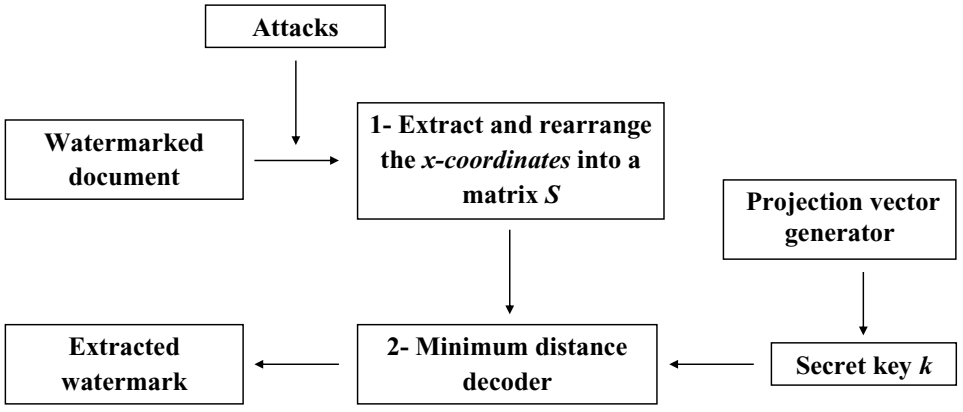


Figure 4.4: Extraction Block diagram

4.3.2/ EXTRACTION PROCESS

The extraction process is divided into 2 main steps as shown in Figure 4.4. The first step consists of reading the watermarked document and extracting the x-coordinates of the existing characters and rearranging them into a matrix S of size $L \times N$. Matrix S is the watermarked matrix Y subject to various attacks. The second step consists of extracting the embedded message by using the minimum distance decoder, based on the secret key k as of the form:

$$\hat{m}_i = \underset{m \in \{0,1\}}{\arg\min} \mid s^T p_i - Q_m(s^T p_i, \Delta) \mid \tag{4.13}$$

where s corresponds to a vector of x-coordinates of length L, and p_i represents the projection vector used to extract the i^{th} bit of the secret message.

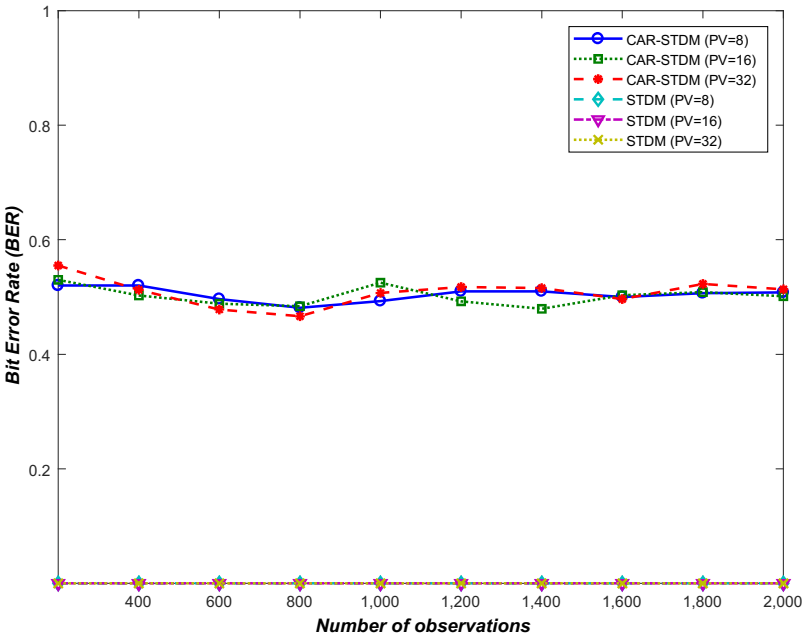


Figure 4.5: The comparison between the proposed CAR-STDM and the traditional STDM while modifying the length of the projection vector from 8 to 32 and the number of observations from 200 to 2000, while applying the ICA attack

4.4/ EVALUATION OF THE PROPOSED APPROACH

In this section, we compare the performance of the traditional STDM watermarking scheme and the proposed CAR-STDM watermarking scheme in terms of security (Section 4.4.1), transparency (Section 4.4.2), and robustness (Section 4.4.3). In a practical point of view, same embedding parameters have been used such as the length of the host signal and the secret message, dither level, and quantization step. All the experiments were conducted while varying the quantization step Δ, and the simulations were repeated 500 times. The embedded PRNG is BBS [1], Blum Blum Shub as an instance of cryptographically secure PRNG.

4.4.1/ SECURITY

PCA and ICA have been used to compare the security level of the traditional STDM watermarking scheme and the proposed CAR-STDM watermarking scheme. The security level is measured by comparing the Bit Error Rate (BER) of the extracted message to the original one after the estimation of the projection vector used at the encoder using the PCA and ICA attacks.
Experiments are done while modifying the length of the embedded message from 200 to 2000 bits, and the length of the projection vector from 8 to 32. The original document chosen during the experimental tests contains 64000 characters. BER has been used as the error measurement between the original watermark m and the extracted watermark $\hat{m}$. If the BER tends to 0, means that the extracted watermark is similar to the original one, which means that the opponent gets an accurate estimation of the projection vector. Figure 4.5 and Figure 4.6 show that the CAR-STDM achieves a high level of security,

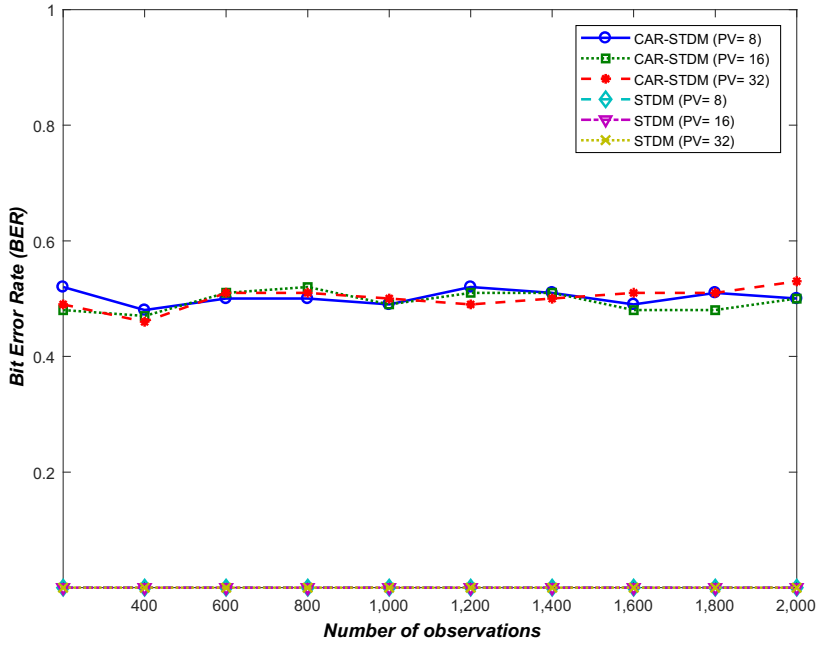


Figure 4.6: The comparison between the proposed CAR-STDM and the traditional STDM while modifying the length of the projection vector from 8 to 32 and the number of observations from 200 to 2000, while applying the PCA attack

where the BER of the extracted watermark is close to a random guess of 0.5. This high security level is due to the fact that multiple projection vectors are used during the embedding process instead of one, which nullifies the effect of PCA and ICA attacks to estimate the projection vectors, and prevents the opponent to copy, modify or remove the watermark.

In the traditional STDM method, a single projection vector is used during the embedding process, which decreases the security level. PCA and ICA can exactly estimate the projection vector due to the fact that the watermark energy is focused in the projection vector after the STDM embedding. Even if we vary the length of projection vector, the BER of the extracted watermark always tends to 0, which means that the opponent will get an accurate estimation of the projection vector through the PCA and ICA attacks. With the CAR-STDM, a unique projection vector p_i is generated to embed each i^{th} bit of the watermark, which will makes the watermarked vectors uncorrelated from each other, and void the effect of PCA. At the same time, we will not get any relevant information from the watermarked vectors, which will block the effect of ICA to estimate the projection vectors. Since we use a cryptographically secure PRNG for building each projection vector, any algebraic method will fail to estimate the projection vectors. The obtained practical results are coherent with this theoretical analysis.

While increasing the security of the watermarking scheme, we should preserve the efficiency in term of transparency and robustness. Therefore, we compared the CAR-STDM to the traditional STDM to make sure that the modification did not affect the efficiency of the main watermarking scheme.

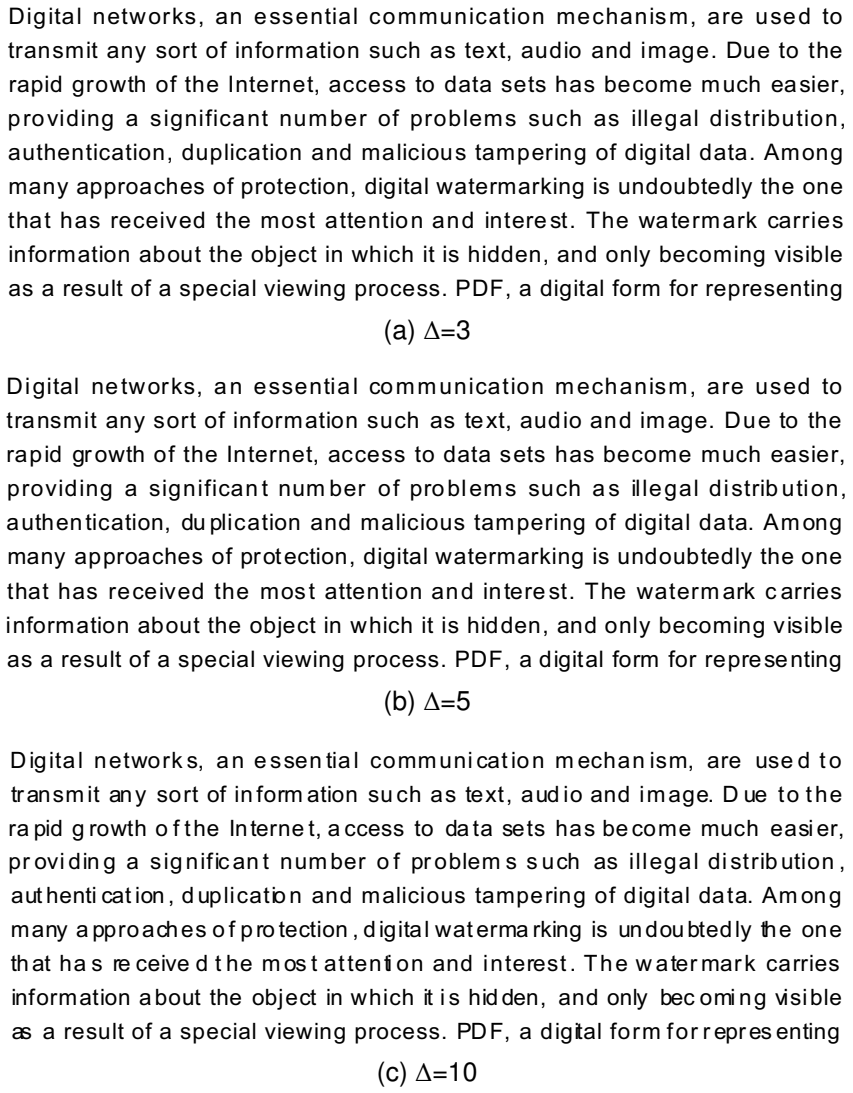


Figure 4.7: Perceptual visualization of the watermarked document using the proposed CAR-STDM for Δ=3, Δ=5 and Δ=10 gradually from top to bottom

4.4.2/ TRANSPARENCY

The transparency of the proposed CAR-STDM and the traditional STDM has been tested using several values of quantization step Δ. We consider a part of the original document containing 952 characters, and a watermark message ”LAW” is encoded using 8 bits per character in order to form a total of 24 bits. For that, the length of the projection vector used during the embedding and decoding concept is L=952/24 ≈ 39.
Table 4.1 presents the distortion plots of each value of Δ, and the error measurements between the watermarked document and the original one, using the Mean Square Error (MSE) and the average expected distortion D_s [21]. When Δ increases, the error values increase as well. As expected, the error values of the CAR-STDM and the Traditional STDM watermarking methods are very close to each other. As shown in Figure 4.7, we

Table 4.1: Distortion values when applying STDM and CAR-STDM

Quantization Step Size Δ	D_s	STDM (MSE)	CAR-STDM (MSE)
0.1	0.00002	2.2444×10^{-5}	2.3127×10^{-5}
0.5	0.00053	6.3574×10^{-4}	5.9178×10^{-4}
1	0.00214	0.0023	0.0020
1.5	0.00481	0.0058	0.0067
2	0.08550	0.0090	0.0087
2.5	0.01335	0.0144	0.0150
3	0.01923	0.0198	0.0199
5	0.05342	0.0687	0.0665
8	0.13675	0.1612	0.1618
10	0.21367	0.1886	0.1739

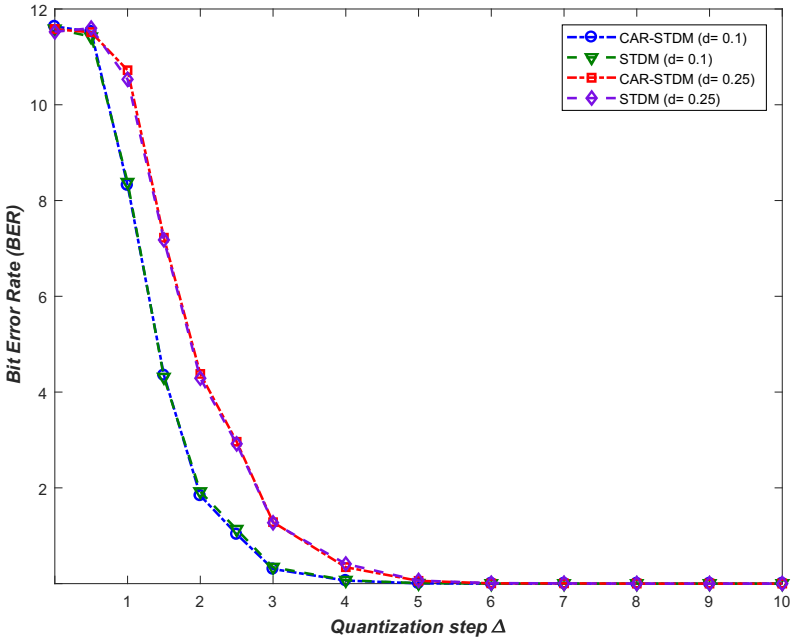


Figure 4.8: Comparison between the proposed CAR-STDM and the traditional STDM in terms of BER under Gaussian attack

get a slight modification in the character's position when Δ is smaller or equal to 3 and a notable modification when Δ is greatest than 3.

4.4.3/ Robustness against Gaussian and Salt&Pepper Noise

The robustness experiments were conducted by applying the Gaussian and Salt&Pepper watermarking attacks to the x-coordinates of the characters in the watermarked document with two density values (d = 0.1 and d = 0.25). Therefore, different robustness threshold levels are computed respectively from the experiments of each type of attacks. Only the

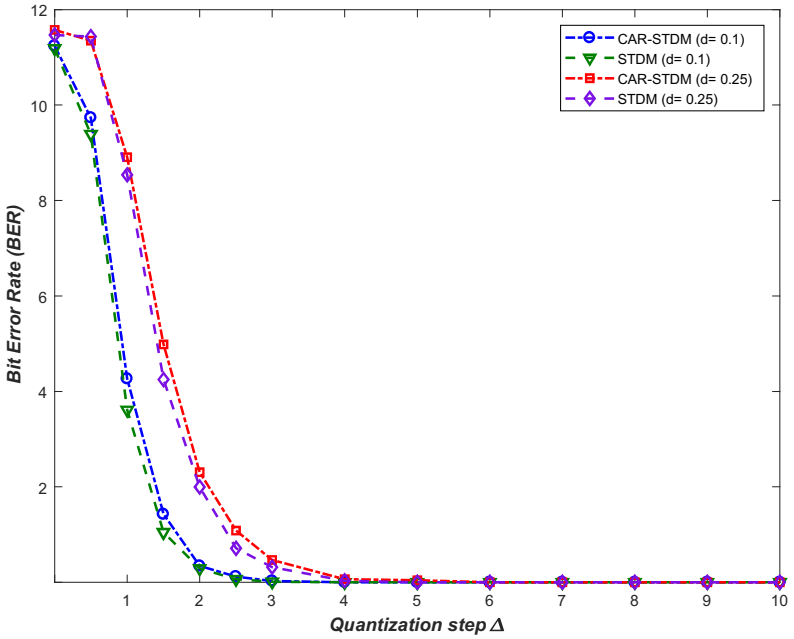


Figure 4.9: Comparison between the proposed CAR-STDM and the traditional STDM in terms of BER under Salt&Pepper attack

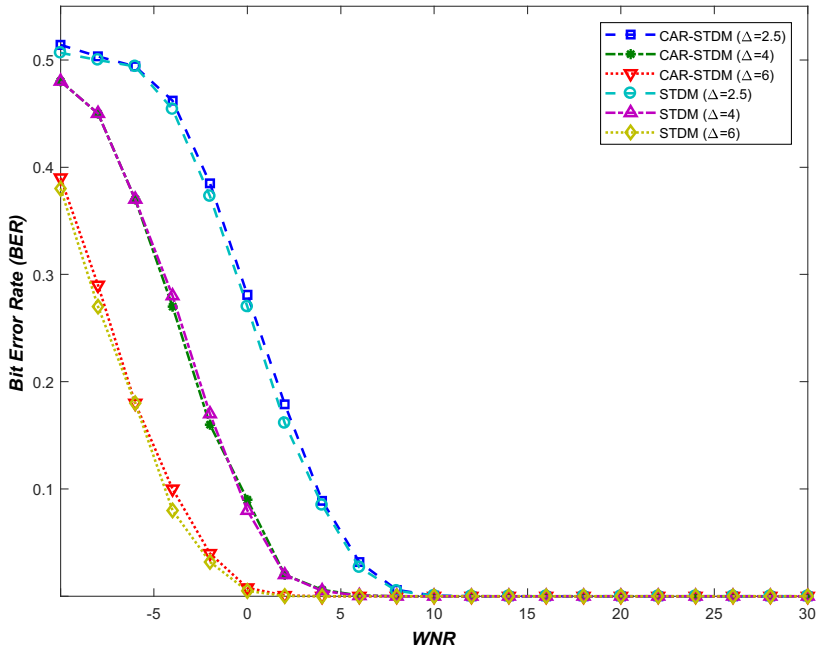


Figure 4.10: The robustness of the proposed CAR-STDM to that of STDM against the AWGN while varying the WNR

digits after the decimal point are modified. The Bit Error Rate (BER) was computed, by comparing the original watermark to the extracted one, to make the objective performance comparison. As shown in Figure 4.8 and Figure 4.9 the values of the proposed CAR-STDM and the traditional STDM are very close to each other and achieve a higher level of robustness. The robustness increases with a higher value of Δ. Figure 4.10 shows

the robustness of the CAR-STDM versus the traditional STDM under the AWGN attack, where the strength of the AWGN attack is evaluated by mean of the Watermark to Noise ratio (WNR):

$$WNR = 10\log_{10}(\frac{D_w}{D_c}), \tag{4.14}$$

where D_w represents the embedding distortion, and D_c denotes channel distortion. The BER assesses the robustness level of the watermarking technique. As shown in Figure 4.10, for Δ equal to 2.5, the BER is equal to 0 when the WNR is greater or equal to 10, and the BER is greater than 0.12 when the WNR is smaller than 4. Consequently, the robustness of the CAR-STDM and the traditional STDM increases while Δ increases.

4.4.4/ Our Method VS Related Work

Cao [83] proved that an attacker could estimate the projection vector of STDM, for that they proposed an improved method called ISTDM. In this method, the host signal is divided into two mutually orthogonal subspaces with the same dimension. The first one is called a reference subspace, and the other one is called an embedding subspace. ISTDM embed the watermark only in the embedding subspace using the STDM algorithm, in which the projection vector p is the projection of the normalized host signal onto the reference subspace. This method has been tested using a Gaussian-distributed with mean vector 0 as a host signal. In addition to that, the capacity has decreased, and only the half size of the host signal could be used to embed the watermark.
In contrast, our proposed watermarking scheme embeds the watermark in a PDF document for copyright protection under a sufficient transparency-robustness tradeoff while taking into account the security constraint. We exploited the x-coordinates values of characters as real cover elements to embed the watermark. Furthermore, CAR-STDM could not be attacked by any algebraic methods such as PCA and ICA and preserved the capacity of STDM, therefore the watermark could be embedded in all the cover elements.

4.5/ Conclusion

In this chapter, we have presented the proposed Component Analysis Resistant-STDM watermarking scheme. Unlike the traditional STDM watermarking, the CAR-STDM uses multiple projection vectors at the encoder and the decoder. The simple but effective idea is to produce one projection vector per an embedded bit thanks to a cryptographically secure PRNG, whose seed is a key shared between the encoder and the decoder.
The x-coordinates values of character PDF elements have been used as cover elements to embed the watermark, but any element can be used as support to contain the mark. Theoretically speaking, any algebraic attack such as PCA and ICA fails with this proposal. The experimental results confirm that the CAR-STDM approach achieves the security against such kind of BSS attacks, with higher level of transparency and robustness against AWGN and Salt&Peper attacks.

5

Robustness of STDM against Fixed Gain Attack

5.1/ Introduction

In the last chapters, we have presented a blind watermarking technique for copyright protection of PDF documents based on STDM, and we have improved the security of this technique against the Blind Source Separation techniques, such as PCA and ICA. STDM watermarking scheme achieves good robustness against varieties of attacks, such as the additive noise attacks, but it is largely vulnerable to the FGA attack [102, 106, 29]. In this type of attack, the received signal is indeed multiplied by a gain factor ρ, which scales the watermark vector and shifts it away from its original quantization cell. Therefore, the decoded watermark would be different from the embedded one. This is mainly due to the fact that when the host signal is scaled by the global factor ρ, the quantization step used for decoding is not scaled simultaneously. More formally, when the FGA attack is applied on the watermarked signal y, it becomes:

$$z = \rho \cdot y,$$

and the decoding message $\hat{m}$ presented in equation (2.15) will be decoded as follows:

$$\hat{m} = arg \min_{m \in \{0,1\}} \mid \rho \cdot y^T p - Q_m(\rho \cdot y^T p, \Delta) \mid .$$

The quantification factor $Q_m(\rho \cdot y^T p, \Delta)$ is indeed equal to $round\left(\frac{\rho \cdot y^T p - d_m}{\Delta}\right)\Delta + d_m$ which may be different from $round\left(\frac{y^T p - d_m}{\Delta}\right)\rho \cdot \Delta + \rho \cdot dm$ and consequently not equal to $\rho \cdot Q_m(y^T p, \Delta)$.
This chapter corresponds to an article published in Multimedia Tools and Applications [106], where we have modified the STDM watermarking scheme to resist the FGA attack, to enhance the robustness against other types of attacks such as AWGN attack and JPEG compression, and improve its effectiveness for images and PDF documents.
The rest of this chapter is organized as follows: Related Works are presented in Section 5.2. The proposed Normalize-STDM method is presented in Section 5.3. Section 5.4 provides a theoretical analysis of STDM and N-STDM. Section 5.5 presents the experiments on PDF documents. Experiments on real images are shown in Section 5.6. The findings and discussion are shown in Section 5.7. Finally, Section 5.8 provides our conclusion.

