Spectral Schemes for Rightful Ownership Protection

Mohammadreza Ghaderpanah

A Thesis

in

The Concordia Institute

for

Information Systems Engineering

Presented in Partial Fulfillment of the Requirements

for the Degree of Master of Applied Science (Information Systems Security) at

Concordia University

Montréal, Québec, Canada

August 2008

Your file Votre référence
ISBN: 978-0-494-42530-5
Our file Notre référence
ISBN: 978-0-494-42530-5

Abstract

Spectral Schemes for Rightful Ownership Protection

Mohammadreza Ghaderpanah

Since their introduction, photo sharing and social network websites such as Flickr, RockYou, MySpace, PhotoShelter, and FaceBook have attracted millions of users, many of whom are using these sites to share digital photos, but there is an increasing concern about controlling copyrighted content. Using digital image watermarking technology, online photo-sharing users will not only be able to monetize nonlinear distribution of copyrighted content, but also apply rules to how the content can be used and shared in order to avoid having the digital photos corraled and reused.

This thesis introduces a robust invisible image watermarking mechanism using nonnegative matrix factorization and singular value decomposition approaches. The proposed watermarking scheme improves the performance of the data embedding system effectively, and provides resistance to a wide range of attacks. Illustrating experimental results show the much improved performance of the proposed watermarking approach in comparison with existing techniques, and also demonstrate its robustness against a variety of intentional attacks and normal visual processes. Inspired by the successful application of graph theory in 2D image watermarking, and looking to address its limitations in often difficult 3D model identification and authentication, we also present in this thesis a 3D mesh fingerprinting technique using spectral graph theory. The main idea is to partition a 3D model into sub-meshes, then apply eigen-decomposition to the Laplace-Beltrami matrix of each sub-mesh, followed by hashing each transformed sub-mesh. The experimental results using a benchmark of 3D models demonstrate the effectiveness of the proposed fingerprinting technique.

Table of Contents

List of Figures

CHAPTER 1

Introduction

1.1 Framework and motivation

Watermarking can be defined as the process of embedding data called a watermark into a digital object without making changes to the quality of the host substantially. The digital object could be an image, video, or audio. The watermark is used as a signature to prove ownership, and can only be detected or extracted by the owner. The watermark carries either the information about the owner of the cover or the recipient. The owner of the special key is the only one who can extract the watermark. Moreover a good watermark should satisfy some watermarking requirements like invisibility and robustness against attacks [1–3].

1.2 Watermarking requirements

There are several properties that are necessary to design a watermarking system. These properties are related to the resistance to malicious attacks, difficulty to be noticed, and robustness against common distortions [1, 2]. Some of the following properties are discussed in more details in the following sections.

Imperceptibility or invisibility

This property is also referred to as fidelity or invisibility. The embedding algorithm must embed the watermark in such a way it does not affect or degrades the perceptual quality of the host media. The watermark embedding process is said to be imperceptible if the naked eye cannot distinguish between the original data and watermarked data.

Robustness

The watermarking scheme is said to be robust if it is strong enough to resist any kind of intentional (e.g. lossy compression, resampling, etc) or unintentional attacks (e.g. rescaling, rotation, etc). Furthermore, the watermark must be difficult for an unauthorized user to remove. In other words, the authorized user is the only person who can remove the watermark because s/he owns the secret key of the watermark embedding. The more robust a watermark, the harder it is to remove or alter the watermark.

Unambiguousness

The retrieval of the watermark should clearly be able to identify the owner.

Capacity

It is also referred to as the data payload, or the amount of information that can be embedded in a watermark without making perceptual distortion. This property is very important since it influences the watermark robustness.

1.3 General watermarking scheme

In general, all watermarking methods consist of two main components, the embedding system and the watermark extraction or recovery system. Figure 1.3 depicts the watermark embedding and extraction model with the presence of an attack.

Watermarking embedding system

The watermark embedding system has as inputs: the cover media, the key and the watermark symbol. The output of the embedding algorithm is the watermarked data. Figure 1.1 gives the model for watermark embedding scheme.

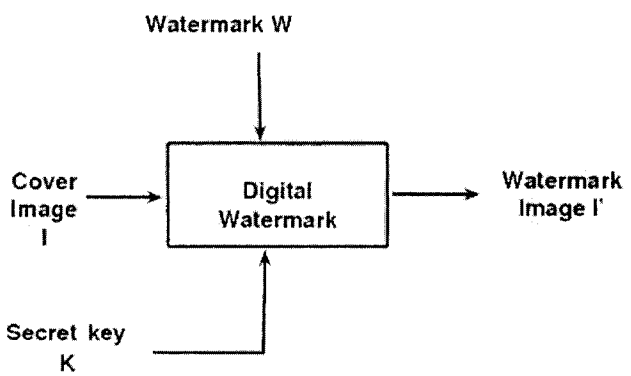


Figure 1.1: Watermark embedding model.