5.2/ Related Works

Many solutions for QIM and STDM watermarking schemes have been proposed using perceptual models based on Watson's model [2] to provide robustness against the FGA attack, we recall hereafter. Watson provided a perceptual model for computing the slack associated with each DCT coefficient within an 8×8 block. This model can be used to determines the acceptable data distortion, which is defined using the contrast masking, luminance masking, and frequency sensitivity table. The contrast masking, which is also referred to as slack, is given by:

$$s[i,j,k] = max(t_L[i,j,k], |C_0[i,j,k]|^{0.7} t_L[i,j,k]^{0.3}), \tag{5.1}$$

where $C_0[i,j,k]$ is the coefficient at the position (i,j) of the k^{th} block of the cover work, and $t_L[i,j,k]$ is the luminance masked threshold given as:

$$t_L[i,j,k] = t[i,j](C_0[0,0,k]/C_{0,0})^{0.649}, \tag{5.2}$$

where $C_0[0,0,k]$ is the DC coefficient of the k^{th} block, $t[i,j]$ is the smallest magnitude of the corresponding DCT coefficient in a block presented in a frequency sensitive table [47], and $C_{0,0}$ is the average of all the DC coefficient in the image. Likewise, $C_{0,0}$ could be fixed to a constant value that represents the mean intensity of the whole image.
Li *et al.* [35] have modified the QIM watermarking scheme based on Watson's perceptual model to resist the FGA attack. In this blind method, the luminance masking presented in equation (5.2) is modified as:

$$t_L[i,j,k] = t[i,j](C_0[0,0,k]/C_{0,0})^{0.649}(C_{0,0}/128), \tag{5.3}$$

and for each 8×8 block, Δ is selected based on the slack presented in equation (5.1) as:

$$\Delta_k = G \times \sum_{i=1}^{L} s_i, \tag{5.4}$$

where G is a constant factor that can be adjusted to alter the watermark strength. Thus, when the watermarked image is scaled by the global factor ρ, the luminance masking, the slack, and Δ_k are also scaled. In this way, the QIM algorithm is adapted to resist the FGA attack.
Similarly, Zhu, X. [46] proposed a value for Δ for each 8×8 block based on Watson's model during the embedding process of STDM as:

$$\Delta_k = 2L^{-\frac{1}{4}} D \sum_{i=1}^{L} s_i. \tag{5.5}$$

The slack s_i is computed using the contrast making presented in equation (5.1) at the position (i,j) of the k^{th} block. D is the perceptual distance between the watermarked signal y and the host signal x computed as:

$$D = \left(\sum_{i=1}^{N} \left| \frac{y_i - x_i}{s_i} \right|^4 \right)^{\frac{1}{4}}. \tag{5.6}$$

During the non-blind detection process, Δ is modified as:

$$\hat{\Delta}_k = \Delta_k \frac{C'_{0,0}}{C_{0,0}},$$

where $C_{0,0}$ and $C'_{0,0}$ are the mean of all the DC coefficients of the original and watermarked image. In this case, when the watermarked image is scaled by a factor ρ, $\hat{\Delta}_k$ will also be scaled, since that $C'_{0,0}/C_{0,0} \simeq \rho$.
In [48], Li *et al.* proposed a blind STDM-MW-SS watermarking scheme where the DCT transform is applied to the original image, and the slacks of each 8×8 block are similarly computed based on Watson's model to determine the projection vector and the quantization step size Δ. Given a length L vector of DCT coefficients and its corresponding vector of modified slacks, Δ_k is computed for each 8×8 block as presented in equation (5.4).
The projection vector is formed from the slack values of the DCT coefficients instead of the pseudo-random values.
Yu *et al.* [54] presented a blind Adaptive STDM (ASTDM) based on Watson's model. The slacks are computed during the embedding process as in equation (5.1), and the luminance masking is modified during the embedding and extraction process as:

$$t_L[i,j,k] = t[i,j] \left(\frac{C_0[0,0,k]}{C_{0,0}} \right)^{0.7} \left(\frac{C_{0,0}}{C'_{0,0}} \right),$$

Noting that during the embedding process, $C_{0,0}$ is equal to $C'_{0,0}$, and $C_{0,0}$ is stored in the watermarked image by combined duplication code and a parity check code.
For each 8×8 block, Δ_k is computed for the i^{th} bit of the watermark as:

$$\Delta_i = 2G|\bar{s}_i|\|v_i\|^2/|\bar{v}_i|,$$

where v_i is a private subvector, s_i presents a vector containing the slacks of the i^{th} 8×8 block, G is used to adjust the embedding strength, $|\bar{s}_i|$ and $|\bar{v}_i|$ are the absolute mean values of s_i and v_i, and $\|v_i\|$ is the l^2-norm of the vector v_i.
In the blind STDM-Step projection (STDM-SP) scheme that was proposed by Li *et al.* [67], Δ_k is similarly selected for each 8×8 block based on Watson's model. In this method, the magnitude $t[i,j]$ of the frequency sensitivity table [47] presented in equation (5.2) has been modified as:

$$t_m(i,j) = t(i,j) \times \frac{C_{0,0}}{\mu},$$

where $C_{0,0}$ is the mean intensity of the image and μ=512.
After that, the slack s of each 8×8 block is projected into a random vector p to gain the maximum imperceptible change of the host signal in the direction of p, which is selected as a positive projection vector to bypass the counteracting between the elements of s. Δ_k is given as:

$$\Delta_k = 2s^T p,$$

Wan *et al.* [78] proposed a Logarithmic STDM (LSTDM-WM) watermarking scheme based on the perceptual model. In this method, the projection of the host signal x onto a random vector p is transformed according to the logarithmic function presented in equation (5.7), and quantized into y_q as presented in equation (5.8).

$$F(x^T p) = \frac{ln(1 + \mu \frac{x^T p}{C_{0,0}})}{ln(1+\mu)}, \tag{5.7}$$

in which the parameter μ is selected based on:

$$\mu \ll \frac{C_{0,0}}{|x^T p|}.$$

$$y_q = Q(F(x^T p), \Delta, m, d_m), \quad m \in \{0, 1\}. \tag{5.8}$$

$Q(.)$ is the quantization presented in equation (2.14).
After that, the watermarked signal is obtained as follows:

$$y = \frac{C_{0,0}}{\mu}[(1+\mu)^{y_q} - 1] \tag{5.9}$$

To provide the robustness against FGA attack, they adapted Δ_k for each 8×8 block based on Watson's perceptual model as:

$$\Delta_k = \frac{ln(1 + 2s^T p \times \frac{\mu}{C_{0,0}})}{ln(1+\mu)}. \tag{5.10}$$

Jiang *et al*. [74] proposed a blind adaptive spread transform QIM (ST-QIM) watermarking algorithm based on improved perceptual models. They proposed four different implementations of perceptual modal and combined it with ST-QIM to form an adaptive quantization watermarking schemes. The best performance was for ST-QIM-fMW-SS and ST-QIM-MS-SS. In this part, Δ_k is selected as in equation (5.4). In ST-QIM-fMW-SS, the luminance masking was modified as:

$$t_L[i, j, k] = t[i, j](C_0[0, 0, k]/C_{0,0})^{0.649}(C_{0,0}/128). \tag{5.11}$$

In ST-QIM-MS-SS, the magnitude $t[i, j]$ of the frequency sensitivity table [47] presented in equation (5.2) has been modified as:

$$t_m(i, j) = t(i, j) \times C_{0,0}.$$

In [92], Wan *et al*. improved the logarithmic spread transform dither modulation presented in [78] using a robust perceptual model. They have introduced a new measurement of the edge, strength, and pixel intensity to calculate the slacks at the watermark embedder and watermark detector. Δ_k is selected for each 8×8 such as in equation (5.10).
Those modified watermarking schemes are applied in the frequency domain and could only be implemented on images to resist the FGA attack since they are dependent on the luminance and contrast making of images. Most of the proposed methods are studied based on the DCT transform, *i.e*. the DCT coefficients are quantized rather than the pixel values.
A different watermarking scheme, Rational Dither Modulation (RDM), has been proposed by Perez-Gonzalez *et al*. [38] in which the feature signal for quantization is constructed using the ratio of the previously generated watermarked sample and the current host sample as:

$$y_k = g(y_{k-L}^{k-1}) Q_m \left(\frac{x_k}{g(y_{k-L}^{k-1})} \right), \tag{5.12}$$

where Q_m represents the standard quantization operation, y_{k-L}^{k-1} denotes the set of past signals $(y_{k-L} \ldots y_{k-1})$, and the function $g(.)$ has the property that for any gain factor $\rho > 0$:

$$g(\rho y) = \rho g(y).$$

The function $g(.)$ include the L_p vector-norm:

$$g(y_{k-L}^{k-1}) = \left(\frac{1}{L} \sum_{i=1}^{L} |y_{k-i}|^p \right)^{\frac{1}{p}}.$$

The decoding is performed by using the minimum euclidean distance rule as:

$$\hat{m} = arg \min_{m \in \{-1,1\}} \left| \frac{z_k}{g(z_{k-L}^{k-1})} - Q_m \left(\frac{z_k}{g(z_{k-L}^{k-1})} \right) \right|. \tag{5.13}$$

5.2.1/ CONTRIBUTION

The QIM and STDM watermarking schemes that are based on Watson's perceptual model to resist the FGA attack are dedicated to image watermarking and could not be applied to other types of signals such as PDF documents. As for the RDM watermarking scheme, it achieves a good performance against the FGA attack with a certain limitation against the additive noise attack. RDM does not benefit from the property of the spreading vector, and the quantization step size is a variable step quantizer, whose size is a function of several past watermarked samples. For that, the attacking noise has more influence on the decoding quantization step size.
We have modified the traditional STDM watermarking scheme by making the quantization step dependent on the original samples during the embedding process and the watermarked samples during the decoding process to resist the FGA attack and enhance the robustness against AWGN attack, JPEG compression attack, and variety of filtering and geometric attacks. Moreover, we affirm that this approach could also be used as a blind watermarking scheme for PDF documents. We have applied our approach PDF documents and grayscale images in the spatial domain and frequency domain and compared its performance with other proposed methods. Our approach aims at being flexible as it is not dependent on the perceptual model to achieve the robustness against the FGA attack, and any element can be used as support to embed the watermark.

5.3/ PROPOSED N-STDM METHOD

The FGA problem will be solved if we scale the quantization step in the same way the watermarked signal is scaled. Obviously, it is difficult to straightforwardly estimate the global factor ρ, but if the quantization step size becomes dependent on the watermarked samples, it will be scaled concurrently with those samples. Therefore, in the proposed watermarking scheme, N-STDM, we have modified the embedding and decoding functions of the traditional STDM as illustrated in Figure 5.1. We have computed the norm value $||x||$ based on the host signal to be used during the embedding process as:

$$y = x + \left(||x|| Q_m \left(\frac{x^T p}{||x||}, \Delta \right) - x^T p \right) p \quad m \in \{0, 1\}. \tag{5.14}$$

$$||x|| = (|x_1|^{\frac{1}{u}} + |x_2|^{\frac{1}{u}} + ... + |x_n|^{\frac{1}{u}})^u, \tag{5.15}$$

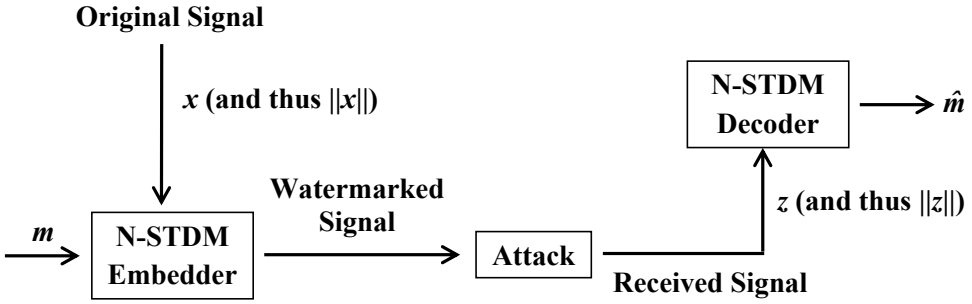


Figure 5.1: Block diagram of the N-STDM watermarking scheme

where n is the length of the extracted vector from the cover elements, $|x_n|$ is the absolute value of element x_n, and $\|x\|$ is a norm function that could be expressed as a l^2-norm when $u = 1/2$ and l^1-norm when $u = 1$ etc. (the influence of $\|x\|$ against the FGA attack while varying u is detailed in Section 5.6).
The detection is performed with a minimum distance decoder to extract the embedded message as follows:

$$\hat{m} = \underset{m' \in \{0,1\}}{\arg\min} \left| \tilde{y} - \|y\| \, Q_{m'} \left(\frac{\tilde{y}}{\|y\|}, \Delta \right) \right|. \tag{5.16}$$

where

$$\tilde{y} = y^T p. \tag{5.17}$$

Algorithm 3 N-STDM Embedder

Input: I, m, p
I: Original signal of size $n = M \times N$
m: Binary watermark of size $b = B_1 \times B_2$
p: Normalized projection vector of length $L = n/b$

Choose Δ and u;
$j = 0$;
$X \leftarrow$ reshape $(I, n, 1)$;
$W \leftarrow$ reshape $(m, b, 1)$;

for $i = L + 1 : L : n + 1$ **do**
  j++;
  $w = W(j)$;
  $x = X(i - L : i - 1)$;
  no = Power(Sum(Power$(x, 1/u)), u)$;
  q=Quantizer$(x^T p, w, \Delta)$;
  $y(i - L : i - 1) = x + (no * q - x^T p)p$;
end for
$Output = y^T$

As a result, when performed, the FGA attack s.t. $z = \rho.y$ will not affect the minimum distance decoder :

$$\begin{aligned}\hat{m} &= arg \min_{m\in\{0,1\}} \mid \tilde{z} - \|z\| Q_m\left(\frac{\tilde{z}}{\|z\|}, \Delta\right) \mid \\ &= arg \min_{m\in\{0,1\}} \mid \rho\cdot\tilde{y} - \rho\cdot\|y\| Q_m\left(\frac{\rho\cdot\tilde{y}}{\rho\cdot\|y\|}, \Delta\right) \mid \\ &= arg \min_{m\in\{0,1\}} \mid \rho\left(\tilde{y} - \|y\| Q_m\left(\frac{\tilde{y}}{\|y\|}, \Delta\right)\right) \mid . \end{aligned} \tag{5.18}$$

Therefore, the non-linear impact of the ρ factor in the quantization is now linear and consequently will not affect the process of decoding the message.
Algorithm 3 details the embedding process of a watermark into an image using the proposed N-STDM. The complexity is directly proportional to n; order of growth is n. In a worst-case scenario, the statement will be executed n times. The time complexity is linear O(n).

5.4/ THEORETICAL ANALYSIS OF STDM AND N-STDM

This section provides a theoretical proof of the correction of the STDM and N-STDM watermarking schemes. In other words, we have verified that when a message is embedded into a host signal and when there is no attack, the message would be extracted without any error.
The quantizer Q_m of the STDM watermarking scheme is given by:

$$Q_m(s,\Delta) = round\left(\frac{s-d_m}{\Delta}\right)\Delta + d_m, \tag{5.19}$$

where:

$$d_m = -\frac{\Delta}{4} + m.\frac{\Delta}{2}. \tag{5.20}$$

Let us first recall the definition of the rounding function.

$$round(x) = \begin{cases} \lfloor x + 0.5 \rfloor & \text{if } x \text{ is postive or null} \\ \lceil x - 0.5 \rceil & \text{otherwise} \end{cases}$$

In all what follows and without loss of generality, we consider $s - d_m$ to be positive. There exists $q_s \in \mathbb{N}$ and $r_s \in \mathbb{R}^+$, $0 \le r_s < \Delta$ such that:

$$\begin{aligned} s - d_m + \frac{\Delta}{2} &= q_s\Delta + r_s, \text{ or equivalently} \\ s - r_s + \frac{\Delta}{2} &= q_s\Delta + d_m. \end{aligned} \tag{5.21}$$

In such a case,

$$\begin{aligned} Q_m(s,\Delta) = round\left(\frac{s-d_m}{\Delta}\right)\Delta + d_m &= q_s\Delta + d_m \\ &= s - r_s + \frac{\Delta}{2}. \end{aligned} \tag{5.22}$$

Let $m \in \{0,1\}$ be the bit to be embedded into the host vector $x = (x_1, \ldots, x_n)$ with respect to the normalized projection vector $p = (p_1, \ldots, p_n)$ and a parameter Δ. Let y be the vector that contains the watermark.

5.4.1/ CORRECTION PROOF OF STDM

In the original STDM algorithm:

$$y = x + \left(Q_m(x^T p, \Delta) - x^T p\right) p$$

$$\hat{m} = \arg\min_{m' \in \{0,1\}} \left| y^T p - Q_{m'}\left(y^T p, \Delta\right) \right|,$$

thanks to equation (5.22), $Q_m(x^T p, \Delta) = x^T p - r_{x^T p} + \frac{\Delta}{2}$ so that

$$y = x - (r_{x^T p})p + \frac{\Delta}{2} p,$$

this allows deducing

$$y^T p = x^T p - r_{x^T p} + \frac{\Delta}{2}.$$

If we evaluate $Q_{m'}\left(y^T p, \Delta\right)$ (Annex A) we will get:

$$\hat{m} = \arg\min_{m' \in \{0,1\}} \left| \left(q_{x^T p} + \frac{m - m'}{2} \right) - round\left(q_{x^T p} + \frac{m - m'}{2} \right) \right|$$

Therefore, we conclude that the extracted message $\hat{m}$ depends on m and m'. Obviously, if $m = m'$, we have to evaluate $q_{x^T p} - round(q_{x^T p})$ which is null since $q_{x^T p}$ is a natural number. Otherwise, *i.e.*, when m and m' are distinct, let us first suppose that $\frac{m - m'}{2} = 0.5$. In this case, $\left| q_{x^T p} + 0.5 - round(q_{x^T p} + 0.5) \right| = \left| q_{x^T p} + 0.5 - (q_{x^T p} + 1) \right| = 0.5$. The last case, *i.e*, when $\frac{m - m'}{2} = -0.5$ is similar and is thus omitted. The minimum value is thus obtained when $m = m'$. Without any attack, $\hat{m}$ is thus m.

5.4.2/ CORRECTION PROOF OF N-STDM

In the following equations, $\|.\|$ is consider as the norm of elements.

Theorem 5.4.1. *Let y be the watermarked host:*

$$y = x + \left(\|x\| \, Q_m\left(\frac{x^T p}{\|x\|}, \Delta \right) - x^T p \right) p.$$

Let $\hat{m}$ be the retrieved watermark bit, which is defined by:

$$\hat{m} = \arg\min_{m' \in \{0,1\}} \left| y^T p - \|y\| \, Q_{m'}\left(\frac{y^T p}{\|y\|}, \Delta \right) \right|. \tag{5.23}$$

If we evaluate $y^T p$ and $Q_{m'}\left(\frac{y^T p}{\|y\|}, \Delta \right)$ (Annex B) we will get:

$$\hat{m} = \arg\min_{m' \in \{0,1\}} \left| \left(\frac{\|x\|}{\|y\|} . q_{\frac{x^T p}{\|x\|}} + \frac{\|x\|}{\|y\|} . \frac{2m - 1}{4} - \frac{2m' - 1}{4} \right) - round\left(\frac{\|x\|}{\|y\|} q_{\frac{x^T p}{\|x\|}} + \frac{\|x\|}{\|y\|} \frac{2m - 1}{4} - \frac{2m' - 1}{4} \right) \right|.$$

Digital networks become an essential communication mechanism. They are used to transmit any sort of information like text, audio and image. Due to the rapid growth of the Internet, access to multimedia data has become much easier, but authors and data providers are reluctant to allow the distribution of their data in a network environment due to a significant number of problems such as illegal distribution, duplication, authentication and malicious tampering of digital data.

(a) $\Delta = 2 \times 10^{-3}$, $MSE = 0.03$

Digital networks become an essential communication mechanism. They are used to transmit any sort of information like text, audio and image. Due to the rapid growth of the Internet, access to multimedia data has become much easier, but authors and data providers are reluctant to allow the distribution of their data in a network environment due to a significant number of problems such as illegal distribution, duplication, authentication and malicious tampering of digital data.

(b) $\Delta = 3 \times 10^{-3}$, $MSE = 0.07$

Digital networks become an essential communication mechanism. They are used to transmit any sort of information like text, audio and image. Due to the rapid growth of the Internet, access to multimedia data has become much easier, but authors and data providers are reluctant to allow the distribution of their data in a network environment due to a significant number of problems such as illegal distribution, duplication, authentication and malicious tampering of digital data.

(c) $\Delta = 4 \times 10^{-3}$, $MSE = 0.13$

Digital networks become an essential communication mechanism. They are used to transmit any sort of information like text, audio and image. Due to the rapid growth of the Internet, access to multimedia data has become much easier, but authors and data providers are reluctant to allow the distribution of their data in a network environment due to a significant number of problems such as illegal distribution, duplication, authentication and malicious tampering of digital data.

(d) $\Delta = 5 \times 10^{-3}$, $MSE = 0.17$

Figure 5.2: Perceptual visualization of the watermarked document using the N-STDM for $\Delta = 2 \times 10^{-3}$ to 5×10^{-3} gradually from top to bottom

When m and m' are equal, $\frac{||x||}{||y||}.\frac{2m-1}{4} - \frac{2m'-1}{4}$ is close to 0; therefore, the $round()$ value will be close to the $round()$ value of $\frac{||x||}{||y||}.q\frac{x^T p}{||x||}$ and the global result will be close to 0, but when m and m' are distinct, $\frac{||x||}{||y||}.\frac{2m-1}{4} - \frac{2m'-1}{4}$ is close to ± 0.5. In this situation, the $round()$ value may be significantly different. Without any attack, $\hat{m}$ is the same as m.

5.5/ EXPERIMENTS AND COMPARISONS ON PDF DOCUMENTS

In this section, we compare the robustness of the proposed N-STDM watermarking scheme, the traditional STDM [21], and RDM [38] against FGA and AWGN attacks. To assure a wider comparison, two scenarios have been applied through which the number of bits b and the length of the projection vector L have been variably used: In the first

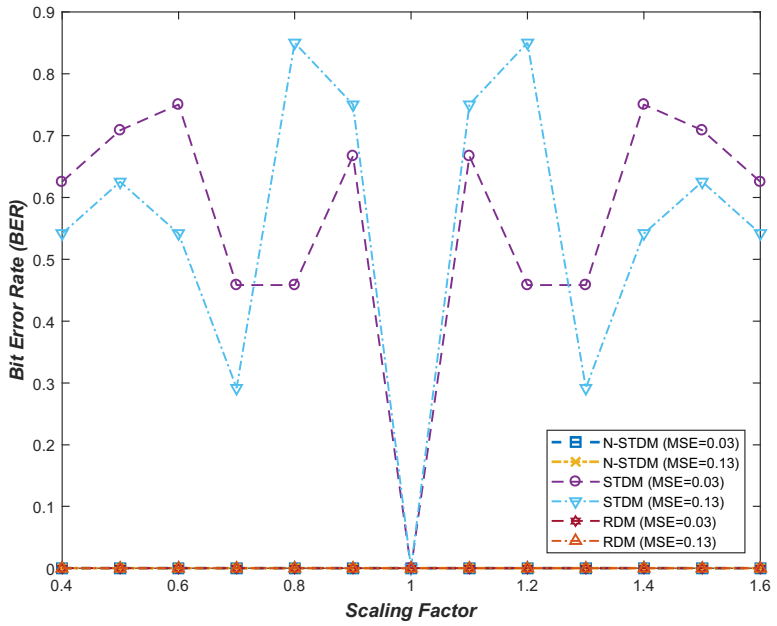


Figure 5.3: Robustness against FGA for b=24 and L=84

scenario, we embedded $k = 1$ bit per line (b=24 and L=84), and in the second one, we embedded $k = 4$ bits per line (b=96 and L=21). The same embedding parameters have been used such as the length of the host signal, the secret message, and dither level. All the experiments were conducted while varying Δ. Several values of Δ have been used taking into account the Mean Squared Error (MSE) values; the Δ value has been adjusted in order to have the same MSE values (0.03 and 0.13).
Each character of the watermark has been encoded into 8 bits in order to form a total of b bits. Each bit message is then embedded into L characters' x-coordinates extracted from the original document. Accordingly, a total of $b \times L$ characters are used from the document to embed the whole bits of the message. Figure 5.2 presents a part of a watermarked paragraph extracted from the watermarked document, which has been watermarked using the N-STDM watermarking scheme with several values of Δ and u=2. A number of 24 lines have been extracted from a PDF document with 84 characters each (n=84), totalling 2016 characters. When Δ increases, the error values increase as well, especially when $\Delta \geq 4 \times 10^{-3}$.

5.5.1/ Comparison Against FGA and AWGN Attacks

In order to prove the superiority of N-STDM in terms of robustness constraint, a comparison between the proposed scheme (N-STDM) and the traditional schemes (RDM and STDM) was performed while varying the parameters b and L. The comparison mainly tackles the robustness against FGA and the robustness against AWGN attacks.
The N-STDM robustness against the FGA attack, to start with, has achieved a great superiority over STDM. Figure 5.3 and Figure 5.4 show that the traditional STDM is affected by the FGA attack even when the Δ value is high (MSE=0.13) and regardless of the length of projection vector and the number of the embedded bits. On the contrary, N-STDM, along with RDM, surpass the FGA attack for MSE = 0.03 and 0.13.
As to the robustness against AWGN attack, the strength is evaluated by mean of the

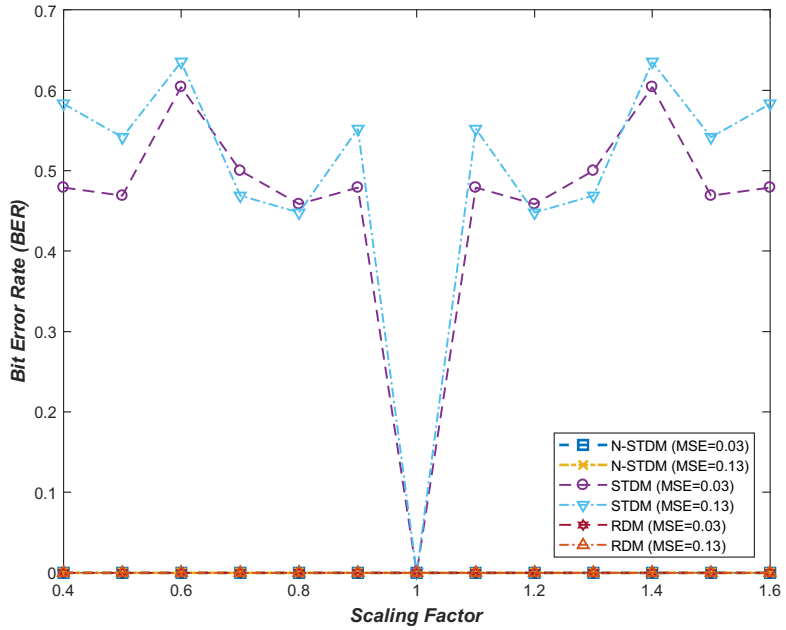


Figure 5.4: Robustness against FGA for b=96 and L=21

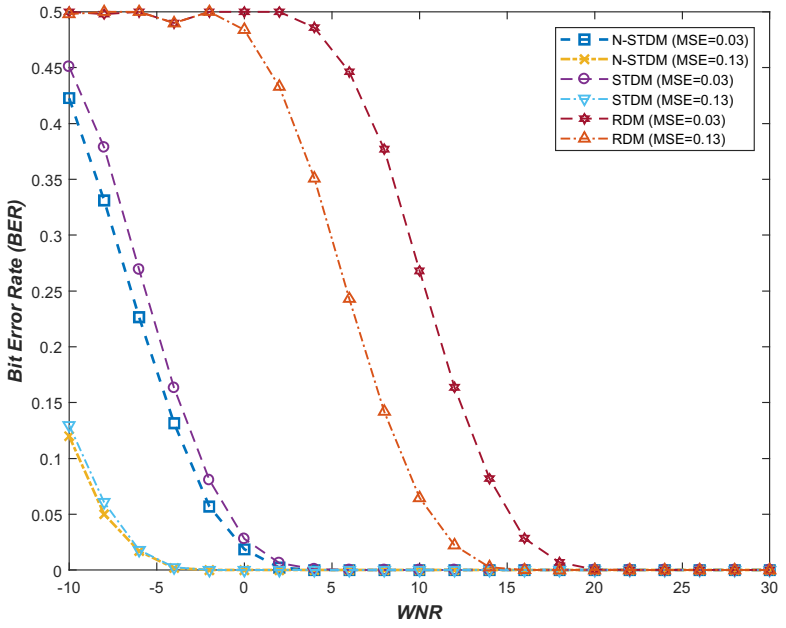


Figure 5.5: Robustness against AWGN for b=24 and L=84

Watermark to Noise Ratio (WNR) presented in equation (4.14). Only the digits after the decimal point are modified. Figure 5.5 and Figure 5.6 show that RDM is noticeably affected by the AWGN attack even when the Δ value is high whereas the proposed N-STDM achieves great performance against this attack. STDM preserves good performance against AWGN, yet the proposed N-STDM could perform even better noting that the BER of STDM and N-STDM are close to each other with an advantage of N-STDM over STDM.

As shown in Figure 5.5 and Figure 5.6, RDM is affected by the AWGN attack even with

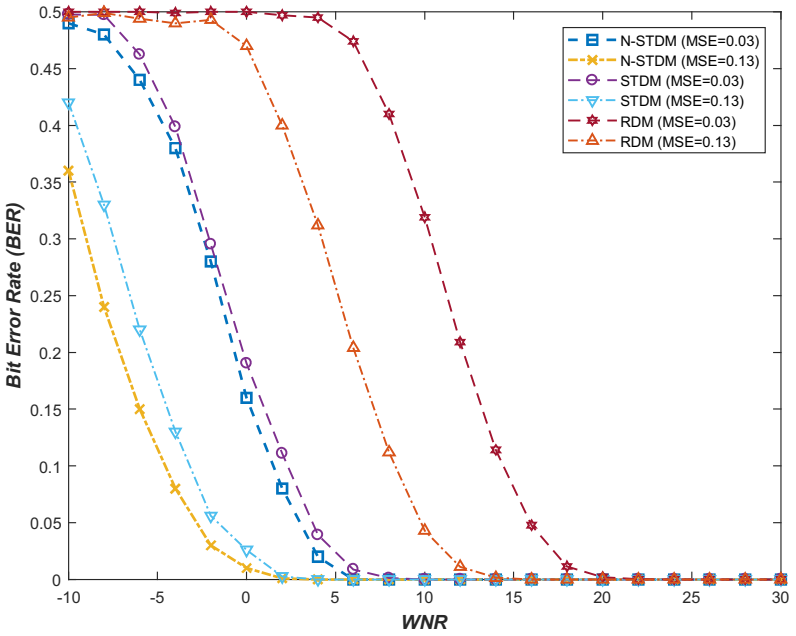


Figure 5.6: Robustness against AWGN for b=96 and L=21

a higher value of Δ. In contrast, our proposed N-STDM watermarking scheme preserves superior robustness against the AWGN attack. The robustness increases while Δ and L increase, with a BER close to 0 when WNR > 0. The BER of STDM and N-STDM watermarking schemes are close to each other, with better performance for N-STDM.
N.B. Based on Watson's model, the family methods could not be applied to PDF documents because they are dependent on the perceptual model of images.

5.6/ Experiments on Real Images

The algorithm is parameterized by two variables, which are the u factor, presented in equation (5.15), and the elements that are considered to compute the norm. The first part of the experiment aimed at finding optimal values for these parameters with respect to the results against the FGA attack. We compared our proposed method against the FGA attack while varying u using two forms. In the first one (Global form), we compute the norm value $\|x\|$ of the whole pixels of the cover image which will be used to embed all the watermark bits. In the second one (Local form), we extract the pixels values from the cover image and arrange them into several vectors. After that, we compute the norm value $\|x\|$ of each vector, in which we will embed the i^{th} bit of the watermark; hence, each bit will have a specific norm value, and each vector will have the same length as the projection vector. In this experiment, grayscale images with size 512×512, such as the images presented in Figure 5.7, have been used as a host signal, the length of the projection vector is set to 64, which allows a 4096-bit message to be embedded into each image, and Δ is adjusted to have watermarked images with same level of fidelity by using a fixed SSIM of 0.982 (PSNR around 45 dB). The robustness of N-STDM method against FGA attack when varying u between 1/5 and 5 using the global form is shown in Figure 5.8. The BER decrease when u increases, with preferable results when u is higher than 1.
Figure 5.9 shows the robustness of N-STDM against FGA attack using the local form. In

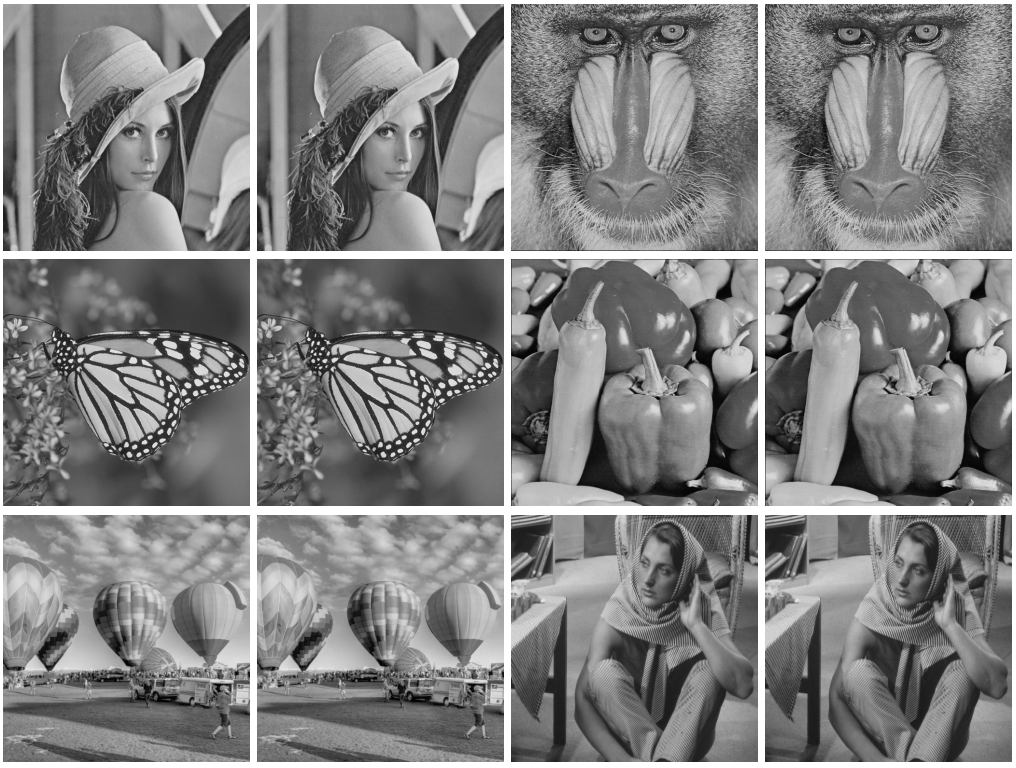


Figure 5.7: The original images (first and third columns) and corresponding watermarked images (second and fourth columns) using the N-STDM method for u=2 with a 4096-bit message embedded and SSIM=0.982

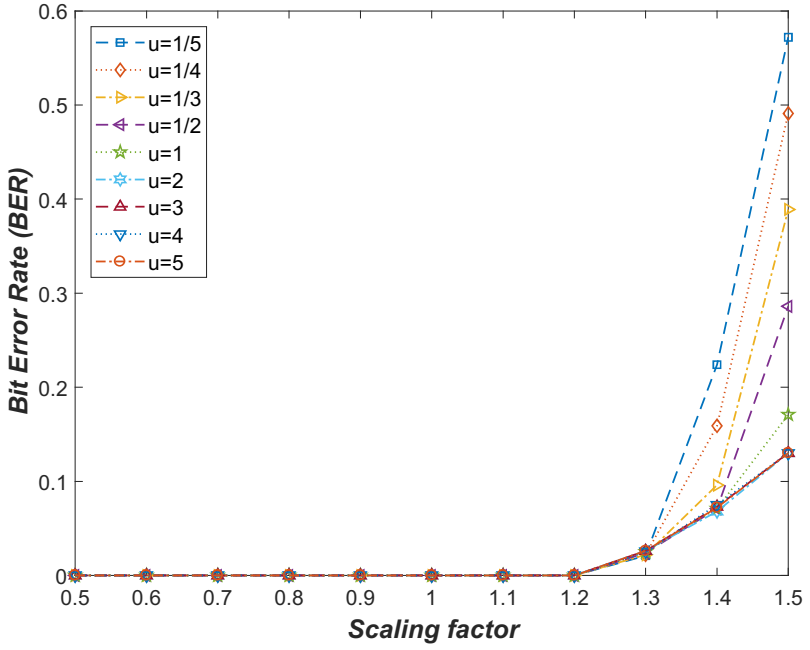


Figure 5.8: Robustness of N-STDM Global form against FGA attack in term of BER while varying u with SSIM=0.982

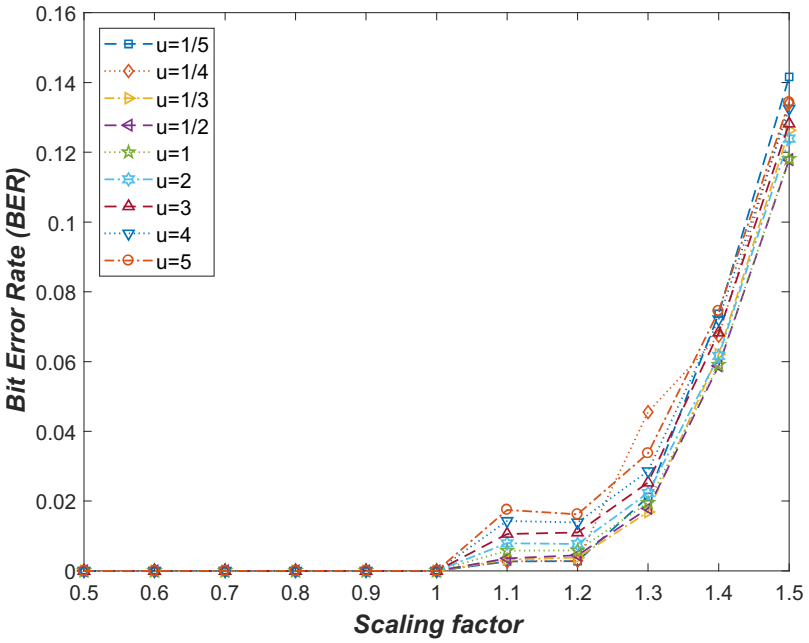


Figure 5.9: Robustness of N-STDM Local form against FGA attack in term of BER while varying u with SSIM=0.982

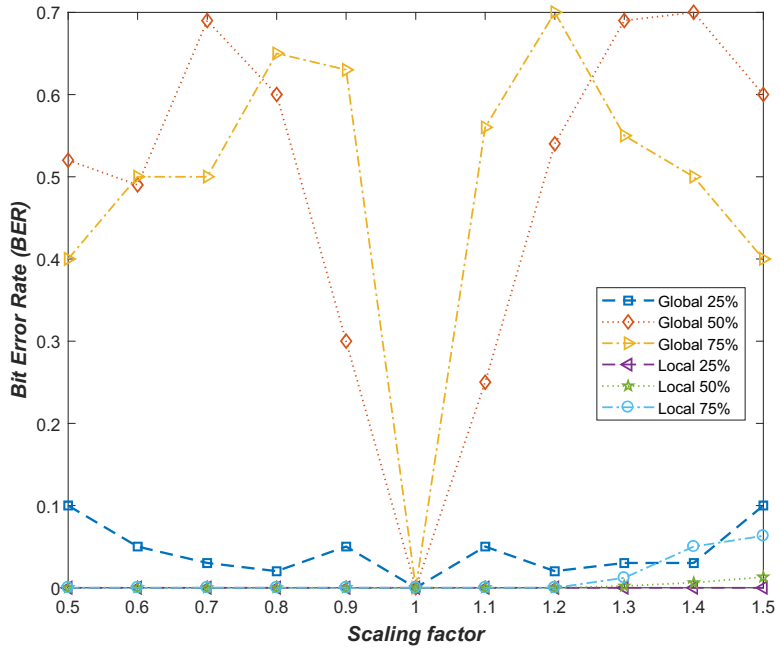


Figure 5.10: Robustness of N-STDM (Local form vs Global form) against FGA attack applied to a given area of the watermarked image (between 25% and 75%) with u=2

this situation, the N-STDM has a good performance regardless of the value of u. The BER increase a little bit when multiplying the pixels of the grayscale images by a gain factor higher than 1.2 due to the clipping error; when the pixels values are beyond 255, it will be clipped to 255. The maximum allowed value is 255. Besides, the robustness of the global form and the local form was tested against the FGA attack, when applied

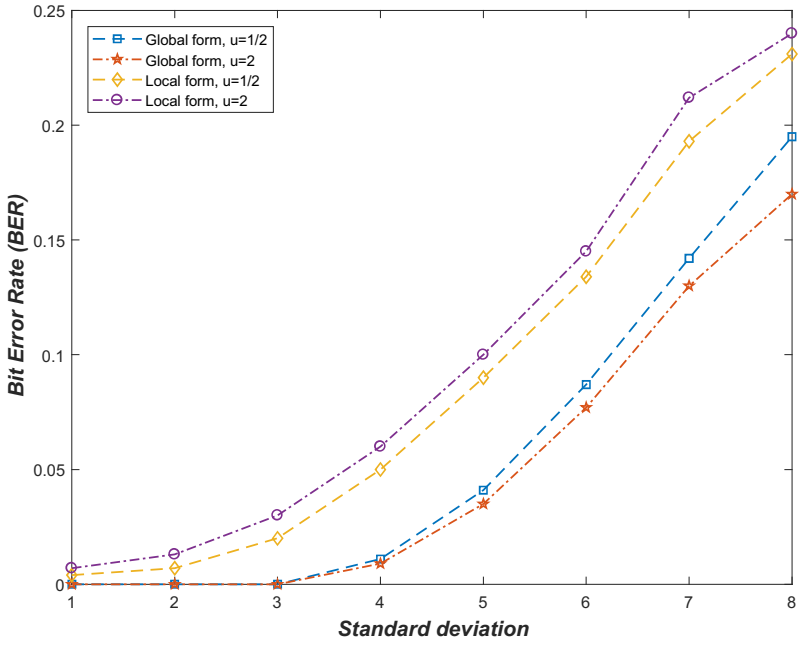
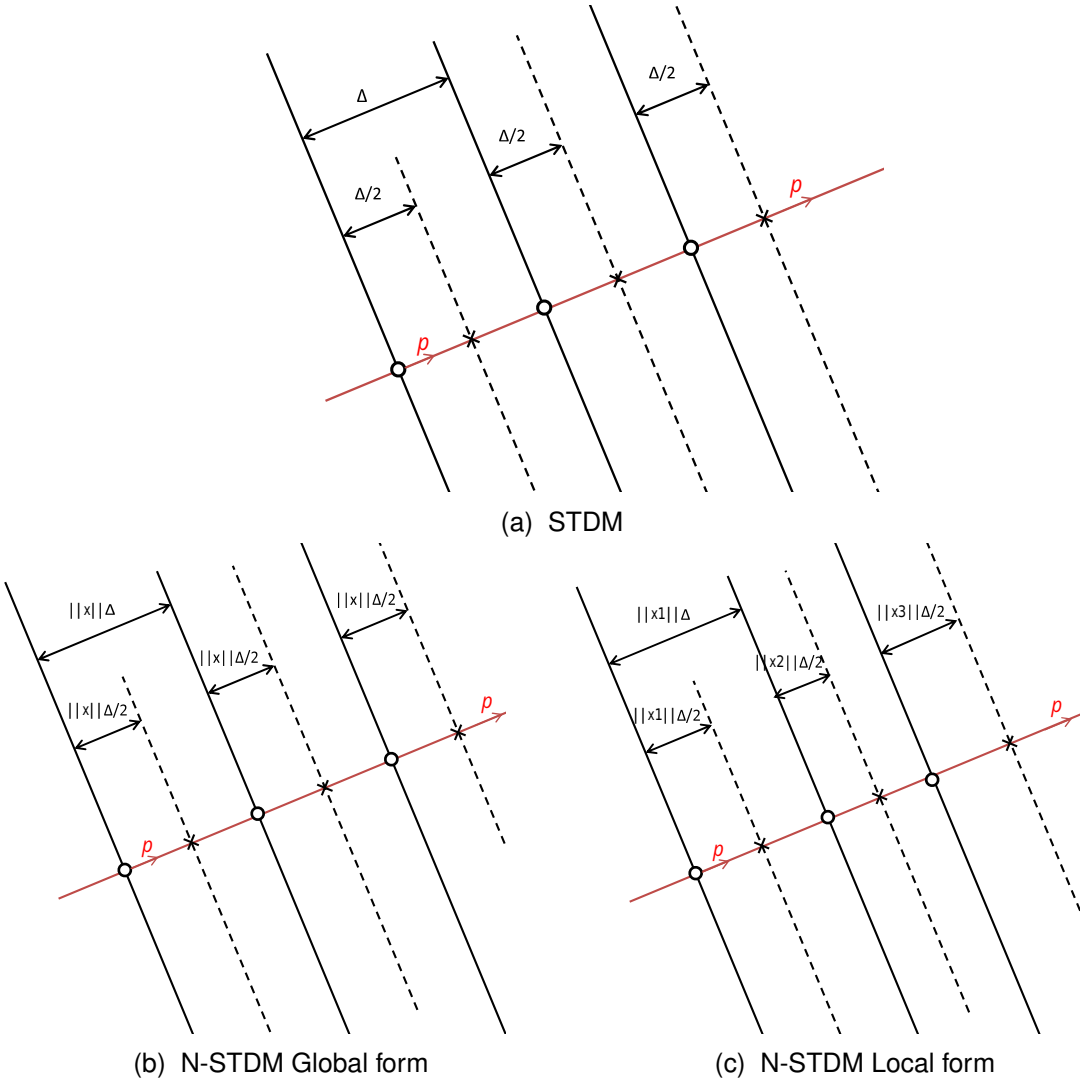


Figure 5.11: Robustness of N-STDM (Local form vs Global form) against AWGN attack in term of BER while varying u with SSIM=0.982

to a given area of the watermarked image (between 25% and 75%). As shown in Figure 5.10, the local form has better robustness comparing to the global form. The global form is affected when the FGA attack is applied to 50% or 75% of the area of the watermarked images. The problem of the global form is solved in the frequency domain when the DCT transform is applied to the cover images. The detailed results are presented in Subsection 5.6.2.

Moreover, we have compared the robustness of global form and local form against the AWGN attack while varying the standard deviation between 1 and 8. As shown in Figure 5.11, the N-STDM watermarking scheme has better robustness using the global form comparing to the local form. Figure 5.12 shows the sets of reconstruction points of the quantizers for embedding each bit in each vector, where the projection of the vector is done before quantization. The signal is quantized to the nearest point on a $\circ$-line to embed a 0-bit and on a $\times$-line to embed a 1-bit. The minimum distance d_{min} between the sets of reconstruction points of different quantizers in the ensemble effectively determines the robustness of the embedding. With STDM [20], as shown in Figure 5.12, $d_{min} = \Delta/2$. In N-STDM global form, the same norm value is used uniformly to embed each bit of the watermark. For that, d_{min}=$||x||\Delta/2$. In the local form, the quantization step size can be seen as a variable step quantizer; each vector has a specific quantization step size. Therefore, d_{min} of the first bit will be different from d_{min} of the second bit, and so on. This variation increases the influence of additive noise attacks on the decoding quantization step size. In the global form, a uniform gain invariant adaptive quantization is obtained at both the embedder and decoder, which improves the robustness against the additive noise attacks. It still better to use the global form since the same norm value $||x||$ is used to embed all the bits of the watermark; accordingly, all the watermarked vectors in the image will have an identical imperceptibility and better robustness.

In conclusion, the parameter value that provides the results with the lowest errors is u=2 using a global form of the norm, which will be applied in the subsequent experiments.



(a) STDM

(b) N-STDM Global form

(c) N-STDM Local form

Figure 5.12: Geometrical representation of STDM, N-STDM Global form, and N-STDM Local form. Points on solid-lines represent embedding for m=1, whereas dashed-lines are for m=0

In the second part of the experiment, we have practically evaluated the correction of the N-STDM watermarking scheme. In other words, we have verified that when a message is embedded into a host signal, prior to applying any attacks, it will be extracted without any error. Practically speaking, 1000 grayscale images with size 512×512 extracted from Boss image database [65] are used as a host signal. The length of the projection vector is set to 64, which allows a 4096-bit message to be embedded into each image. The quantization step size Δ is adjusted in order to have watermarked images with uniform fidelity, a fixed SSIM of 0.982. For all the images, the BER of the extracted watermark are equal to 0, which leads to a conviction that any embedded message could be retrieved without errors prior to applying any attacks. In the third part of the experiment, the visual aspect of the presented approach has been studied. The length of the projection vector is set to 64, which allows a 4096-bit message to be embedded into each image. We used the Structural Similarity Index Measurement (SSIM) to evaluate the quality performance of N-STDM. Figure 5.13 shows the comparison between N-STDM and STDM

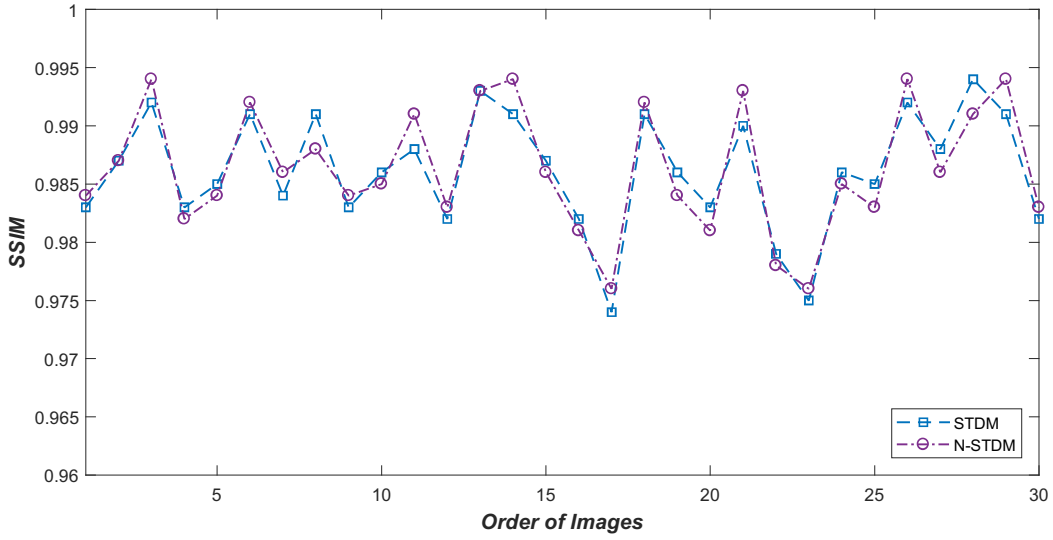


Figure 5.13: SSIM comparison between N-STDM and STDM

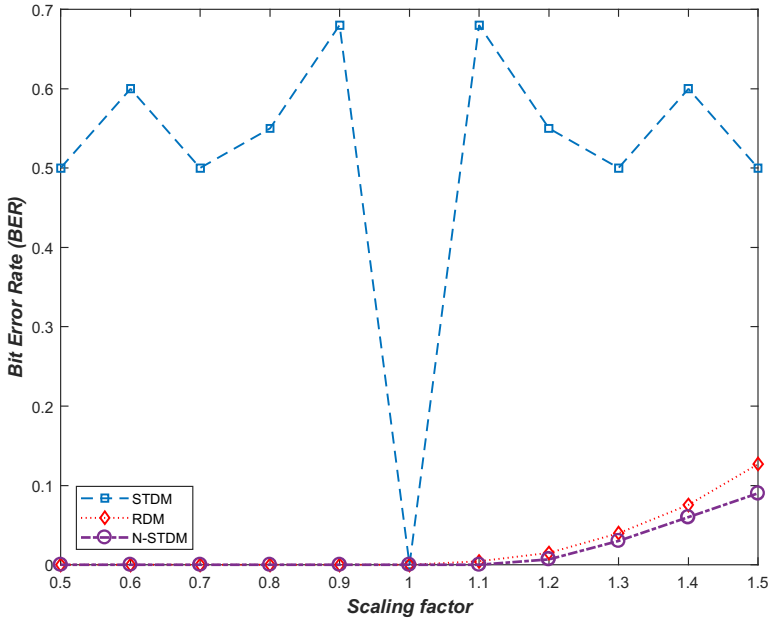


Figure 5.14: Robustness against FGA attack in term of BER with SSIM=0.982

on 30 standard watermarked images in term of SSIM. N-STDM produces watermarked images that yield nearly the same SSIM performance as that of the traditional STDM. Figure 5.7 shows a part of grayscale images. The second and fourth columns display the obtained watermarked images using the N-STDM method with a SSIM=0.982. These watermarked images appear identical to the original ones, to a far extent that they cannot be told differently with the naked eye.

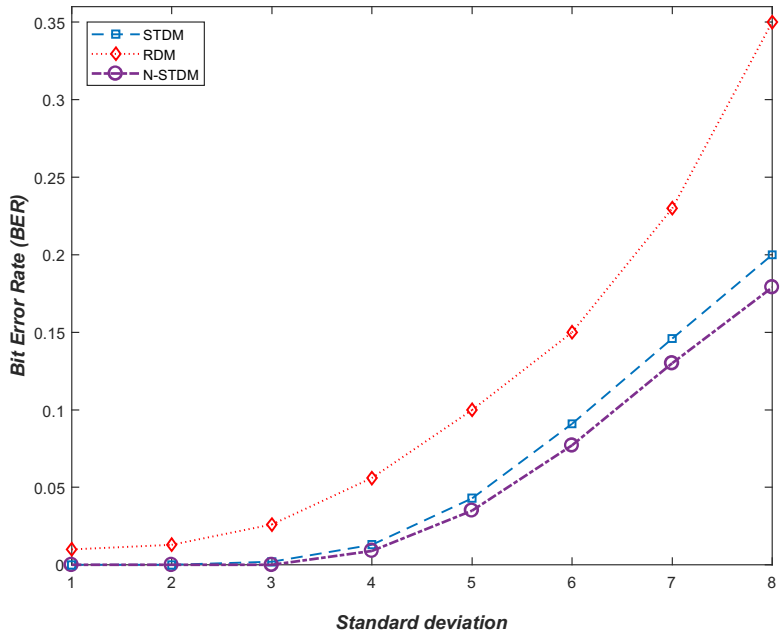


Figure 5.15: Robustness against AWGN attack in term of BER with SSIM=0.982

5.6.1/ Comparison in the Spatial Domain

In this section, we compare the robustness in the spatial domain of our proposed approach with the traditional STDM watermarking scheme and RDM against the FGA attack and AWGN attack. The comparison is conducted using the grayscale images of size 512×512. The length of the projection vector is set to 64 which allows a 4096-bit message to be embedded into each image, and the tested images were watermarked with a uniform fidelity, where SSIM is fixed to 0.982 (PSNR around 45 dB).
As shown in Figure 5.14, N-STDM and RDM have good robustness against the FGA attack, while STDM has low robustness against the FGA attack; even when the values of a gain factor are close to 1, such as 1.1 or 0.9, the BER are excessively high.
As to the RDM, the feature signal for quantization is constructed using the ratio of the previously generated watermarked sample and the current host sample. As a result, it resists the FGA attack but will be affected by the AWGN attack as shown in Figure 5.15. The quantization step size in RDM is a variable step quantizer, whose size is a function of several past watermarked samples, and does not benefit from the randomness property of the spreading vector. Therefore, the attacking noise has more influence on the decoding quantization step size. Concerning the N-STDM, the quantization step size depends on the watermarked samples during the decoding process which will scale linearly with the FGA attack, and based on the global form, a uniform gain invariant adaptive quantization is obtained at both the embedder and decoder. However, since the proposed N-STDM keep on using the spreading vector in the embedding process, it withstands AWGN attack. This is due to the randomness property of the spreading vector, which affects the embedding of the watermark and hence makes it randomly distributed in the host signal samples. Therefore, the proposed N-STDM sustain the robustness against various noise attacks.
Also, the quantization step size Δ in RDM depends on $g(y_{k-L}^{k-1})$, which is not uniform for all the samples. Therefore, as shown in Figure 5.16, each sample will have a specific min-

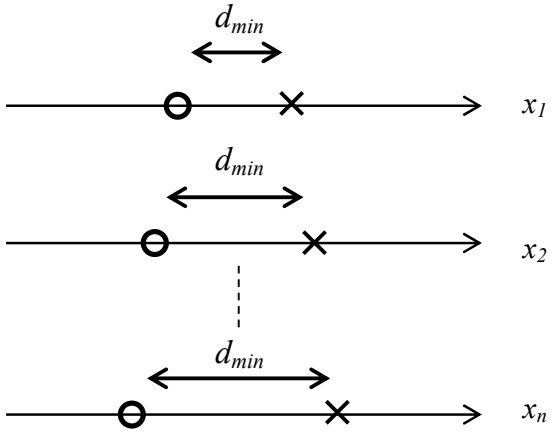
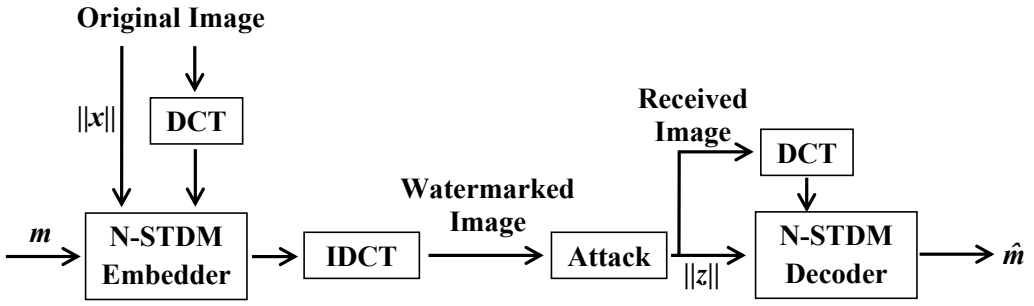


Figure 5.16: Geometrical representation of RDM in term of d_{min}. The signal is quantized to the nearest point on a $\circ$-line to embed a 0-*bit* and on a $\times$-line to embed a 1-*bit*

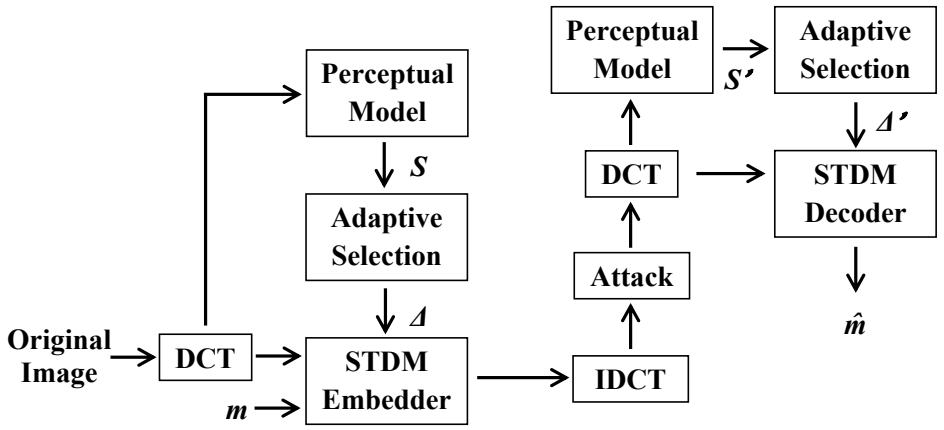
imum distance d_{min}. By this way, each watermarked sample will have a specific level of robustness. Comparing to N-STDM global form, as shown in Figure 5.12, d_{min} is uniform for all the sets of reconstruction points of different quantizers. Therefore, the trade-off between robustness and transparency is achieved. The watermark bit is spread into a group of samples instead of one sample, which also increase the level of robustness. Each time the watermarking space increases, the probability of accurate decoding is higher. This is mainly because of the spreading property of the watermark using the spreading vector. Our approach achieves significant robustness against the FGA attack with an improvement of 98% in terms of BER compared to traditional STDM and 24% compared to RDM. As for the AWGN attack, an improvement of 21% is shown compared to STDM and 50% compared to RDM.

5.6.2/ Comparison in the Frequency Domain

To test the performance of N-STDM in the frequency domain, we have implemented the DCT transform on the grayscale images of size 512×512 as shown in the block diagram in Figure 5.17a. First of all, we have computed the norm value $||x||$ of the original image to be used during the embedding process. After that, we have divided the image into 8×8 blocks of pixels, upon which the DCT transform was later performed to get the DCT coefficients. A part of these coefficients was used as a host vector of length L, in which we have later embedded the i^{th} bit of the watermark message m. Then, we have performed the inverse DCT transform at each block to get the watermarked image.
To assure a fair comparison, we compared the proposed N-STDM watermarking scheme with I-ASTDM [46], STDM-MW-SS [48] and STDM-SP-wm [67]; family methods based on the perceptual model. The block diagram of those family methods is shown in Figure 5.17b, where the quantization step size Δ is modified based on the slacks vectors S which are computed using Watson's model. The FGA attack, AWGN attack, JPEG compression, and a variety of common signal processing attacks are used to verify the performance of our proposed scheme. The 2nd-21st DCT coefficients have been used for all the algorithms. These coefficients have been selected based on the zig-zag-scanned order of each 8×8 block, in which we embed 1 bit of the watermark. The embedding rate



(a) Block diagram of the N-STDM method



(b) Block diagram of the family methods based on Watson's model

Figure 5.17: Block diagrams of the N-STDM method (a) and the family methods based on Watson's model (b)

is 1/64, *i.e.* one bit in each 8×8 block, which allows the embedding of a 4096-bit message into each image. The tested images were watermarked using a uniform fidelity, with a fixed SSIM of 0.982 (PSNR around 45 dB) for the first part of comparisons, and a fixed SSIM of 0.953 (PSNR around 40 dB) for the second part of comparisons by regulating the quantization step size and the factors that adjust the watermarking strength. All of the experimental results are obtained by averaging over 100 runs. In the first part of the experiments, the global form of the N-STDM watermarking scheme was examined facing the FGA attack, when applied to a given area of the watermarked image (between 25% and 100%). As shown in Figure 5.18, the robustness is highly improved comparing to the global form in the spatial domain presented in Figure 5.10, especially when the FGA attack is applied to 50% or 75% of the area of the watermarked image.
Moreover, the proposed N-STDM is compared with the family methods based on Watson's model against the FGA attack. As expected, the results of the posted comparison have met the suggestions of the proposed method. According to Figures 5.19 and 5.20, the proposed N-STDM watermarking scheme has a slightly better level of robustness than the family methods, varying from 5% comparing to STDM-MW-SS to 22% comparing to STDM-SP-wm for SSIM = 0.982, while proved to have better to more superior robustness against the AWGN attack and JPEG compression.

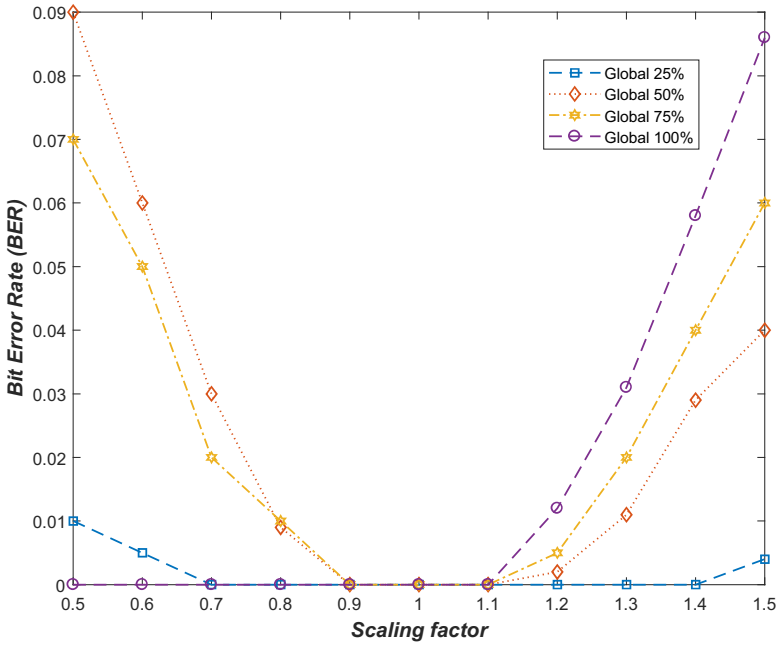


Figure 5.18: Robustness of N-STDM against FGA attack applied to a given area of the watermarked image (between 25% and 100%) with SSIM=0.982

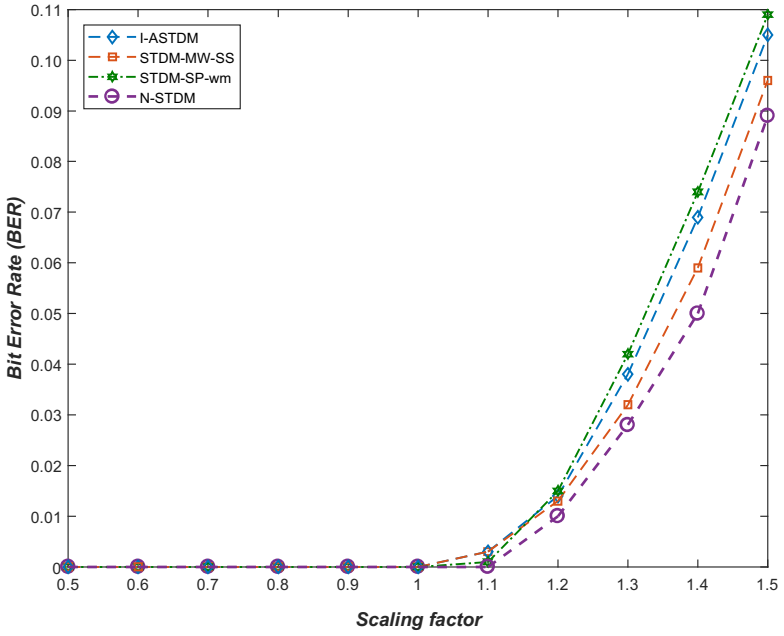


Figure 5.19: Robustness against FGA attack in term of BER with SSIM=0.982

The robustness against AWGN attack has been tested in terms of BER while varying the standard deviation between 1 and 8. As shown in Figures 5.21 and 5.22, the proposed N-STDM watermarking scheme achieves better performance than the STDM-SP-wm (improvement of 14%) and notably superior performance to the I-ASTDM (improvement of 38%) and STDM-MW-SS (improvement of 56%).
Figures 5.23 and 5.24 illustrate the robustness to JPEG compression in term of BER while varying the JPEG quality between 10 and 100. The JPEG quality denotes the compress-

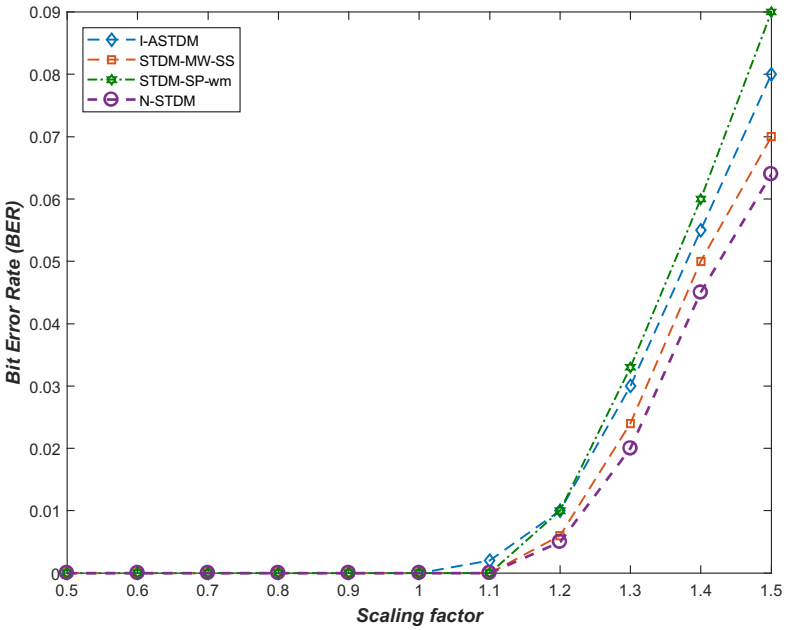


Figure 5.20: Robustness against FGA attack in term of BER with SSIM=0.953

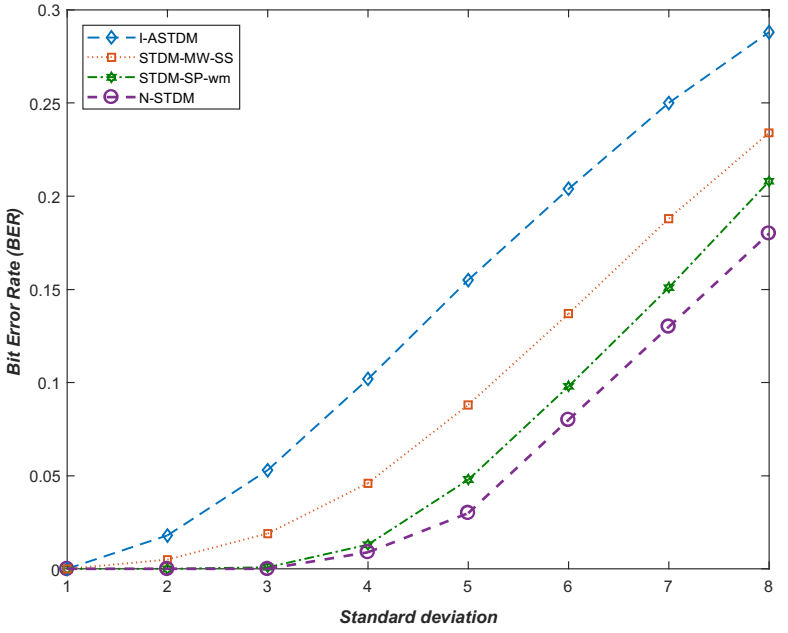


Figure 5.21: Robustness against AWGN attack in term of BER with SSIM=0.982

ibility of the JPEG compressor; lower numbers mean lower quality. The N-STDM has better performance than STDM-SP-wm (improvement of 13%) with more notable variations comparing to STDM-MW-SS (improvement of 18%) and I-ASTDM (improvement of 27%) against the JPEG compression.

To further evaluate the performance of N-STDM watermarking scheme, we have compared the robustness against noise addition attacks, image filtering attacks, and geometric attacks. All the watermarked images were exposed to many different attacks such as Salt&Pepper with noise densities $\in \{0.005, 0.01\}$, Gaussian filtering, Median filtering,

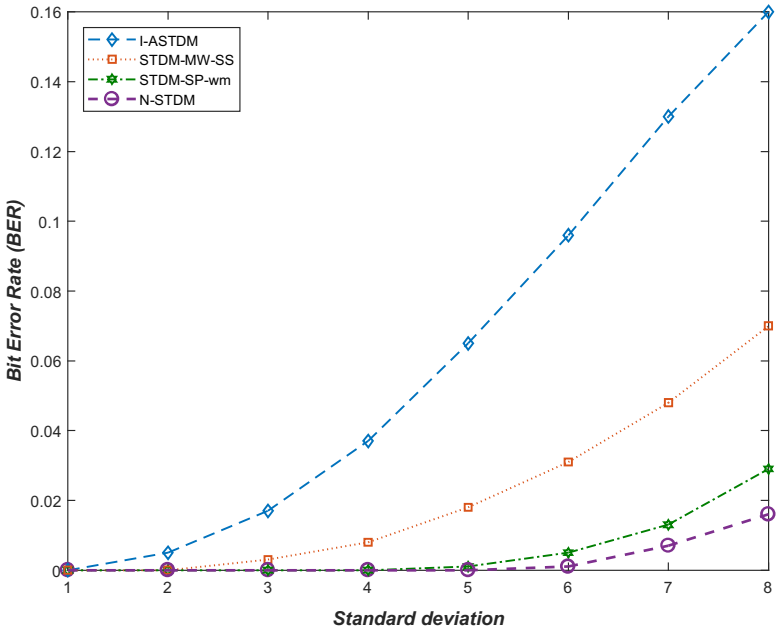


Figure 5.22: Robustness against AWGN attack in term of BER with SSIM=0.953

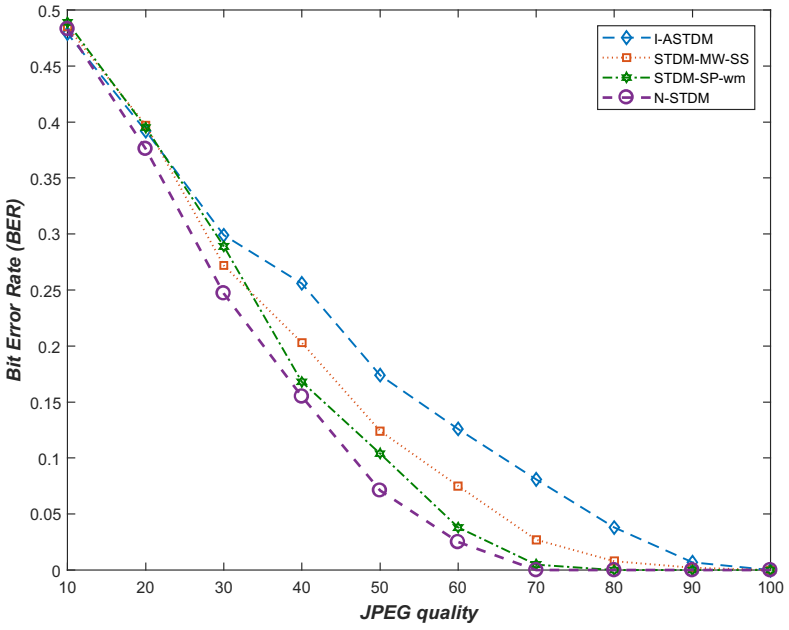


Figure 5.23: Robustness against JPEG compression in term of BER with SSIM=0.982

Wiener filtering, Average filtering, Cropping, Rotation, and Resizing. The detailed experimental results are shown in Table 5.1 with SSIM=0.982 and Table 5.2 with SSIM=0.953. The experiments have verified that our proposed scheme is not only robust against the FGA attack but also robust to common signal processing attacks. N-STDM has achieved good robustness against Salt&Pepper attack and image filtering attacks comparing to the family methods based on Watson's model. As for the geometric attacks, N-STDM achieved good robustness against the cropping attack when we cropped 10% of the image, but the BER slightly increased when we increased the cropping dimension of the

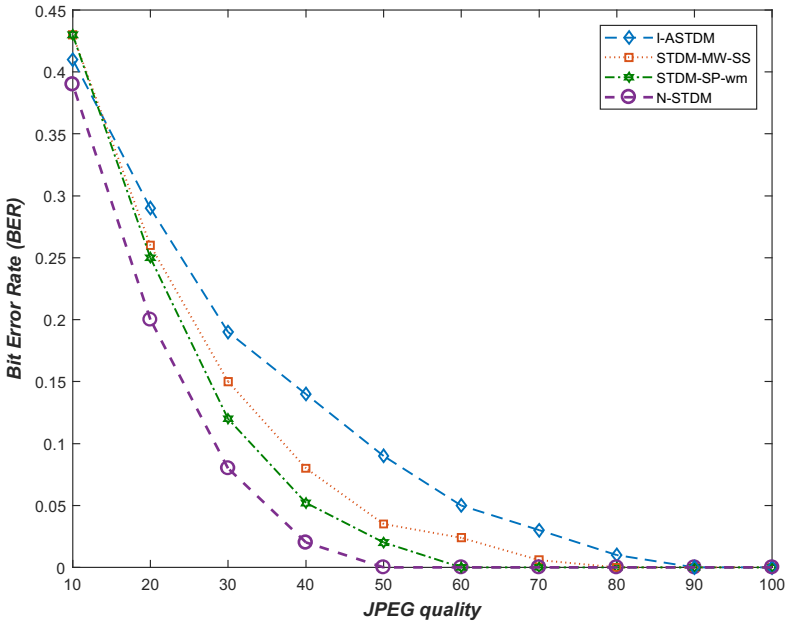


Figure 5.24: Robustness against JPEG compression in term of BER with SSIM=0.953

Table 5.1: BER comparison under noise addition, image filtering, and geometric attacks with SSIM=0.982

Attacks	I-ASTDM	STDM-MW-SS	STDM-SP-wm	N-STDM
Salt&Pepper (d=0.005)	0.12	0.11	0.12	0.09
Salt&Pepper (d=0.01)	0.21	0.19	0.22	0.17
Gaussian filtering (3×3)	0.016	0.024	0.035	0.014
Median filtering (3×3)	0.25	0.21	0.22	0.16
Wiener filtering (3×3)	0.26	0.2	0.22	0.16
Average filtering (3×3)	0.35	0.31	0.29	0.24
Crop (10%)	0.0064	0.0063	0.0066	0.0061
Crop (20%)	0.023	0.022	0.024	0.08
Rotation (1°)	0.48	0.46	0.47	0.46
Re-Rotation (-1°)	0.18	0.09	0.1	0.07
Resizing (1024×1024)	0.025	0.003	0.008	0
Resizing (256×256)	0.31	0.28	0.24	0.21

Table 5.2: BER comparison under noise addition, image filtering, and geometric attacks with SSIM=0.953

Attacks	I-ASTDM	STDM-MW-SS	STDM-SP-wm	N-STDM
Salt&Pepper (d=0.005)	0.09	0.07	0.08	0.06
Salt&Pepper (d=0.01)	0.16	0.14	0.14	0.1
Gaussian filtering (3×3)	0.004	0.002	0.01	0
Median filtering (3×3)	0.17	0.14	0.15	0.11
Wiener filtering (3×3)	0.16	0.13	0.12	0.07
Average filtering (3×3)	0.26	0.22	0.21	0.16
Crop (10%)	0.0053	0.0051	0.0052	0.0041
Crop (20%)	0.019	0.017	0.018	0.06
Rotation 1°	0.47	0.48	0.46	0.45
Re-Rotation 1°	0.05	0.05	0.06	0.04
Resizing (1024×1024)	0.001	0.001	0.001	0
Resizing (256×256)	0.23	0.21	0.19	0.12

image to 20%. Concerning the rotation attack, all the compared methods were vulnerable to the rotation of 1° of the image, though the BER highly decrease if we re-rotate the image by -1°. As for the resizing attack, N-STDM achieved better robustness comparing to the family methods, noting that the robustness improved as well while adjusting the watermarking strength based on Δ.

5.7/ FINDINGS AND DISCUSSION

We have studied the performance of the proposed N-STDM watermarking scheme in the spatial domain and the frequency domain using the grayscale images and PDF documents. As for the experiments on real images in the spatial domain, the proposed N-STDM has achieved high robustness against FGA and AWGN attacks comparing to STDM and RDM. STDM has good robustness against AWGN attack, but it is highly affected by the FGA attack. RDM achieves good robustness against FGA attack but has a weak performance against the AWGN attack.
With respect to the experiments on real images in the frequency domain, the family methods (I-ASTDM, STDM-MW-SS, and STDM-SP-wm) have good robustness against the FGA attack, and an accepted robustness against AWGN attack and JPEG compression. The proposed watermarking scheme, N-STDM, on the other hand, achieves a superior performance comparing to the family methods varying from little different against FGA

Table 5.3: Summary of the findings

Watermarking scheme	**Findings**
STDM	*Strong against the AWGN attack *Week against the FGA attack
RDM	*Strong against the FGA attack *Week against the AWGN attack
Family methods based on Watson's model	*Robust against FGA, AWGN, JPEG compression, Image filtering, and Cropping attacks *Weak against Rotation attack *Dedicated for images *Could only be applied in the frequency domain
N-STDM	*More flexible *Could be applied in the spatial domain and frequency domain *Any element can be used as support to embed the watermark (such as images and PDF documents) *High level of robustness against FGA, AWGN, JPEG compression, and Image filtering attacks *Robust against Copping, and Resizing attacks *Weak against Rotation attack

attack to a greater one against AWGN, JPEG compression, and image filtering attacks. As for the geometric attacks, N-STDM has achieved good robustness against the cropping attack, but the BER slightly increased while increasing the cropping dimension of the image. N-STDM and the family methods are vulnerable to the rotation attack, though the BER highly decrease if we re-rotate the image. Concerning the resizing attack, N-STDM has achieved better robustness comparing to the family methods, noting that the robustness improved as well while adjusting the watermarking strength based on Δ.
Moreover, we proved that our approach could also be used as a blind watermarking scheme for PDF documents under a sufficient transparency-robustness tradeoff. We exploited the x-coordinates values of characters as real cover elements to embed the watermark. The comparison between N-STDM and the traditional STDM and RDM against FGA and AWGN attacks shows that the proposed N-STDM achieves better performance than the mentioned watermarking schemes. A summary of the findings is presemted in Table 5.3.

5.8/ CONCLUSION

In this chapter, we have presented an improved version of STDM watermarking scheme, which is essentially invariant to FGA attack. In this method, called N-STDM, we scaled the quantization step size in the same way the watermarked signal is scaled as an invariant adaptive size based on the global form. Experiments on real images in the spatial domain and frequency domain have verified that our method is not only robust to the FGA attack but also robust to common signals processing attacks such as AWGN attack, JPEG compression, and Image filtering attacks comparing to STDM, RDM, and the family methods based on Watson's model. As for the geometric attacks, N-STDM has achieved good robustness against the resizing and cropping attacks, but the BER slightly increased while increasing the cropping dimension of the image. N-STDM and the family methods are vulnerable to the rotation attack, though the BER highly decrease if we re-rotate the image. Moreover, we have verified that our approach could also be used as a blind watermarking scheme for PDF documents with a perceptual advantage and better robustness over STDM and RDM.

6

USING DEEP LEARNING FOR IMAGE WATERMARKING ATTACK

6.1/ INTRODUCTION

In the last chapter, we have presented an improved version of the STDM watermarking scheme that resists the FGA attack and achieves good robustness against a variety of filtering and geometric attacks.
Protection of digital contents was and still one of the most important topics in scientific research. With the progression of internet technologies, unauthorized users illegaly duplicate, authenticate, and distribute digital contents. Therefore, various watermarking methods have been studied for a wide range of applications, such as broadcast monitoring, copyright protection, content authentication, and copy control [47]. The embedded watermark could be a single bit or multi-bit generated from a pseudo-random sequence, obtained from a pseudo-random number generator. Also, this watermark could be a binary image or a gray-scale image.
The watermarking algorithms are characterized by several properties such as payload, robustness, and fidelity. These properties are contradictory, and a tradeoff should be made based on the targeted application. Therefore, the transform domains such as SVD, DCT, and DWT are usually used. The DCT domain is more robust than the spatial domain. Especially against simple image processing operations like brightness, blurring, and low pass filtering [39]. Also, the DWT is a very attractive transform which makes the watermarked images much more robust. The DWT domain composes the image in different levels of resolution and processed from high resolution to low resolution [17]. Hiding the watermark with more energy in an image will enhance the level of robustness. Furthermore, the Singular Value Decomposition (SVD) has been widely used for digital image watermarking. The SVD preserves the visual perception of the cover image and good robustness against most types of attacks [49, 81].
In the latest years, Artificial Intelligence (AI) and Deep Learning fields have exploded as computers and servers get closer to delivering human-level capabilities. Nowadays, companies around the world are looking to use their big data sets as a training ground to develop programs that can interact with the world in more natural ways, and extract from it useful information that has never been done before. Deep Learning and Neural Networks currently provide a perfect solution to many problems in speech recognition, image recognition, and natural language processing [73, 88, 80, 94]. Moreover, the Convolutional Neural Network (CNN), which is a type of artificial neural network, has been widely used for image processing, segmentation, classification, and other auto-correlated data

[72, 91, 90, 85]. Image denoising and super-resolution are topics of great interest in image processing that can lead to an improvement in image quality. Lately, the denoising accuracy is performed by deep neural networks by creating a mapping between the clean and noisy images [101, 97, 105, 98]. The successful results of CNN for image denoising are assigned to its large modeling capacity and enormous advances in network training and design. CNN with deep architecture effectively increases the flexibility for exploiting image characteristics and improving the training process and denoising performance. The advances are achieved with the learning methods for training CNN, including batch normalization and Rectifier Linear Unit (ReLU). This type of denoising could be harmful to watermarked images since the embedded watermark is like a sequence of noise embedded in the images.
Image filtering and denoising are a dangerous type of attacks in digital watermarking since it recovers the original value for each pixel of the image. In this paper, we studied the effect of a Fully Convolutional Neural Network (FCNN) against watermarked images. We evaluated such type of denoising against SS and STDM watermarking schemes, and whether it could be used as a new type of attack in digital watermarking.
This chapter corresponds to an article submitted to an international journal (Signal Processing: Image Communication). The remainder of this chapter is organized as follows. Related work of watermarking attacks is presented in Section 6.2. Section 6.3 briefly presents the architecture of the Convolutional Neural Network. The Fully Convolutional Neural Network is presented in Section 6.4. The evaluation of the proposed attack is presented in Section 6.5. Finally, in Section 6.6 we give our conclusion.

6.2/ Related Works

The watermarked images could be affected by different types of attacks, such as additive noise, lossy compression, geometric distortions, and image filtering attacks [64, 82]. The most common types of noise attack are salt&pepper noise and additive Gaussian noise attacks. Salt&pepper noise alters the pixel value to 0 or 255 (black and white) for an 8-bit gray-scale image and the additive Gaussian noise reduces the visual quality of the image.
Median, Wiener, Average, and Gaussian filters are part of image filtering attacks that could destroy the watermark embedded in the watermarked images. The median filter is a non-linear digital filtering technique, used to remove noise from an image. The median filter is a non-linear digital filtering technique, which preserves the edges in the image while removing noise. Wiener filter is usually used for removal of blur in images. The average filter reduces the amount of intensity variation between pixels; each pixel value is replaced with the mean value of its neighbors, including itself. The Gaussian filter usually used to blur the image and to reduce contrast and noise.
Geometric attacks are geometric distortions to an image which include operations such as scaling, rotation, cropping, and translation [37]. They are classified basically into local and global geometric attacks. Local geometric attacks affect portions of an image using such as the cropping attack, and the global geometric attacks affect all the pixels of an image using such as the rotation and the scaling attacks. Below are basic transformations of geometric attacks.
Cropping attack refers to clipping or cutting part of the watermarked image. Rotation attack rotates the watermarked image by different angles by changing the coordinate axes. Scaling attack is the process of resizing the watermarked image. Translation attack repo-

sitions an image by shifting a coordinate location to other coordinate location along a straight line. Flipping or mirroring attack flips an image vertically or horizontally.
Several methods were proposed to improve the robustness against geometric attacks. Enping Li *et al*. [43] presented a blind image watermarking scheme using a wavelet tree quantization to enhance the robustness against geometric attacks such as rotation, scaling, and cropping. Liu *et al*. [50] introduced a robust multi-scale full-band image watermarking based on the Singular Value Decomposition (SVD) and the Distributed Discrete Wavelet Transform (DDWT). This method has good robustness against cropping and rotation attacks. Li [62] proposed a robust image watermarking scheme based on a computer-generated hologram against geometric attacks, including translation, rotation, cropping, flipping, and scaling attacks. He *et al*. [84] proposed an image watermarking algorithm based on histogram modification resistant to geometrical attacks, including rotation, cropping, scaling, and translation attacks. Fazli and Moeini [87] presented a robust image watermarking method based on DCT, SVD, and DWT for the correction of geometric attacks. This method enhances the robustness against cropping, translation, and rotation attacks.
JPEG compression and Fixed Gain Attack (FGA) are also a type of attacks that could destroy the embedded watermark in the watermarked image. Li *et al*. [44] improved the STDM watermarking scheme using a perceptual model to enhance the robustness against JPEG compression. Lin *et al*. [63] improved an image watermarking technique against JPEG compression. The watermark is embedded in the low-frequency coefficients after applying the DCT frequency transform on the original image. Li and Cox [48] improved the robustness of STDM against amplitude scaling and JPEG compression using a perceptual model based on Watson's model. We have also proposed a blind image watermarking using Normalized STDM robust against FGA attack, AWGN attack, and JPEG compression [102].
The robustness is one of the main properties in digital watermarking. Each watermarking scheme survives a specific type of attacks based on the target application. It is the first time that we propose Deep Learning to attack watermarked images, and it could be a harmful type of attack for watermarked images.

6.3/ CONVOLUTIONAL NEURAL NETWORK

Deep Learning (DL) is a subset of machine learning, inspired by the function and the structure of the brain [86]. DL architectures have been applied in many fields, such as computer vision, image analysis, audio recognition, and image classification [79]. Varieties of DL architectures such as Convolutional Neural Network (CNN) have been studied and used by researchers for a special use case data [72, 91, 90, 85]. CNN model is constructed with input layers, output layers, and hidden layers in between. The major components, as shown in Figure 6.1 are the convolution layers, pooling or subsampling layers, activation functions, and fully connected layers.
CNN takes an input image, which passes through multiple convolutions and subsampling layers. Each convolution layer includes a series of filters that are presented as matrix numbers. A convolution product will be applied between these filters and previous image matrix to extract important features known as output channels maps. After that, the pooling layers will reduce the dimension of the input map and retain the important information. Max pooling technique is one of the techniques of subsampling, which returns the maximum value from a block. Besides, activation functions such as RELU (Rectified Linear

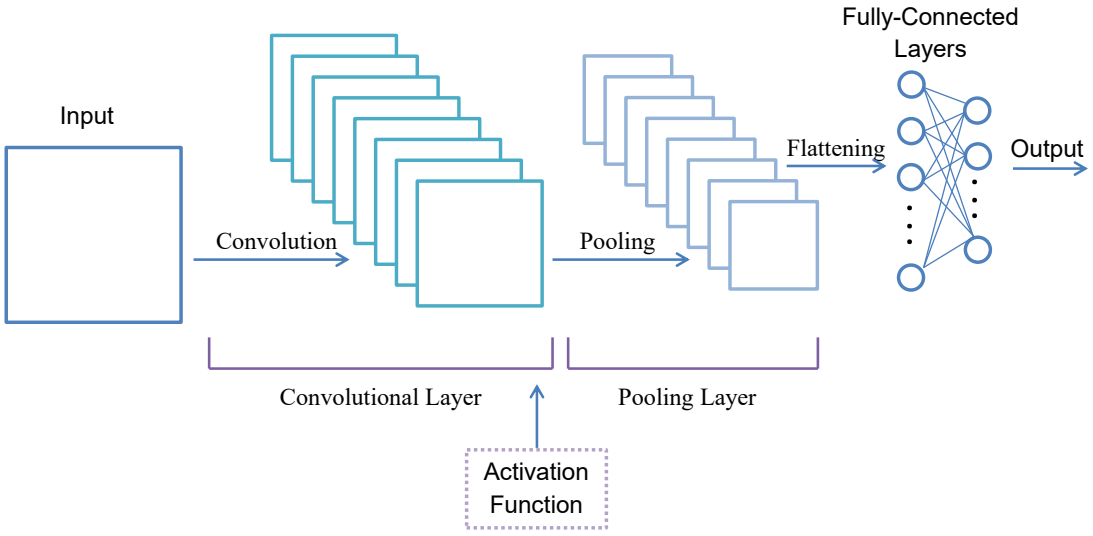


Figure 6.1: Convolutional Neural Network architecture

Unit) are usually employed to introduce the non-linearity in the network. The RELU function round the negative values to zero. Noting that, other non-linear functions could be used, such as sigmoid and hyperbolic tangent denoted as Tanh. Also, the Batch normalization could be used when training the network to reduce the overfitting, and decrease the learning time.
CNN models are dominant in many computer vision tasks and have accomplished startling achievements across a variety of domains, such as face recognition, image classification, self-driving cars, and many more.

6.4/ Fully Convolutional Neural Network based Denoising

Nowadays, several types of noise affect the visual quality of digital images. A deep network for image denoising can deal with different kind of noises such as speckle noise, and Additive White Gaussian Noise (AWGN) [101]. In this network, the considered generator is an encoder-decoder, which has skip connections between the mirrored layers of the encoder and decoder as shown in Figure 6.2. In this figure, each white box denotes a set of feature maps arisen from an encoding layer, and each blue box expresses a set of feature maps arisen from a decoding layer.
Notable features are extracted using the encoder, to remove the noise and preserve the detailed structure of the image simultaneously. Besides, the decoder recovers successive image details and provides a clean version of the noisy image. The skip connections transport directly the information from the encoder layers to its corresponding decoder layers, to share the low-level information between the noisy image and the clean image. Different levels of details are recollected using the skip connections to be used during the reconstruction of the output clean image.
The first layer of the encoder is a Convolution-ReLU layer, and the other layers

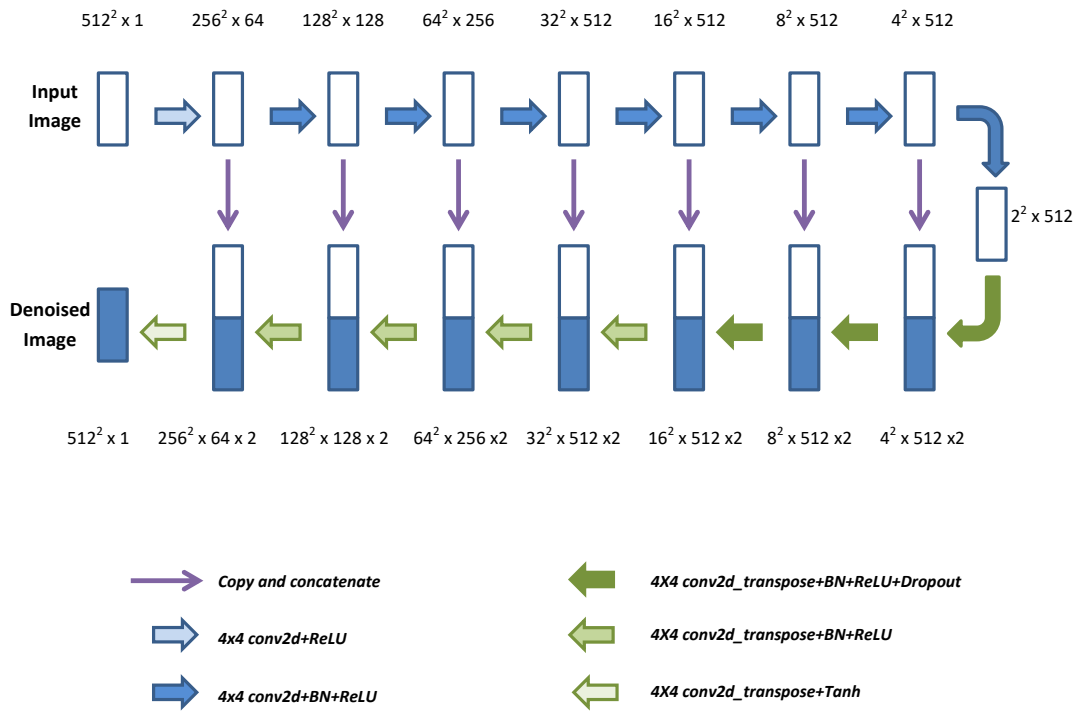


Figure 6.2: Schematic diagram of the Fully Convolutional Neural Network based Denoising [101]

are Convolution-BatchNorm-ReLU layers. The decoder is constituted of transpose Convolution-BatchNorm-ReLU layers with a dropout rate of 50%, and of transpose Convolution-BatchNorm-ReLU layers without dropout. In the end, the denoised image is obtained by a transpose Convolution-Tanh layer.
The loss function has a major impact during the training process. The choice of the loss function is usually based on the $L1$ norm or the $L2$ norm, which is the popular option. However, in the case of image denoising, the $L2$ norm does not improve the visual quality of the image. Zhao *et al.* [99] compared several losses, such as the Structural Similarity (SSIM) index for image quality. The metric of a loss function should also reflect the visual quality. Therefore, a combination of loss functions will fulfill the desired objective. In this network, the combination is made between the $L1$ norm and the Structural Similarity (SSIM) index, denoted by $\zeta^{L1+SSIM}$, which are defined as:

$$\zeta^{L1}(x,y) = \frac{1}{N}\sum_{p\in P}|x(p) - y(p)|, \tag{6.1}$$

$$\zeta^{SSIM}(x,y) = 1 - \frac{1}{N}\sum_{p\in P}\frac{2\mu_x\mu_y + c_1}{\mu_x^2 + \mu_y^2 + c_1}\cdot\frac{2\sigma_{xy} + c_2}{\sigma_x^2 + \sigma_y^2 + c_2}, \tag{6.2}$$

where N is the number of pixels p in the patch P. x denotes the noisy image, and y denotes the clean image. σ and μ represent the standard deviation and the means that depend on a pixel p, which are computed using a Gaussian filter with standard deviation

σ_G. c_1 and c_2 are two constants $\ll 1$. σ_{xy} can be estimated as [34]:

$$\sigma_{xy} = \frac{1}{N-1}\sum_{i=1}^{N}(x_i - \mu_x)(y_i - \mu_y). \tag{6.3}$$

This fully convolutional network was introduced for image denoising. It can remove AWGN and multiplicative speckle noise [101], and it could be a harmful type of attack for watermarked images.

6.5/ EVALUATION

STDM and SS watermarking schemes are examined against the Fully Convolutional Neural Network, which can be considered as denoising attack (FCNNDA). A subset of 10000 gray-scale images of 512×512 pixels was used as a data set provided by the BOSS database [66].
For this type of attack, 2000 images are used to train the network during 50 epochs, and 500 remaining images are used during the test. The watermarked images were also tested against different attacks such as salt&pepper, Median filtering, Gaussian filtering, Average filtering, and Wiener filtering. This comparison will show the quality and robustness variation against each type of attack. Structural Similarity Index Measure (SSIM) is used to compare the quality of the watermarked images after applying the attacks. Bit Error Rate (BER) is used to compare the level of robustness when a watermark in form of bits is embedded in the original image. Normalized Correlation (NC) is used to compare the level of robustness for a binary watermark. Normalized Cross-Correlation (NCC) is used to measure the level of robustness of gray-scale watermark.

$$NC(w,\hat{w}) = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}[w(i,j)][\hat{w}(i,j)]}{\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}[w(i,j)]^2}\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}[\hat{w}(i,j)]^2}}, \tag{6.4}$$

$$NCC(w,\hat{w}) = \frac{\sum_{i=1}^{M}\sum_{j=1}^{N}[w(i,j)-\mu_w][\hat{w}(i,j)-\mu_{\hat{w}}]}{\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}[w(i,j)-\mu_w]^2}\sqrt{\sum_{i=1}^{M}\sum_{j=1}^{N}[\hat{w}(i,j)-\mu_{\hat{w}}]^2}}, \tag{6.5}$$

where M and N indicate the number of pixels of the watermark, w and $\hat{w}$ represent the original watermark and the extracted one. μ_w and $\mu_{\bar{w}}$ denote the mean of the original and the extracted watermarks respectively.
Also, the Standard Deviation (SD) is used to show the variation from the mean value.

$$SD = \sqrt{\frac{\sum_{i=1}^{n}|s_i - \bar{s}|^2}{n-1}}, \tag{6.6}$$

where n indicates the number of samples and $\bar{s}$ represents the sample mean.

6.5.1/ STDM AND DCT BASED WATERMARKING

The watermarks are embedded in the original images based on STDM and DCT transform. Each image is divided into 16×16 blocks of pixels, and the DCT transform is applied on each block to get the DCT coefficients. The zigzag scanned order is used to select the middle-frequency components, through which the watermark bit is embedded using the STDM watermarking scheme. The embedding rate is 1/256, which allowed the embedding of 1024 bit into each image.
STDM is evaluated against FCNNDA with four different scenarios:

- Scenario 1: Embedding binary image watermark in the original image.
- Scenario 2: Embedding gray-scale watermark in the original image. In this part, each gray-scale value is transformed to the binary form of 8 bits, and each bit is embedded into 1 block of the DCT transform. In total, 1024 bits could be embedded into each image. Therefore, we need 128 gray-scale values to be embedded in each image.
- Scenario 3: Embedding identical redundant bits (0 or 1) in the original image. In this part, the robustness is tested in term of BER, and we have also computed the percentage of faulty extracted bits using a voting algorithm.
- Scenario 4: Embedding random bits (0 and 1) produced by a Pseudo-Random Number-Generator in the original image.

For a fair evaluation, the robustness of watermarked schemes is usually tested using watermarked images with uniform fidelity [54, 78, 92]. Therefore, the quantization step size Δ was selected with a value close to 80 to get watermarked images with a fixed SSIM close to 0.986.

Table 6.1: Robustness and quality of the attacked images, when binary image watermarks are embedded in the original images with STDM-DCT watermarking (Scenario 1)

Attacks	**NC (SD)**	**SSIM (SD)**
Salt&Pepper (d=0.005)	0.951 (0.011)	0.868 (0.021)
Salt&Pepper (d=0.01)	0.890 (0.013)	0.770 (0.033)
Gaussian filtering (5×5)	0.998 (0.003)	0.976 (0.005)
Median filtering (3×3)	0.966 (0.027)	0.907 (0.051)
Median filtering (5×5)	0.870 (0.065)	0.813 (0.102)
Wiener filtering (3×3)	0.994 (0.008)	0.929 (0.037)
Wiener filtering (5×5)	0.917 (0.047)	0.861 (0.077)
Average filtering (3×3)	0.941 (0.028)	0.882 (0.005)
Average filtering (5×5)	0.820 (0.064)	0.775 (0.107)
FCNNDA	**0.651 (0.142)**	**0.976 (0.007)**

Figure 6.3: A binary image watermark of size 32×32

(a)Watermarked Image. (b)Salt & Pepper (d=0.01) (c)Median Filtering (5×5) (d)Wiener Filtering (5×5) (e)Average Filtering (5×5) (f)FCNNDA

Figure 6.4: Quality distortion of attacked image, when binary watermark is embedded in the original image with STDM-DCT watermarking (Scenario 1)

6.5.1.1/ SCENARIO 1

In the first scenario, a binary image watermark such as the one presented in Figure 6.3 of size 32×32 is embedded in the original image of size 512×512 based on STDM and DCT. After that, FCNNDA and other types of attacks are applied to the watermarked images, to compare the quality and robustness levels. Table 6.1 shows the level of robustness and the quality of the attacked images. STDM has good robustness against the additive noise and the filtering attacks when the density of the noise is low, or the filters window size is small. But the NC decreases when the noise density or the filters window size increases. In parallel, the quality of the attacked images is also affected after applying such type of attacks.
As shown in Table 6.1, the watermarks are affected against the Average filtering (5×5), with a NC average close to 0.820. But the quality of the attacked images is also affected with an SSIM average close to 0.775. On the other hand, the FCNNDA disturbs almost all the watermarks, with a NC average close to 0.651, while preserving a good quality of the attacked images (SSIM = 0.976). The Standard Deviation (SD) values of NC and SSIM are low, which means that most of the values are close to the average. Figure 6.4 presents the quality effect of those attacks on the watermarked image.
Figure 6.5 shows the extracted binary image watermark after applying the different types of attacks. With all the attacks except for FCNNDA, the binary image watermark could be reconstructed using an algorithm of denoising. With FCNNDA, the extracted binary image watermark is relatively different from the extracted one. Therefore, it would be difficult and somehow impossible to reconstruct the extracted binary image watermark after the FCNNDA attack.
The Histograms in Figure 6.6 show the absolute difference between the pixels values of the original images, watermarked images, and attacked images by Average filtering (5×5), and by FCNNDA. 96% of the absolute difference between the pixels values of the

(a)Salt& Pepper (d=0.01). (b)Gaussian Filtering (5×5) (c)Median Filtering (5×5) (d)Wiener Filtering (5×5) (e)Average Filtering (5×5) (f)FCNNDA

Figure 6.5: Extracted binary image watermark after applying the different types of attacks

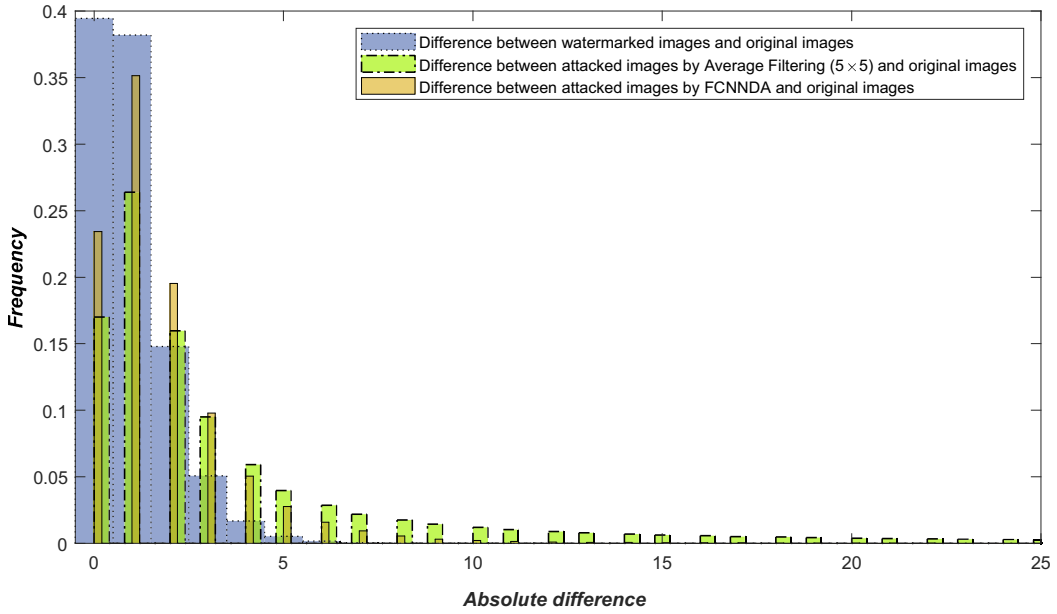


Figure 6.6: Comparing the pixels values of the original images to the watermarked images, and the attacked images by Average filtering (5×5) and FCNNDA, when the watermark is a binary image of size 32×32, and the watermarking scheme is STDM-DCT

watermarked images and the original images are distributed between 0 and 3. 89% of the absolute difference between the pixels values of the attacked images by FCNNDA and the original images are distributed between 0 and 3. 68% of the absolute difference between the pixels values of the attacked images by Average filtering (5×5) and the original images are distributed between 0 and 3. The attacked images by FCNNDA are thus closer to the original images comparing to the attacked images by Average filtering (5×5). Noting that, we have only compared the FCNNDA to the Average filtering attack because they have the lowest NC values.

Moreover, we have visually interpreted the absolute difference between the pixels values of the original images, the watermarked images, and attacked images by FCNNDA and Average filtering. The results are presented in Figure 6.7, where the watermark was embedded into the original image ”Lena” using different projection vectors. We have used different projection vectors with STDM during the test to study their impact on the effectiveness of the FCNNDA attack. The first and the second columns in Figure 6.7 contain two watermarked images ”Lena” with different projection vectors. The third and the fourth columns present zooming to the same block of the first two images for more details. The first line presents the absolute difference between the pixels values of the original image and the watermarked image. We can recognize the patches in the images, and we can notice that the watermark bits are embedded everywhere in the image. The second line

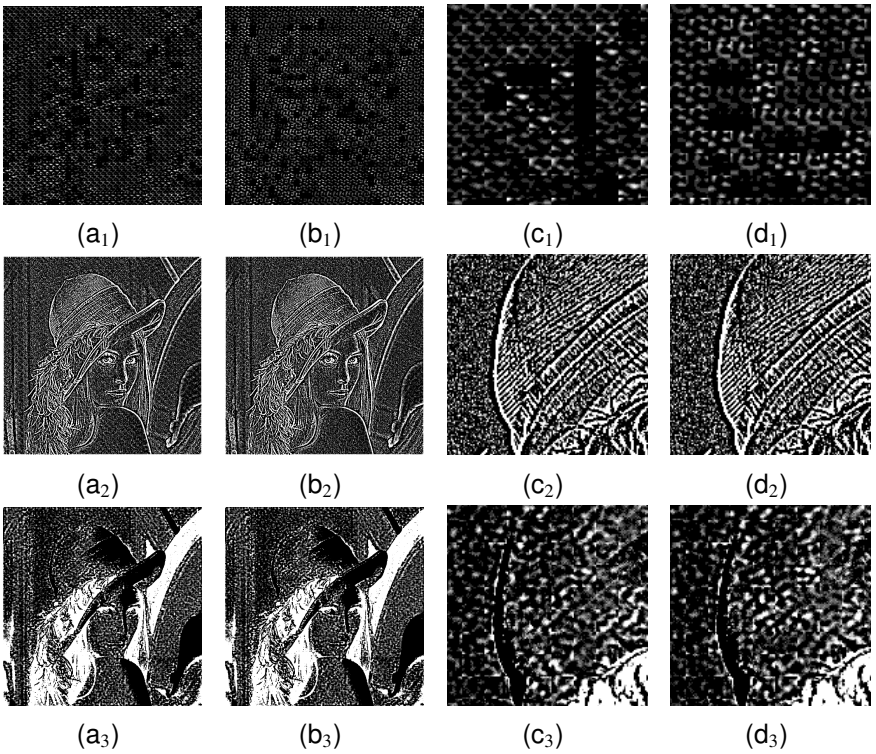


Figure 6.7: Absolute difference (a_1) between the pixels values of the original images and the watermarked images, (a_2) between the pixels values of the original images and attacked images by Average filtering (5×5), (a_3) between the pixels values of the original images and attacked images by FCNNDA, (b_i) presents the absolute difference with different projection vector for STDM, (c_i) and (d_i) present the details for (a_i) and (b_i) respectively

shows the absolute difference between the pixels values of the attacked images by Average filtering (5×5) and the original images. All the edges are changed due to the average filtering. The third line shows the absolute difference between the pixels values of the attacked images by FCNNDA and the original images. We can distinguish how the patches are changed in the images. The darken parts, and the brighten parts of the images are also changed compared to the original one. This variation destroys the watermarks that have been embedded in the original images.

6.5.1.2/ SCENARIO 2

In the second scenario, a gray-scale watermark is embedded in each original image. Each gray-scale value is converted to the binary form of 8 bits, and each bit is embedded into 1 block of the DCT transform. The robustness and the quality of the attacked images are tested in term of NCC and SSIM. 1024 bits could be embedded into each image. Hence, 128 gray-scale values are embedded to each image based on STDM and DCT.
As shown in Table 6.2, the watermarks are affected against the Average filtering (5×5), with a NCC average close to 0.418. But the quality of the attacked images is also affected with an SSIM average close to 0.774. On the other hand, the FCNNDA distroys the gray-scale watermark with a NCC average close to 0.259, while preserving a good quality

Table 6.2: Percentage of faulty bits and quality of the attacked images, when gray-scale watermark is embedded per image, and STDM with DCT is used as watermarking scheme (Scenario 2)

Attacks	**NCC (SD)**	**SSIM (SD)**
Salt&Pepper (d=0.005)	0.796 (0.073)	0.868 (0.021)
Salt&Pepper (d=0.01)	0.612 (0.086)	0.769 (0.032)
Gaussian filtering (5×5)	0.998 (0.005)	0.976 (0.005)
Median filtering (3×3)	0.878 (0.128)	0.907 (0.052)
Median filtering (5×5)	0.626 (0.234)	0.813 (0.102)
Wiener filtering (3×3)	0.975 (0.043)	0.928 (0.037)
Wiener filtering (5×5)	0.609 (0.212)	0.861 (0.077)
Average filtering (3×3)	0.836 (0.136)	0.881 (0.055)
Average filtering (5×5)	0.418 (0.225)	0.774 (0.107)
FCNNDA	**0.259 (0.198)**	**0.974 (0.008)**

(a)Watermarked Image (b)Average Filtering (5×5) (c)FCNNDA

Figure 6.8: Quality distortion of attacked images, when gray-scale watermark is embedded per image, and STDM with DCT is used as watermarking scheme (Scenario 2)

of the attacked images (SSIM = 0.974). The gray-scale watermark is highly affected comparing to the binary image watermark because the gray-scale value is converted to the binary form of 8 bits. In this case, if one bit out of 8 bits is changed after the attack, the gray-scale value will also be changed. The quality effect of Average filtering and FCNNDA attacks on the watermarked image are presented in Figure 6.8. The absolute difference between the pixels values of the original images, the watermarked images, and attacked images (by FCNNDA and Average filtering) are presented in the histograms in Figure 6.9.

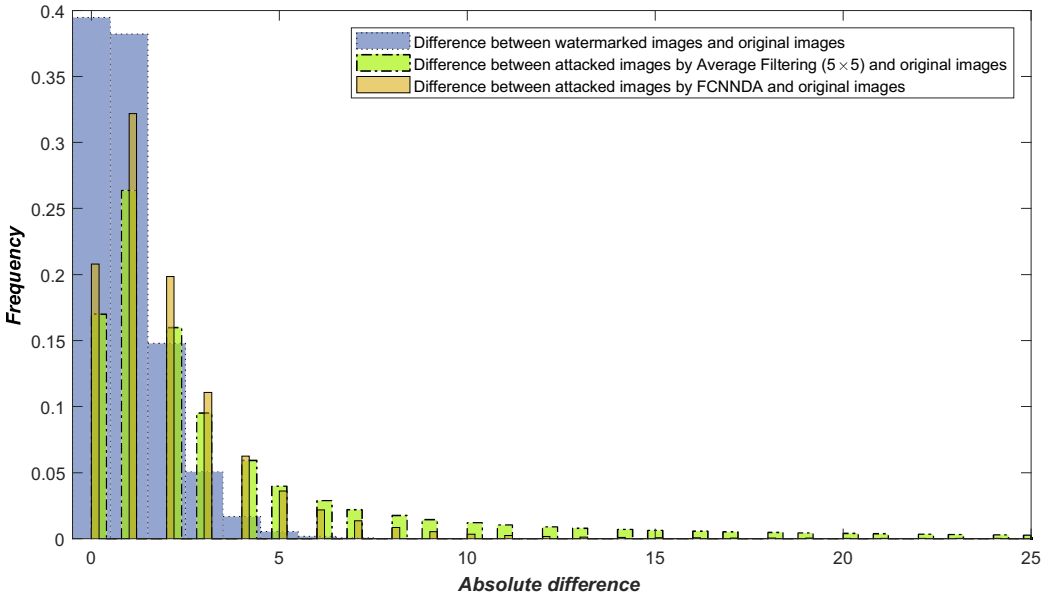


Figure 6.9: Comparing the pixels values of the original images to the watermarked images, and the attacked images by FCNNDA and Average filtering (5×5), when gray-scale watermark is embedded per image, and STDM with DCT is used as watermarking scheme

95% of the absolute difference between the pixels values of the watermarked images and the original images are distributed between 0 and 3. 87% of the absolute difference between the pixels values of the attacked images by FCNNDA and the original images are distributed between 0 and 3. 68% of the absolute difference between the pixels values of the attacked images by Average filtering (5×5) and the original images are distributed between 0 and 3. The attacked images by FCNNDA are closer to the original images comparing to the attacked images by Average filtering (5×5).

6.5.1.3/ SCENARIO 3

In the third scenario, a sequence of 1024 length 0 or 1 identical bit is embedded in each original image. The robustness and the quality of the attacked images are tested in term of BER and SSIM. We have also computed the percentage of faulty extracted bits using a voting algorithm. If the majority of extracted bits per image are wrong, the voting value will increase by 1, and in this way, the percentage value will increase as well. By this way, we could determine if the identical redundant bit could be extracted without error after applying the attacks. As shown in Table 6.3, the majority of extracted bits per image are correct. For that the total result of voting algorithm was equal to 0% faulty bits for all the attacks excepting FCNNDA. Conversely, we have got 48% faulty bits when applied the FCNNDA attack; with 48% of the images, the majority of extracted bits per image are incorrect. The average of BER is close to 0.481, and the quality of the attacked images has an SSIM value close to 0.970. The quality effect of Average filtering and FCNNDA attacks on the watermarked image are presented in Figure 6.10.
The absolute difference between the pixels values of the original images, the watermarked images, and attacked images (by FCNNDA and Average filtering) are presented in the histograms in Figure 6.11. 96% of the absolute difference between the pixels values

Table 6.3: Percentage of faulty bits and quality of the attacked images, when an identical bit is embedded per image, and STDM with DCT is used as watermarking scheme (Scenario 3)

Attacks	**% Faulty Bits**	**BER (SD)**	**SSIM (SD)**
Salt&Pepper (d=0.005)	0	0.064 (0.007)	0.876 (0.003)
Salt&Pepper (d=0.01)	0	0.132 (0.011)	0.779 (0.004)
Gaussian filtering (5×5)	0	0.007 (0.002)	0.975 (0.008)
Median filtering (3×3)	0	0.049 (0.002)	0.896 (0.004)
Median filtering (5×5)	0	0.212 (0.011)	0.781 (0.001)
Wiener filtering (3×3)	0	0.017 (0.001)	0.913 (0.005)
Wiener filtering (5×5)	0	0.160 (0.003)	0.827 (0.003)
Average filtering (3×3)	0	0.120 (0.002)	0.864 (0.003)
Average filtering (5×5)	0	0.261 (0.011)	0.731 (0.008)
FCNNDA	**48**	**0.481 (0.149)**	**0.970 (0.011)**

of the watermarked images and the original images are distributed between 0 and 3. 85% of the absolute difference between the pixels values of the attacked images by FCNNDA and the original images are distributed between 0 and 3. 69% of the absolute difference between the pixels values of the attacked images by Average filtering (5×5) and the original images are distributed between 0 and 3. The attacked images by FCNNDA are closer to the original images comparing to the attacked images by Average filtering (5×5). The extracted identical bit could be different from the embedded one with a probability close to 0.5 after applying the FCNNDA attack. This is a high value comparing to the other type of attacks, where the value of the faulty extracted bits was equal to 0.

6.5.1.4/ SCENARIO 4

In this scenario, random bits (0 and 1) are embedded in the original images. 1024 random bits are embedded in each image based on STDM and DCT. The robustness and the quality of the attacked images are tested in term of BER and SSIM. As shown in Table 6.4, the extracted bits are affected by Average filtering attack, with a BER average close to 0.241. But the quality of the attacked images is also affected with an SSIM average close to 0.774. On the other hand, the extracted bits are highly affected by the FCNNDA attack, with a BER average close to 0.483. The quality of the attacked images is also

(a)Watermarked Image (b)Average Filtering (5×5) (c)FCNNDA

Figure 6.10: Quality distortion of attacked images, when an identical bit is embedded per image, and STDM with DCT is used as watermarking scheme (Scenario 3)

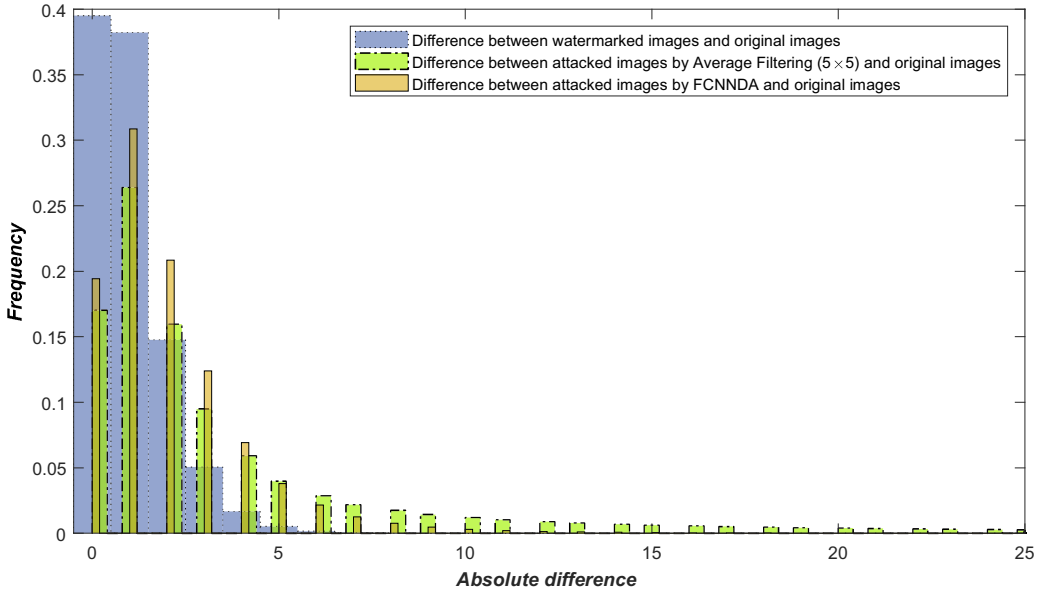


Figure 6.11: Comparing the pixels values of the original images to the watermarked images, and the attacked images by FCNNDA and Average filtering (5×5), when an identical bit is embedded per image, and STDM with DCT is used as watermarking scheme

preserved, with an SSIM average close to 0.971. Figure 6.12 presents the quality effect of Average filtering and FCNNDA attacks on the watermarked image.
The absolute difference between the pixels values of the original images, the watermarked images, and attacked images (by FCNNDA and Average filtering) are presented in the histograms in Figure 6.13. 97% of the absolute difference between the pixels values of the watermarked images and the original images are distributed between 0 and 3. 84% of the absolute difference between the pixels values of the attacked images by

Table 6.4: Robustness and quality of the attacked images, when random bits are embedded per image, and STDM with DCT is used as watermarking scheme (Scenario 4)

Attacks	**BER (SD)**	**SSIM (SD)**
Salt&Pepper (d=0.005)	0.076 (0.013)	0.869 (0.021)
Salt&Pepper (d=0.01)	0.142 (0.017)	0.770 (0.033)
Gaussian filtering (5×5)	0.001 (0.002)	0.976 (0.005)
Median filtering (3×3)	0.029 (0.028)	0.907 (0.051)
Median filtering (5×5)	0.166 (0.079)	0.812 (0.102)
Wiener filtering (3×3)	0.008 (0.012)	0.929 (0.037)
Wiener filtering (5×5)	0.139 (0.078)	0.860 (0.077)
Average filtering (3×3)	0.082 (0.043)	0.882 (0.055)
Average filtering (5×5)	0.241 (0.085)	0.774 (0.107)
FCNNDA	**0.483 (0.137)**	**0.971 (0.011)**

(a)Watermarked Image (b)Average Filtering (5×5) (c)FCNNDA

Figure 6.12: Quality distortion of attacked image, when random bits are embedded per image, and STDM with DCT is used as watermarking scheme (Scenario 4)

FCNNDA and the original images are distributed between 0 and 3. 68% of the absolute difference between the pixels values of the attacked images by Average filtering (5×5) and the original images are distributed between 0 and 3. The attacked images by FCNNDA are closer to the original images comparing to the attacked images by Average filtering (5×5). FCNNDA destroys the random embedded bits while preserving the quality of the attacked images.

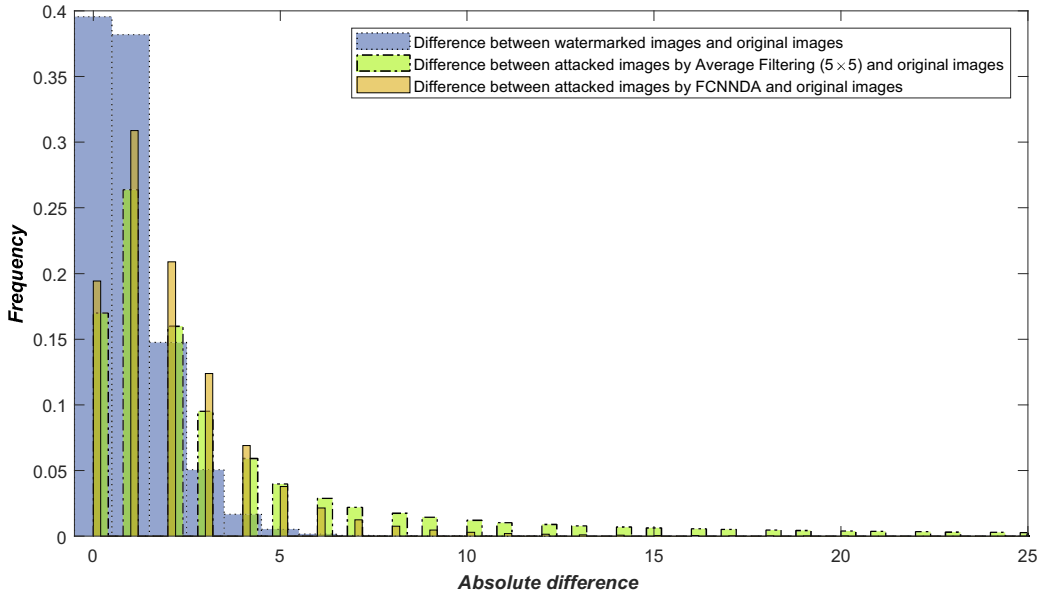


Figure 6.13: Comparing the pixels values of the original images to the watermarked images, and the attacked images by FCNNDA and Average filtering (5×5), when random bits are embedded per image, and STDM with DCT is used as watermarking scheme

6.5.2/ SS AND DWT-SVD BASED WATERMARKING

In this part of the experiments, the SS watermarking scheme was tested against the FCNNDA attack. SVD and DWT were widely used with SS to increase the level of robustness and fidelity [40, 60, 76]. Therefore, we have embedded the watermark in the original images using the DWT-SVD as follows:

- Perform 3-level DWT onto the cover image to get the four sub-bands (LL3, LH3, HL3, and HH3).
- Apply the SVD to LL3 sub-band ($A = USV^T$), and the watermark ($A_w = U_w S_w V_w^T$).
- Modify the singular values of LL3 by adding the singular values of the watermark as: S'= S+ αS_w. Alpha is the scaling factor.
- Perform the modified LL3: $LL^{new} = US'V^T$.
- Finally, apply the inverse DWT to obtain the watermarked image A_w.

The watermark was extracted as follows:

- Perform 3-level DWT onto the watermarked image A_W^* to decompose it into four sub-bands (LL3, LH3, HL3, and HH3).
- Apply SVD to LL3: $A^* = U^* S^* V^{*T}$.
- Compute: $S_W^* = \frac{S^* - S}{\alpha}$.
- Get the watermark as: $S_W^* = U_w S_W^* V_w^T$.

Figure 6.14: Gray-scale watermark of size 64×64

The tested images are watermarked using a uniform fidelity. The scaling factor α was selected with a value close to 0.15 to get watermarked images with fixed SSIM values close to 0.986.
SS is evaluated against FCNNDA with four different scenarios:

- Scenario 1: Embedding gray-scale watermark of size 64×64 in the original image.
- Scenario 2: Embedding binary image watermark of size 64×64 in the original image. (To apply the SVD on the binary watermark, we have multiplied the binary values by 255).
- Scenario 3: Embedding identical redundant bits (0 or 1) in the original image. The redundant bits are grouped in a matrix of 64×64, and the bits 0 are replaced by 64, and the bits 1 are replaced by 192, to apply the SVD on the matrix.
- Scenario 4: Embedding random bits (0 and 1) in the original image using a Pseudo-Random-Number Generator. The random bits are grouped in a matrix of 64×64 and multiplied by 255, to apply the SVD on the matrix.

Structural Similarity Index Measure (SSIM) is used to compare the quality of the watermarked images after applying the attacks. Bit Error Rate (BER) is used to compare the level of robustness when a watermark in form of bits is embedded in the original image. Normalized Correlation (NC) is used to compare the level of robustness for a binary watermark. Normalized Cross-Correlation (NCC) is used to compare the level of robustness of gray-scale watermark.

6.5.2.1/ SCENARIO 1

In the first scenario, a gray-scale watermark like the cameraman presented in Figure 6.14 of size 64×64 is embedded in the original image using the SS and DWT-SVD based watermarking. After that, FCNNDA and the other types of attacks are applied to the watermarked images, to compare the quality and robustness levels.
As shown in Table 6.5, the extracted watermark is affected when the Average filtering (5×5) is applied to the watermarked images, where the average of NCC value is close to 0.518. In parallel, the quality of the attacked images is also affected, where the average of SSIM values is close to 0.776. Conversely, the extracted watermark is highly affected by the FCNNDA attack, where the average of NCC is close to 0.152. Also, the quality of the attacked images is preserved, where the average of SSIM values is close to 0.984. The quality effect of Average filtering and FCNNDA attacks on the watermarked image are presented in Figure 6.15. FCNNDA destroys the embedded watermark while preserving a good quality of the watermarked image.
Figure 6.16 shows the extracted gray-scale watermark after applying the different types

Table 6.5: Robustness and quality of the attacked images, when gray-scale watermarks are embedded in the original images, and SS with DWT-SVD is used as watermarking scheme (Scenario-1)

Attacks	**NCC (SD)**	**SSIM (SD)**
Salt&Pepper (d=0.005)	0.952 (0.064)	0.879 (0.018)
Salt&Pepper (d=0.01)	0.875 (0.143)	0.778 (0.032)
Gaussian filtering (5×5)	0.985 (0.021)	0.984 (0.006)
Median filtering (3×3)	0.968 (0.044)	0.914 (0.052)
Median filtering (5×5)	0.861 (0.118)	0.816 (0.102)
Wiener filtering (3×3)	0.969 (0.037)	0.934 (0.038)
Wiener filtering (5×5)	0.851 (0.127)	0.862 (0.078)
Average filtering (3×3)	0.888 (0.128)	0.889 (0.056)
Average filtering (5×5)	0.518 (0.227)	0.776 (0.107)
FCNNDA	**0.136 (0.152)**	**0.986 (0.034)**

(a)Watermarked Image (b)Average Filtering (5×5) (c)FCNNDA

Figure 6.15: Quality distortion of attacked images, when gray-scale watermark is embedded in the original image, and SS with DWT-SVD is used as watermarking scheme (Scenario-1)

of attacks. With all the attacks except for FCNNDA, the gray-scale watermark could be repaired using an algorithm of denoising or filtering. With FCNNDA, the extracted gray-scale watermark is relatively different from the extracted one. Hence, it would be tricky and somehow impossible to reconstruct the extracted gray-scale watermark after the FCNNDA attack.

(a)Salt& Pepper (d=0.01) (b)Gaussian Filtering (5×5) (c)Median Filtering (5×5) (d)Wiener Filtering (5×5) (e)Average Filtering (5×5) (f)FCNNDA

Figure 6.16: Extracted gray-scale watermark after applying the different types of attacks

6.5.2.2/ SCENARIO 2

In the second scenario, a binary image watermark of size 32×32 is embedded in the original image of size 512×512 based on SS and DWT-SVD. SVD is applied to the binary watermark after multiplying the binary values by 255.
Table 6.6 shows the level of robustness and the quality of the attacked images, when binary watermarks are embedded in the original images. In this scenario SS watermarking scheme achieved good robustness against the additive noise and the filtering attacks. But the NC decreases with the Average filtering (5×5) attack with a NC average close to 0.589. But the quality of the attacked images is also affected with an SSIM average close to 0.776. On the other hand, the FCNNDA disturbs almost all the binary image watermarks, with a NC average close to 0.211, while preserving the quality of the attacked images (SSIM = 0.985). The SD values of NC and SSIM are low, which means that most of the values are close to the average. Figure 6.17 presents the quality effect of FCNNDA attack on the watermarked image.
Figure 6.18 shows the extracted binary image watermark after applying the different types of attacks. With all the attacks except for FCNNDA, the binary image watermark could

Table 6.6: Robustness and quality of the attacked images, when binary image watermarks are embedded in the original images, and SS with DWT-SVD is used as watermarking scheme (Scenario-2)

Attacks	**NC (SD)**	**SSIM (SD)**
Salt&Pepper (d=0.005)	0.989 (0.051)	0.879 (0.017)
Salt&Pepper (d=0.01)	0.957 (0.125)	0.778 (0.32)
Gaussian filtering (5 × 5)	0.999 (0.001)	0.986 (0.006)
Median filtering (3 × 3)	0.992 (0.047)	0.914 (0.051)
Median filtering (5 × 5)	0.898 (0.114)	0.816 (0.102)
Wiener filtering (3 × 3)	0.996 (0.011)	0.934 (0.038)
Wiener filtering (5 × 5)	0.896 (0.097)	0.862 (0.077)
Average filtering (3 × 3)	0.941 (0.081)	0.888 (0.056)
Average filtering (5 × 5)	0.589 (0.207)	0.776 (0.107)
FCNNDA	**0.211 (0.157)**	**0.985 (0.007)**

(a)Watermarked Image (b)Watermarked Image (Zoom) (c)FCNNDA (a)FCNNDA (Zoom)

Figure 6.17: Quality distortion of FCNNDA, when Binary watermark image is embedded in the original image, and SS with DWT-SVD is used as watermarking scheme (Scenario-2)

(a)Salt& Pepper (d=0.01) (b)Gaussian Filtering (5×5) (c)Median Filtering (5×5) (d)Wiener Filtering (5×5) (e)Average Filtering (5×5) (f)FCNNDA

Figure 6.18: Extracted binary image watermark after applying the different types of attacks

be reconstructed using an algorithm of denoising. With FCNNDA, the extracted binary image watermark is relatively different from the extracted one. Therefore, it would be difficult and somehow impossible to reconstruct the extracted binary image watermark after the FCNNDA attack.

6.5.2.3/ SCENARIO 3

In the third scenario, a sequence of 1024 length 0 or 1 identical bit is embedded in the original image. The redundant bits are grouped in a matrix of 64×64, and the bits 0 are replaced by 64, and the bits 1 are replaced by 192, to apply the SVD on the matrix. The robustness and the quality of the attacked images are tested in term of BER and SSIM. We have also computed the percentage of faulty extracted bits using a voting algorithm. If the majority of extracted bits per image are wrong, the voting value will increase by 1, and in this way, the percentage value will increase as well. Based on this scenario, we could determine if the identical redundant bit could be extracted without error after applying the attacks.
As shown in Table 6.7, the majority of extracted bits per image are correct. For that, the total result of the voting algorithm was equal to 0% faulty bits for all the attacks excepting FCNNDA. Conversely, we have got 32% faulty bits when the FCNNDA attack is applied; with 32% of the images, the majority of extracted bits per image was incorrect. The average of BER is close to 0.342, and the quality of the attacked images has an SSIM value close to 0.987. The identical embedded bit has a probability close to 0.3 to be extracted with error after applying the FCNNDA attack. This is a high value comparing to the other type of attacks, where the faulty extracted bits was equal to 0%. The quality effect of wa-

Table 6.7: Percentage of faulty bits and quality of the attacked images, when an identical bit is embedded per image, and SS with DWT-SVD is used as watermarking scheme (Scenario-3)

Attacks	**% Faulty Bits**	**BER (SD)**	**SSIM (SD)**
Salt&Pepper (d=0.005)	0	0.045 (0.182)	0.876 (0.019)
Salt&Pepper (d=0.01)	0	0.051 (0.192)	0.775 (0.031)
Gaussian filtering (5×5)	0	0.033 (0.151)	0.984 (0.007)
Median filtering (3×3)	0	0.013 (0.099)	0.912 (0.051)
Median filtering (5×5)	0	0.034 (0.162)	0.815 (0.103)
Wiener filtering (3×3)	0	0.042 (0.161)	0.932 (0.037)
Wiener filtering (5×5))	0	0.051 (0.183)	0.860 (0.076)
Average filtering (3×3)	0	0.032 (0.098)	0.886 (0.055)
Average filtering (5×5)	0	0.072 (0.234)	0.774 (0.106)
FCNNDA	**32**	**0.342 (0.241)**	**0.987 (0.008)**

(a)Watermarked Image (b)Watermarked Image (Zoom) (c)FCNNDA (a)FCNNDA (Zoom)

Figure 6.19: Quality distortion of FCNNDA, when identical bit is embedded per image, and SS with DWT-SVD is used as watermarking scheme (Scenario-3)

termarked image and FCNNDA attack on the watermarked image are presented in Figure 6.19.

Table 6.8: Robustness and quality of the attacked images, when random bits are embedded per image, and SS with DWT-SVD is used as watermarking scheme (Scenario-4)

Attacks	**BER (SD)**	**SSIM (SD)**
Salt&Pepper (d=0.005)	0.001 (0.002)	0.868 (0.020)
Salt&Pepper (d=0.01)	0.002 (0.008)	0.771 (0.032)
Gaussian filtering (5×5)	0.000 (0.001)	0.976 (0.006)
Median filtering (3×3)	0.002 (0.001)	0.908 (0.049)
Median filtering (5×5)	0.005 (0.014)	0.814 (0.099)
Wiener filtering (3×3)	0.001 (0.001)	0.931 (0.037)
Wiener filtering (5×5)	0.002 (0.001)	0.859 (0.076)
Average filtering (3×3)	0.003 (0.002)	0.883 (0.054)
Average filtering (5×5)	0.021 (0.017)	0.775 (0.105)
FCNNDA	**0.537 (0.076)**	**0.985 (0.008)**

(a)Watermarked Image (b)Watermarked Image (Zoom) (c)FCNNDA (a)FCNNDA (Zoom)

Figure 6.20: Quality distortion of FCNNDA, when random bits are embedded per image, and SS with DWT-SVD is used as watermarking scheme (Scenario-4)

6.5.2.4/ SCENARIO 4

In the last scenario, random bits (0 and 1) are embedded in the original images. The random bits are grouped in a matrix of 64×64 and multiplied by 255, to apply the SVD on the matrix. The robustness and the quality of the attacked images are tested in term of BER and SSIM.
As shown in Table 6.8, the extracted bits are not affected by the additive noise and filtering attacks. On the other hand, the extracted bits are highly affected by the FCNNDA attack, with a BER average close to 0.537. The quality of the attacked images is also preserved, with an SSIM average close to 0.985. Figure 6.20 presents the quality effect of watermarked image and FCNNDA attack on the watermarked image.

6.6/ CONCLUSION

This chapter presented an evaluation of digital images watermarking against a Fully Convolutional Neural Network Denoising Attack (FCNNDA). STDM and SS watermarking schemes are examined against FCNNDA using different scenarios in the frequency domain. Several types of watermarks are embedded during the test, such as binary watermarks, one redundant bit, random bits, and gray-scale watermarks. FCNNDA was also compared to other types of attacks to examine the difference in terms of quality and robustness. The experimental results confirmed that the FCNNDA could be considered as a harmful attack. It could destroy the embedded watermarks while preserving a good quality of the attacked images.

IV

Conclusion & Perspectives

7

CONCLUSION & PERSPECTIVES

7.1/ CONCLUSION

In this thesis, we have proposed and improved blind digital watermarking techniques for PDF documents and images in terms of transparency, robustness, and security. The x-coordinates values of the characters in a PDF document are non-constant and could be used as cover work to embed the watermark. Each bit of the watermark is embedded into a group of x-coordinates values based on the STDM watermarking scheme.
Besides imperceptibility and robustness, security is an important requirement for watermarking schemes. While increasing the security, we guarantee that the embedded watermark is safe against an opponent, whose aim is to estimate the private key. STDM is based on a projection vector that is used as a secret key during the embedding and decoding process. The observation of several watermarked signals can provide sufficient information for an opponent to estimate the projection vector by using Blind Source Separation (BSS) techniques. Therefore, we have proposed the CAR-STDM (Component Analysis Resistant-STDM) watermarking method. The experimental results show that the CAR-STDM achieves the security against the BSS techniques and preserves its transparency and robustness against noise attacks.
STDM is also largely vulnerable to the Fixed Gain Attack (FGA). In this type of attack, the received signal is indeed multiplied by a gain factor, which scales the watermark vector and shifts it away from its original quantization cell. Therefore, we have proposed the N-STDM watermarking method. Experiments on PDF documents and real images in the spatial and frequency domain have shown that our proposed method achieved significant robustness against the FGA attack with a perceptual advantage and better robustness against common signals processing attacks, geometric attacks, and image filtering attacks comparing to other related methods. Moreover, the correctness of the approach has been completely proven.
Denoising is one of the active topics in image processing that aims to recover clean images from noisy observations. Diverse models such as Convolutional Neural Network (CNN) are exploited for modeling image priors for denoising. CNN has a suitable denoising performance, and it could be harmful to watermarked images. It is the first time that we propose Deep Learning to attack watermarked images. We have presented and evaluated the effect of a Fully Convolutional Neural Network Denoising Attack (FCNNDA) on watermarked images. We have evaluated such type of denoising against SS and STDM watermarking schemes, and whether it could be used as a new type of attack in digital watermarking. FCNNDA was also compared to other types of attacks to examine the difference in terms of quality and robustness. The experimental results confirmed

that the FCNNDA could be considered as a harmful attack. The effectiveness of FCNNDA surpasses the traditional watermarking attacks because It destroys the embedded watermarks while preserving the quality of the attacked images.

7.2/ PERSPECTIVES

The robustness is one of the main requirements for digital watermarking. We have shown the harmful effect of a Fully Convolutional Neural Network (FCNN), as a denoising attack, on the watermarked images. Therefore, we plan to integrate deep learning with digital watermarking to enhance the robustness against such type of attacks.
The first question that comes to the mind is how to explore the digital watermarking system with deep learning? Which deep learning model would make the tradeoff between the transparency, robustness, and security as watermarking did? The first step to be done is to train a simple model that could embed and extract a watermark without error, with a good level of transparency. What would be the appropriate parameters' values for the best perceptual similarity between the watermarked work and the original one?
After that, we should train such a model to resist specific types of attacks, such as image filtering attacks, noise attacks, and denoising attacks. Would this system achieve the robustness against varieties of attacks, and especially against new types of attacks such as the FCNNDA attack?
The robustness and transparency constraints are two opposite objectives. Therefore, during the experimental tests, two threshold levels should be recognized. The first one should be for the transparency tests and the second one for the robustness experimental tests. What would be the best range of thresholds that will achieve sufficient robustness against varieties of attacks while preserving a good level of transparency?
In addition to the traditional watermarking attacks, new types of attacks could be created based on deep learning. Therefore, another study should be maid around the types of attacks that could be inspired and created from deep learning. If exist, how could we improve the system to resist such type of attacks?
The security is also the main topic for this study. The created model will fail without security, even if it achieves a good level of transparency and robustness. Without security, an adversary would be able to break the model, and later on, remove, delete, or modify the embedded watermark. What would be the main map and topology to secure the system?
We should also take into consideration the capacity of the created model. What would be the amount of data that could be embedded in the cover work? Specifically, what would be the amount of information that could be hidden in the cover work without perceptible distortion, while preserving good robustness against several types of attacks?

PUBLICATIONS

JOURNAL PAPERS

- Makram Hatoum, Jean-François Couchot and Rony Darazi. *"Normalized Blind STDM Watermarking Scheme for Images and PDF Documents Robust against Fixed Gain Attack"*. In **Multimedia Tools and Applications** (Nov 2019).

CONFERENCE PAPERS

- Makram Hatoum, Rony Darazi and Jean-François Couchot. *"Blind PDF Document Watermarking Robust Against PCA and ICA Attacks"*. In **Proceedings of the 15th International Joint Conference on e-Business and Telecommunications** - Volume 2 SECRYPT: SECRYPT, (2018), INSTICC, SciTePress, pp. 420–427.
- Makram Hatoum, Rony Darazi and Jean-François Couchot. *"Blind Image Watermarking using Normalized STDM robust against Fixed Gain Attack"*. In **2018 IEEE International Multidisciplinary Conference on Engineering Technology (IMCET)** (Nov 2018), pp. 1–6.

SUBMITTED PAPERS

- Makram Hatoum, Jean-François Couchot, Raphaël Couturier and Rony Darazi. *"Using Deep Learning for Image Watermarking Attack"*. In **Signal Processing: Image Communication**. Submitted in November 27, 2019

A

CORRECTION PROOF OF STDM

Let us evaluate $Q_{m'}\left(y^T p, \Delta\right) = Q_{m'}\left(x^T p - r_{x^T p} + \frac{\Delta}{2}, \Delta\right)$.

First of all:

$$Q_{m'}\left(x^T p - r_{x^T p} + \frac{\Delta}{2}, \Delta\right) = Q_{m'}\left(q_{x^T p}\Delta + d_m, \Delta\right) \text{ thanks to equation } (5.21)$$

$$= round\left(\frac{q_{x^T p}\Delta + d_m - d_{m'}}{\Delta}\right)\Delta + d_{m'}$$

$$= round\left(\frac{q_{x^T p}\Delta + \frac{\Delta}{2}(m - m')}{\Delta}\right)\Delta + d_{m'}$$

$$= round\left(q_{x^T p} + \frac{m - m'}{2}\right)\Delta + d_{m'}$$

We then have

$$\begin{aligned}
\hat{m} &= \underset{m' \in \{0,1\}}{\arg\min} \left| y^T p - Q_{m'}\left(y^T p, \Delta\right)\right| \\
&= \underset{m' \in \{0,1\}}{\arg\min} \left| x^T p - r_{x^T p} + \frac{\Delta}{2} - round\left(q_{x^T p} + \frac{m - m'}{2}\right)\Delta - d_{m'})\right| \\
&= \underset{m' \in \{0,1\}}{\arg\min} \left| q_{x^T p}\Delta + d_m - round\left(q_{x^T p} + \frac{m - m'}{2}\right)\Delta - d_{m'}\right| \\
&= \underset{m' \in \{0,1\}}{\arg\min} \left| \left(q_{x^T p} + \frac{m - m'}{2}\right)\Delta - round\left(q_{x^T p} + \frac{m - m'}{2}\right)\Delta\right| \\
&= \underset{m' \in \{0,1\}}{\arg\min} \left| \left(q_{x^T p} + \frac{m - m'}{2}\right) - round\left(q_{x^T p} + \frac{m - m'}{2}\right)\right| \qquad \text{(A.1)}
\end{aligned}$$

B

CORRECTION PROOF OF N-STDM

First of all, let us identify $y^T p$:

$$y^T p = \left(x + \left(\|x\| \, Q_m \left(\frac{x^T p}{\|x\|}, \Delta \right) - x^T p \right) p \right)^T p$$
$$= x^T p + \left(\|x\| \, Q_m \left(\frac{x^T p}{\|x\|}, \Delta \right) - x^T p \right) p^T p.$$

Since p is a normalized vector, $p^T p$ is 1. Thus,

$$y^T p = \|x\| \, Q_m \left(\frac{x^T p}{\|x\|}, \Delta \right)$$
$$= \|x\| \left(\frac{x^T p}{\|x\|} - r_{\frac{x^T p}{\|x\|}} + \frac{\Delta}{2} \right) \text{ thanks to equation } (5.22)$$
$$= \|x\| \left(q_{\frac{x^T p}{\|x\|}} \Delta + d_m \right)$$
$$= x^T p - \|x\| \, . r_{\frac{x^T p}{\|x\|}} + \frac{\Delta}{2} \|x\|$$

Hence,

$$y = x - \|x\| \, . r_{\frac{x^T p}{\|x\|}} . p + \frac{\Delta}{2} \|x\| \, . \tag{B.1}$$

Let us now evaluate $Q_{m'} \left(\frac{y^T p}{\|y\|}, \Delta \right)$.

$$Q_{m'} \left(\frac{y^T p}{\|y\|}, \Delta \right) = round \left(\frac{\frac{y^T p}{\|y\|} - d_{m'}}{\Delta} \right) \Delta + d_{m'}$$

$$= round\left(\frac{x^T p - \|x\| . r_{\frac{x^T p}{\|x\|}} + \frac{\Delta}{2}\|x\|}{\frac{\|y\|}{\Delta}} - d_{m'}\right)\Delta + d_{m'}$$

$$= round\left(\frac{\|x\|\,(q_{\frac{x^T p}{\|x\|}}\Delta + d_m)}{\frac{\|y\|}{\Delta}} - d_{m'}\right)\Delta + d_{m'}$$

$$= round\left(\frac{\|x\|\, q_{\frac{x^T p}{\|x\|}}\Delta + \|x\|\, d_m - \|y\|\, d_{m'}}{\Delta\|y\|}\right)\Delta + d_{m'}$$

$$= round\left(\frac{\|x\|\, q_{\frac{x^T p}{\|x\|}}\Delta + \|x\|\,\frac{\Delta}{4}(2m-1) - \|y\|\,\frac{\Delta}{4}(2m'-1)}{\Delta\|y\|}\right)\Delta + \frac{\Delta}{4}(2m'-1)$$

$$= round\left(\frac{\|x\|\, q_{\frac{x^T p}{\|x\|}} + \|x\|\,\frac{2m-1}{4} - \|y\|\,\frac{2m'-1}{4}}{\|y\|}\right)\Delta + \frac{\Delta}{4}(2m'-1)$$

$$= round\left(\frac{\|x\|}{\|y\|} q_{\frac{x^T p}{\|x\|}} + \frac{\|x\|}{\|y\|}\frac{2m-1}{4} - \frac{2m'-1}{4}\right)\Delta + \frac{\Delta}{4}(2m'-1).$$

Let's go back to the definition 5.23 of $\hat{m}$ which becomes:

$$\begin{aligned}
\hat{m} = \underset{m' \in \{0,1\}}{\arg\min} & \left| \|x\| \left(q_{\frac{x^T p}{\|x\|}} \Delta + \frac{\Delta}{4}(2m-1) \right) \right. \\
& \left. - \left\|y\right\| \left(round \left(\frac{\|x\|}{\left\|y\right\|} q_{\frac{x^T p}{\|x\|}} + \frac{\|x\|}{\left\|y\right\|} \frac{2m-1}{4} - \frac{2m'-1}{4} \right) \Delta + \frac{\Delta}{4}(2m'-1) \right) \right| \\
= \underset{m' \in \{0,1\}}{\arg\min} & \left| \|x\| \left(q_{\frac{x^T p}{\|x\|}} + \frac{2m-1}{4} \right) \right. \\
& \left. - \left\|y\right\| \left(round \left(\frac{\|x\|}{\left\|y\right\|} q_{\frac{x^T p}{\|x\|}} + \frac{\|x\|}{\left\|y\right\|} \frac{2m-1}{4} - \frac{2m'-1}{4} \right) + \frac{2m'-1}{4} \right) \right| \\
= \underset{m' \in \{0,1\}}{\arg\min} & \left| \frac{\|x\|}{\left\|y\right\|} . q_{\frac{x^T p}{\|x\|}} + \frac{\|x\|}{\left\|y\right\|} . \frac{2m-1}{4} - \frac{2m'-1}{4} \right. \\
& \left. - round \left(\frac{\|x\|}{\left\|y\right\|} q_{\frac{x^T p}{\|x\|}} + \frac{\|x\|}{\left\|y\right\|} \frac{2m-1}{4} - \frac{2m'-1}{4} \right) \right| .
\end{aligned}$$

One can first notice that what is inside the absolute value is close to 0 since it has the form $X - round(X)$, where $X \in \mathbb{R}$.

Let us now evaluate $\frac{\|x\|}{\left\|y\right\|}$. Thanks to equation (B.1), we have:

$$\begin{aligned}
\left\|y\right\| &\leq \|x\| + \|x\| . r_{\frac{x^T p}{\|x\|}} . \left\|p\right\| + \frac{\Delta}{2}\|x\| \\
&\leq \|x\| \left(1 + r_{\frac{x^T p}{\|x\|}} + \frac{\Delta}{2} \right) \\
&\leq \|x\| \left(1 + \frac{3\Delta}{2} \right).
\end{aligned}$$

In all what follows, x and p are supposed to only have postive or null values and Δ is assumed to be less than 2.
Again, thanks to equation (B.1),

$$\begin{aligned}
y_i &= x_i - \|x\| . r_{\frac{x^T p}{\|x\|}} . p_i + \frac{\Delta}{2}\|x\| . p_i \\
y_i &> x_i - \|x\| . \Delta . p_i + \frac{\Delta}{2}\|x\| . p_i \text{ since } r < \Delta
\end{aligned}$$

$$y_i \quad > \quad x_i - \frac{\Delta}{2}\|x\| \,.p_i \text{ which is postive if } \Delta \text{ is sufficently small.}$$

Thus

$$\begin{aligned} \left\|y\right\| &\geq \left\|x - \frac{\Delta}{2}\|x\| \,.p\right\| \\ &\geq \|x\|\left(1 - \frac{\Delta}{2}\right). \end{aligned}$$

Finally,

$$\frac{2}{2+3\Delta} \leq \frac{\|x\|}{\left\|y\right\|} \leq \frac{2}{2-\Delta}, \text{ for } \Delta < 2 \text{ sufficiently small.} \tag{B.2}$$

Next, if Δ is tiny, then r_s is too; consequently, q_s will have a very large value. Equation (B.1) implies that y (resp. $\left\|y\right\|$) is close to x (resp. $\|x\|$). In this context, $\frac{\|x\|}{\left\|y\right\|}.q_{\frac{x^T p}{\|x\|}}$ is significantly larger than $\frac{\|x\|}{\left\|y\right\|}.\frac{2m-1}{4} - \frac{2m'-1}{4}$.

BIBLIOGRAPHY

[1] BLUM, L., BLUM, M., AND SHUB, M. **A simple unpredictable pseudo-random number generator**. *SIAM Journal on computing 15*, 2 (1986), 364–383.

[2] WATSON, A. B. **DCT quantization matrices visually optimized for individual images**. In *Human vision, visual processing, and digital display IV* (1993), vol. 1913, International Society for Optics and Photonics, pp. 202–217.

[3] COX, I. J., KILIAN, J., LEIGHTON, T., AND SHAMOON, T. **A secure, robust watermark for multimedia**. In *Information Hiding* (Berlin, Heidelberg, 1996), R. Anderson, Ed., Springer Berlin Heidelberg, pp. 185–206.

[4] BARTON, J. M. **Method and apparatus for embedding authentication information within digital data**, July 8 1997. US Patent 5,646,997.

[5] COX, I. J., KILIAN, J., LEIGHTON, F. T., AND SHAMOON, T. **Secure spread spectrum watermarking for multimedia**. *IEEE Transactions on Image Processing 6*, 12 (Dec 1997), 1673–1687.

[6] PUN, T., AND OTHERS. **Rotation, Translation and Scale Invariant Digital Image Watermarking**. In *icip* (1997), IEEE, p. 536.

[7] SWANSON, M. D., BIN ZHU, AND TEWFIK, A. H. **Data hiding for video-in-video**. In *Proceedings of International Conference on Image Processing* (Oct 1997), vol. 2, pp. 676–679 vol.2.

[8] BARNI, M., BARTOLINI, F., CAPPELLINI, V., AND PIVA, A. **A DCT-domain system for robust image watermarking**. *Signal Processing 66*, 3 (1998), 357 – 372.

[9] DUGAD, R., RATAKONDA, K., AND AHUJA, N. **A new wavelet-based scheme for watermarking images**. In *Proceedings 1998 International Conference on Image Processing. ICIP98 (Cat. No.98CB36269)* (Oct 1998), vol. 2, pp. 419–423 vol.2.

[10] LANGELAAR, G. C., LAGENDIJK, R. L., AND BIEMOND, J. **Removing spatial spread spectrum watermarks by non-linear filtering**. In *9th European Signal Processing Conference (EUSIPCO 1998)* (Sep. 1998), pp. 1–4.

[11] LINNARTZ, J. P. M. G., AND VAN DIJK, M. **Analysis of the Sensitivity Attack against Electronic Watermarks in Images**. In *Information Hiding* (Berlin, Heidelberg, 1998), D. Aucsmith, Ed., Springer Berlin Heidelberg, pp. 258–272.

[12] XIA, X.-G., BONCELET, C. G., AND ARCE, G. R. **Wavelet transform based watermark for digital images**. *Opt. Express 3*, 12 (Dec 1998), 497–511.

[13] HYVARINEN, A. **Fast and robust fixed-point algorithms for independent component analysis**. *IEEE transactions on Neural Networks 10*, 3 (1999), 626–634.

[14] RAMKUMAR, M., AKANSU, A. N., AND ALATAN, A. A. **A robust data hiding scheme for images using DFT**. In *Proceedings 1999 International Conference on Image Processing (Cat. 99CH36348)* (Oct 1999), vol. 2, pp. 211–215 vol.2.

[15] EJIMA, M., AND MIYAZAKI, A. **A wavelet-based watermarking for digital images and video**. *IEICE Transactions on Fundamentals of Electronics, Communications and Computer Sciences 83*, 3 (2000), 532–540.

[16] HERNANDEZ, J. R., AMADO, M., AND PEREZ-GONZALEZ, F. **DCT-domain watermarking techniques for still images: detector performance analysis and a new structure**. *IEEE Transactions on Image Processing 9*, 1 (Jan 2000), 55–68.

[17] LANGELAAR, G. C., SETYAWAN, I., AND LAGENDIJK, R. L. **Watermarking digital image and video data. A state-of-the-art overview**. *IEEE Signal Processing Magazine 17*, 5 (Sep. 2000), 20–46.

[18] LEE, Y. **High capacity image steganographic model**. *IEE Proceedings - Vision, Image and Signal Processing 147* (June 2000), 288–294(6).

[19] PEREIRA, S., AND PUN, T. **Robust template matching for affine resistant image watermarks**. *IEEE Transactions on Image Processing 9*, 6 (June 2000), 1123–1129.

[20] CHEN, B., AND WORNELL, G. W. **Quantization index modulation: A class of provably good methods for digital watermarking and information embedding**. *IEEE Transactions on Information Theory 47*, 4 (2001), 1423–1443.

[21] CHEN, B., AND WORNELL, G. W. **Quantization index modulation methods for digital watermarking and information embedding of multimedia**. *Journal of VLSI signal processing systems for signal, image and video technology 27*, 1-2 (2001), 7–33.

[22] KUNDUR, D., AND HATZINAKOS, D. **Diversity and attack characterization for improved robust watermarking**. *IEEE Transactions on Signal Processing 49*, 10 (Oct 2001), 2383–2396.

[23] LIN, C. ., WU, M., BLOOM, J. A., COX, I. J., MILLER, M. L., AND LUI, Y. M. **Rotation, scale, and translation resilient watermarking for images**. *IEEE Transactions on Image Processing 10*, 5 (May 2001), 767–782.

[24] SOLACHIDIS, V., AND PITAS, L. **Circularly symmetric watermark embedding in 2-D DFT domain**. *IEEE Transactions on Image Processing 10*, 11 (Nov 2001), 1741–1753.

[25] SERDEAN, C. V., AMBROZE, M. A., TOMLINSON, M., AND WADE, G. **Combating geometrical attacks in a DWT based blind video watermarking system**. In *International Symposium on VIPromCom Video/Image Processing and Multimedia Communications* (June 2002), pp. 263–266.

[26] DEGUILLAUME, F., VOLOSHYNOVSKIY, S., AND PUN, T. **Secure hybrid robust watermarking resistant against tampering and copy attack**. *Signal Processing 83*, 10 (2003), 2133 – 2170.

[27] RAVAL, M. S., AND REGE, P. P. **Discrete wavelet transform based multiple watermarking scheme**. In *TENCON 2003. Conference on Convergent Technologies for Asia-Pacific Region* (Oct 2003), vol. 3, pp. 935–938 Vol.3.

[28] XIANGUI KANG, JIWU HUANG, YUN Q SHI, AND YAN LIN. **A DWT-DFT composite watermarking scheme robust to both affine transform and jpeg compression**. *IEEE Transactions on Circuits and Systems for Video Technology 13*, 8 (Aug 2003), 776–786.

[29] BARTOLINI, F., BARNI, M., AND PIVA, A. **Performance analysis of ST-DM watermarking in presence of nonadditive attacks**. *IEEE Transactions on Signal Processing 52*, 10 (2004), 2965–2974.

[30] CHAN, C.-K., AND CHENG, L. **Hiding data in images by simple LSB substitution**. *Pattern Recognition 37*, 3 (2004), 469 – 474.

[31] GANIC, E., AND ESKICIOGLU, A. M. **Robust DWT-SVD domain image watermarking: embedding data in all frequencies**. In *Proceedings of the 2004 Workshop on Multimedia and Security* (2004), ACM, pp. 166–174.

[32] HUANG, F., AND GUAN, Z.-H. **A hybrid SVD-DCT watermarking method based on lpsnr**. *Pattern Recognition Letters 25*, 15 (2004), 1769 – 1775.

[33] HYVÄRINEN, A., KARHUNEN, J., AND OJA, E. **Independent component analysis**, vol. 46. John Wiley & Sons, 2004.

[34] WANG, Z., BOVIK, A. C., SHEIKH, H. R., SIMONCELLI, E. P., AND OTHERS. **Image quality assessment: from error visibility to structural similarity**. *IEEE transactions on image processing 13*, 4 (2004), 600–612.

[35] LI, Q., AND COX, I. J. **Using perceptual models to improve fidelity and provide invariance to valumetric scaling for quantization index modulation watermarking**. In *Acoustics, Speech, and Signal Processing, 2005. Proceedings.(ICASSP'05). IEEE International Conference on* (2005), vol. 2, IEEE, pp. ii–1.

[36] LICKS, V., AND JORDAN, R. **Geometric attacks on image watermarking systems**. *IEEE multimedia*, 3 (2005), 68–78.

[37] LICKS, V., AND JORDAN, R. **Geometric attacks on image watermarking systems**. *IEEE multimedia*, 3 (2005), 68–78.

[38] PÉREZ-GONZÁLEZ, F., MOSQUERA, C., BARNI, M., AND ABRARDO, A. **Rational dither modulation: A high-rate data-hiding method invariant to gain attacks**. *IEEE Transactions on Signal Processing 53*, 10 (2005), 3960–3975.

[39] POTDAR, V. M., HAN, S., AND CHANG, E. **A survey of digital image watermarking techniques**. In *Industrial Informatics, 2005. INDIN'05. 2005 3rd IEEE International Conference on* (2005), IEEE, pp. 709–716.

[40] SVERDLOV, A., DEXTER, S., AND ESKICIOGLU, A. M. **Robust DCT-SVD domain image watermarking for copyright protection: Embedding data in all frequencies**. In *2005 13th European Signal Processing Conference* (Sep. 2005), pp. 1–4.

[41] BAS, P., AND HURRI, J. **Vulnerability of dm watermarking of non-iid host signals to attacks utilising the statistics of independent components**. *IEE Proceedings-Information Security 153*, 3 (2006), 127–139.

[42] CHUHONG FEI, KUNDUR, D., AND KWONG, R. H. **Analysis and design of secure watermark-based authentication systems**. *IEEE Transactions on Information Forensics and Security 1*, 1 (March 2006), 43–55.

[43] ENPING LI, HUAQING LIANG, AND XINXIN NIU. **Blind Image Watermarking Scheme Based on Wavelet Tree Quantization Robust to Geometric Attacks**. In *2006 6th World Congress on Intelligent Control and Automation* (June 2006), vol. 2, pp. 10256–10260.

[44] LI, Q., DOERR, G., AND COX, I. J. **Spread Transform Dither Modulation using a Perceptual Model**. In *2006 IEEE Workshop on Multimedia Signal Processing* (Oct 2006), pp. 98–102.

[45] WANG, X., AND ZHAO, H. **A novel Synchronization Invariant Audio Watermarking Scheme Based on DWT and DCT**. *IEEE Transactions on Signal Processing 54*, 12 (Dec 2006), 4835–4840.

[46] ZHU, X. **Image-adaptive spread transform dither modulation using human visual model**. In *Computational Intelligence and Security, 2006 International Conference on* (2006), vol. 2, IEEE, pp. 1571–1574.

[47] COX, I., MILLER, M., BLOOM, J., FRIDRICH, J., AND KALKER, T. **Digital watermarking and steganography**. Morgan kaufmann, 2007.

[48] LI, Q., AND COX, I. J. **Improved Spread Transform Dither Modulation using a Perceptual Model: Robustness to Amplitude Scaling and JPEG Compression**. In *2007 IEEE International Conference on Acoustics, Speech and Signal Processing - ICASSP '07* (April 2007), vol. 2, pp. II–185–II–188.

[49] LI, Q., YUAN, C., AND ZHONG, Y. **Adaptive DWT-SVD Domain Image Watermarking Using Human Visual Model**. In *The 9th International Conference on Advanced Communication Technology* (Feb 2007), vol. 3, pp. 1947–1951.

[50] LIU, J.-C., LIN, C.-H., KUO, L.-C., AND CHANG, J.-C. **Robust Multi-scale Full-Band Image Watermarking for Copyright Protection**. In *New Trends in Applied Artificial Intelligence* (Berlin, Heidelberg, 2007), H. G. Okuno and M. Ali, Eds., Springer Berlin Heidelberg, pp. 176–184.

[51] ADOBE, J. **Document management—portable document 493 format—part 1: Pdf 1.7**, 2008.

[52] POR, L. Y., AND DELINA, B. **Information hiding: A new approach in text steganography**. In *WSEAS International Conference. Proceedings. Mathematics and Computers in Science and Engineering* (2008), no. 7, World Scientific and Engineering Academy and Society.

[53] WANG, C., AND TSAI, W. **Data hiding in pdf files and applications by imperceivable modifications of pdf object parameters**. In *Proceedings of 2008 conference on computer vision, graphics and image processing, Ilan, Taiwan, 8p* (2008), pp. 1–6.

[54] YU, D., MA, L., WANG, G., AND LU, H. **Adaptive spread-transform dither modulation using an improved luminance-masked threshold**. In *Image Processing, 2008. ICIP 2008. 15th IEEE International Conference on* (2008), IEEE, pp. 449–452.

[55] ASLANTAS, V. **An optimal robust digital image watermarking based on SVD using differential evolution algorithm**. *Optics Communications 282*, 5 (2009), 769 – 777.

[56] MANSOURI, A., MAHMOUDI AZNAVEH, A., AND TORKAMANI AZAR, F. **SVD-based digital image watermarking using complex wavelet transform**. *Sadhana 34*, 3 (Jun 2009), 393–406.

[57] RAGHAVENDRA, K., AND CHETAN, K. R. **A blind and robust watermarking scheme with scrambled watermark for video authentication**. In *2009 IEEE International Conference on Internet Multimedia Services Architecture and Applications (IMSAA)* (Dec 2009), pp. 1–6.

[58] SANTHI, V., AND THANGAVELU, A. **DWT-SVD combined full band robust watermarking technique for color images in YUV color space**. *International Journal of Computer Theory and Engineering 1*, 4 (2009), 424.

[59] DING, Y., ZHENG, X., ZHAO, Y., AND LIU, G. **A Video Watermarking Algorithm Resistant to Copy Attack**. In *2010 Third International Symposium on Electronic Commerce and Security* (July 2010), pp. 289–292.

[60] LAI, C., AND TSAI, C. **Digital Image Watermarking Using Discrete Wavelet Transform and Singular Value Decomposition**. *IEEE Transactions on Instrumentation and Measurement 59*, 11 (Nov 2010), 3060–3063.

[61] LEE, I.-S., AND TSAI, W.-H. **A new approach to covert communication via PDF files**. *Signal Processing 90*, 2 (2010), 557 – 565.

[62] LI, J. **Robust image watermarking scheme against geometric attacks using a computer-generated hologram**. *Applied optics 49*, 32 (2010), 6302–6312.

[63] LIN, S. D., SHIE, S.-C., AND GUO, J. **Improving the robustness of DCT-based image watermarking against JPEG compression**. *Computer Standards & Interfaces 32*, 1 (2010), 54 – 60.

[64] SONG, C., SUDIRMAN, S., MERABTI, M., AND LLEWELLYN-JONES, D. **Analysis of Digital Image Watermark Attacks**. In *2010 7th IEEE Consumer Communications and Networking Conference* (Jan 2010), pp. 1–5.

[65] FILLER, T., PEVNÝ, T., CRAVER, S., AND KER, A. D., Eds. *Information Hiding - 13th International Conference, IH 2011, Prague, Czech Republic, May 18-20, 2011, Revised Selected Papers* (2011), vol. 6958 of *Lecture Notes in Computer Science*, Springer.

[66] BAS, P., FILLER, T., AND PEVNÝ, T. **"break Our Steganographic System": The Ins and Outs of Organizing BOSS**. In *Information Hiding* (Berlin, Heidelberg, 2011), T. Filler, T. Pevný, S. Craver, and A. Ker, Eds., Springer Berlin Heidelberg, pp. 59–70.

[67] LI, X., LIU, J., SUN, J., YANG, X., AND LIU, W. **Step-projection-based spread transform dither modulation**. *IET information security 5*, 3 (2011), 170–180.

[68] ALIZADEH-FAHIMEH, F., CANCEILL-NICOLAS, N., DABKIEWICZ-SEBASTIAN, S., AND VANDEVENNE-DIEDERIK, D. **Using Steganography to hide messages inside PDF files**.

[69] ANSARI, R., DEVANALAMATH, M. M., MANIKANTAN, K., AND RAMACHANDRAN, S. **Robust Digital image Watermarking Algorithm in DWT-DFT-SVD domain for color images**. In *2012 International Conference on Communication, Information Computing Technology (ICCICT)* (Oct 2012), pp. 1–6.

[70] GUPTA, A. K., AND RAVAL, M. S. **A robust and secure watermarking scheme based on singular values replacement**. *Sadhana 37*, 4 (Aug 2012), 425–440.

[71] KAUSHIK, A. K. **A novel approach for digital watermarking of an image using DFT**. *Int JElectronComp Sci Eng 1*, 1 (2012), 35–41.

[72] KRIZHEVSKY, A., SUTSKEVER, I., AND HINTON, G. E. **Imagenet Classification with Deep Convolutional Neural Networks**. In *Advances in Neural Information Processing Systems 25* (2012), F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, Eds., Curran Associates, Inc., pp. 1097–1105.

[73] GRAVES, A., MOHAMED, A., AND HINTON, G. **Speech recognition with deep recurrent neural networks**. In *2013 IEEE International Conference on Acoustics, Speech and Signal Processing* (May 2013), pp. 6645–6649.

[74] JIANG, Y., ZHANG, Y., PEI, W., AND WANG, K. **Adaptive spread transform QIM watermarking algorithm based on improved perceptual models**. *AEU - International Journal of Electronics and Communications 67*, 8 (2013), 690 – 696.

[75] LIN, H.-F., LU, L.-W., GUN, C.-Y., AND CHEN, C.-Y. **A copyright protection scheme based on PDF**. *Int J Innov Comput Inf Control 9*, 1 (2013), 1–6.

[76] MAKBOL, N. M., AND KHOO, B. E. **Robust blind image watermarking scheme based on Redundant Discrete Wavelet Transform and Singular Value Decomposition**. *AEU - International Journal of Electronics and Communications 67*, 2 (2013), 102 – 112.

[77] SU, Q., NIU, Y., ZOU, H., AND LIU, X. **A blind dual color images watermarking based on singular value decomposition**. *Applied Mathematics and Computation 219*, 16 (2013), 8455 – 8466.

[78] WAN, W., LIU, J., SUN, J., YANG, X., NIE, X., AND WANG, F. **Logarithmic spread-transform dither modulation watermarking based on perceptual model**. In *Image Processing (ICIP), 2013 20th IEEE International Conference on* (2013), IEEE, pp. 4522–4526.

[79] DENG, L. **A tutorial survey of architectures, algorithms, and applications for deep learning**. *APSIPA Transactions on Signal and Information Processing 3* (2014).

[80] KARPATHY, A., TODERICI, G., SHETTY, S., LEUNG, T., SUKTHANKAR, R., AND FEI-FEI, L. **Large-scale video classification with convolutional neural networks**. In *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition* (2014), pp. 1725–1732.

[81] SINGH, A. K., DAVE, M., AND MOHAN, A. **Hybrid Technique for Robust and Imperceptible Image Watermarking in DWT–DCT–SVD Domain**. *National Academy Science Letters 37*, 4 (Aug 2014), 351–358.

[82] TAO, H., CHONGMIN, L., ZAIN, J. M., AND ABDALLA, A. N. **Robust image watermarking theories and techniques: A review**. *Journal of applied research and technology 12*, 1 (2014), 122–138.

[83] CAO, J. **Improved Spread Transform Dither Modulation: A New Modulation Technique for Secure Watermarking**. In *Digital-Forensics and Watermarking* (Cham, 2015), Y.-Q. Shi, H. J. Kim, F. Pérez-González, and C.-N. Yang, Eds., Springer International Publishing, pp. 399–409.

[84] HE, X., ZHU, T., AND YANG, G. **A geometrical attack resistant image watermarking algorithm based on histogram modification**. *Multidimensional Systems and Signal Processing 26*, 1 (Jan 2015), 291–306.

[85] LONG, J., SHELHAMER, E., AND DARRELL, T. **Fully convolutional networks for semantic segmentation**. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2015), pp. 3431–3440.

[86] NIELSEN, M. A. **Neural networks and deep learning**, vol. 25. Determination press San Francisco, CA, USA:, 2015.

[87] FAZLI, S., AND MOEINI, M. **A robust image watermarking method based on DWT, DCT, and SVD using a new technique for correction of main geometric attacks**. *Optik 127*, 2 (2016), 964 – 972.

[88] GATYS, L. A., ECKER, A. S., AND BETHGE, M. **Image style transfer using convolutional neural networks**. In *Proceedings of the IEEE conference on computer vision and pattern recognition* (2016), pp. 2414–2423.

[89] MEHTA, S., PRABHAKARAN, B., NALLUSAMY, R., AND NEWTON, D. **mPDF: Framework for Watermarking PDF Files using Image Watermarking Algorithms**. *arXiv preprint arXiv:1610.02443* (2016).

[90] MILLETARI, F., NAVAB, N., AND AHMADI, S. **V-net: Fully Convolutional Neural Networks for Volumetric Medical Image Segmentation**. In *2016 Fourth International Conference on 3D Vision (3DV)* (Oct 2016), pp. 565–571.

[91] RASTEGARI, M., ORDONEZ, V., REDMON, J., AND FARHADI, A. **Xnor-net: Imagenet Classification Using Binary Convolutional Neural Networks**. In *Computer Vision – ECCV 2016* (Cham, 2016), B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds., Springer International Publishing, pp. 525–542.

[92] WAN, W., LIU, J., SUN, J., AND GAO, D. **Improved logarithmic spread transform dither modulation using a robust perceptual model**. *Multimedia Tools and Applications 75*, 21 (2016), 13481–13502.

[93] BITAR, A. W., DARAZI, R., COUCHOT, J.-F., AND COUTURIER, R. **Blind digital watermarking in PDF documents using Spread Transform Dither Modulation**. *Multimedia Tools and Applications 76*, 1 (Jan 2017), 143–161.

[94] GOLDBERG, Y. **Neural network methods for natural language processing**. *Synthesis Lectures on Human Language Technologies 10*, 1 (2017), 1–309.

[95] SHIH, F. Y. **Digital watermarking and steganography: fundamentals and techniques**. CRC press, 2017.

[96] THAKKAR, F. N., AND SRIVASTAVA, V. K. **A blind medical image watermarking: DWT-SVD based robust and secure approach for telemedicine applications**. *Multimedia Tools and Applications 76*, 3 (Feb 2017), 3669–3697.

[97] ZHANG, K., ZUO, W., CHEN, Y., MENG, D., AND ZHANG, L. **Beyond a Gaussian Denoiser: Residual Learning of Deep CNN for Image Denoising**. *IEEE Transactions on Image Processing 26*, 7 (July 2017), 3142–3155.

[98] ZHANG, K., ZUO, W., GU, S., AND ZHANG, L. **Learning Deep CNN Denoiser Prior for Image Restoration**. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (July 2017).

[99] ZHAO, H., GALLO, O., FROSIO, I., AND KAUTZ, J. **Loss Functions for Image Restoration With Neural Networks**. *IEEE Transactions on Computational Imaging 3*, 1 (March 2017), 47–57.

[100] BORRA, S., THANKI, R., AND DEY, N. **Digital image watermarking: theoretical and computational advances**. CRC Press, 2018.

[101] COUTURIER, R., PERROT, G., AND SALOMON, M. **Image Denoising Using a Deep Encoder-Decoder Network with Skip Connections**. In *Neural Information Processing* (Cham, 2018), L. Cheng, A. C. S. Leung, and S. Ozawa, Eds., Springer International Publishing, pp. 554–565.

[102] HATOUM, M. W., DARAZI, R., AND COUCHOT, J. **Blind Image Watermarking using Normalized STDM robust against Fixed Gain Attack**. In *2018 IEEE International Multidisciplinary Conference on Engineering Technology (IMCET)* (Nov 2018), pp. 1–6.

[103] HATOUM., M. W., DARAZI., R., AND COUCHOT., J. **Blind PDF Document Watermarking Robust Against PCA and ICA Attacks**. In *Proceedings of the 15th International Joint Conference on e-Business and Telecommunications - Volume 2 SECRYPT: SECRYPT,* (2018), INSTICC, SciTePress, pp. 420–427.

[104] KURIBAYASHI, M., FUKUSHIMA, T., AND FUNABIKI, N. **Data Hiding for Text Document in PDF File**. In *Advances in Intelligent Information Hiding and Multimedia Signal Processing* (Cham, 2018), J.-S. Pan, P.-W. Tsai, J. Watada, and L. C. Jain, Eds., Springer International Publishing, pp. 390–398.

[105] ZHANG, K., ZUO, W., AND ZHANG, L. **Ffdnet: Toward a Fast and Flexible Solution for CNN-Based Image Denoising**. *IEEE Transactions on Image Processing 27*, 9 (Sep. 2018), 4608–4622.

[106] HATOUM, M. W., DARAZI, R., AND COUCHOT, J.-F. **Normalized blind STDM watermarking scheme for images and PDF documents robust against fixed gain attack**. *Multimedia Tools and Applications* (Nov 2019).

[107] NURSIAH, N., WONG, K., AND KURIBAYASHI, M. **Reversible data hiding in pdf document exploiting prefix zeros in glyph coordinates**. In *2019 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference (APSIPA ASC)* (2019), pp. 1298–1302.

List of Figures

List of Tables

Document generated with LaTeX and:
the LaTeX style for PhD Thesis created by S. Galland — http://www.multiagent.fr/ThesisStyle
the tex-upmethodology package suite — http://www.arakhne.org/tex-upmethodology/