Watermarking extraction system

The watermark recovery process has as inputs: the watermarked data, the secret key, the original cover and/or the original watermark. Output of the extraction algorithm is either the suspect watermark or some kind of confidence measure. The watermark detector decides whether a watermark is present. If so, the extracted watermark is then compared with original watermark to measure their differences. Figure 1.2 gives the model for the watermark recovery scheme.

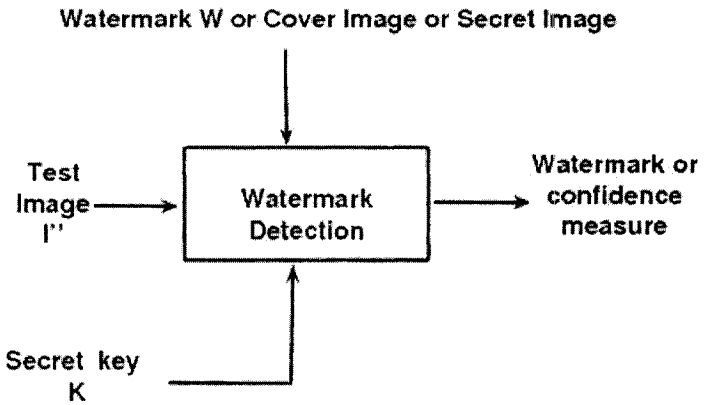


Figure 1.2: Watermark extraction model.

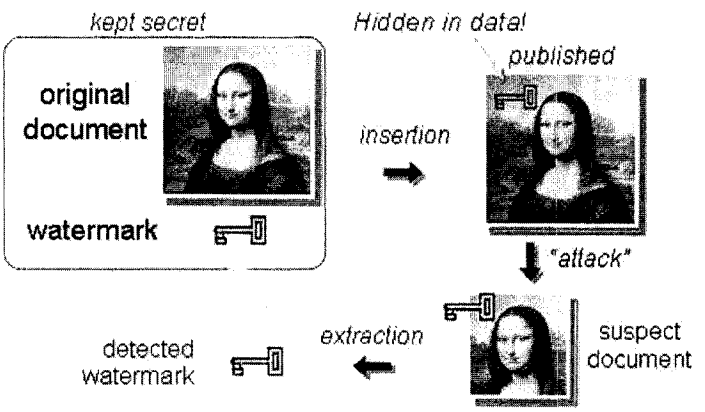


Figure 1.3: Watermark embedding and extraction model.

1.4 Types of watermarking systems

Watermarking system can be classified into different categories depending on different criterion [3,4].

Visible watermark

In visible watermarking, the watermark data is embedded in the host (cover) such that the watermark is intentionally perceptible to the human observer [5]. Primary purpose of this technique is to prevent unauthorized use of the media commercially. These watermarks commonly take the form of logo.

Invisible watermark

In the contrast of visible watermark, The invisible watermark is embedded into the host contents such that the watermark is not perceptible to the observer, but may be extracted/detected by a computer program. The invisible watermark [6, 7] is used to make assertion about the object ownership.

Private watermarking system

Also called non-blind or non-oblivious watermarking [8, 9]. Both the host and the watermarked media are needed for watermark detection. Only the authorized users are able to access the watermarks.

Public watermarking system

Also referred to as blind or oblivious watermarking. This type of watermarking technique requires only the secret key during the detection of the watermark information. It is used when access to

the cover is not feasible.

Semi-private watermarking system

Also called semi-blind or semi-oblivious watermarking. This type of watermarking technique does not use the original cover to extract the watermark. It only requires the secret keys and a copy of the embedded watermark data.

Watermarking in spacial domain

The watermark is encoded by modifying pixels directly [4], the spatial domain methods are the earliest and simplest watermarking techniques. They often require low computational cost but they have a low information hiding capacity, and also the watermark can be easily erased by lossy image compression.

Watermarking in transform domain

The transform domain approaches insert the watermark into the transform coefficients of the image cover, yielding more information embedding and more robustness against watermarking attacks. Many different ideas have been presented, most of them originating from Cox et al [8]. Barni et al [10] have improved the idea by providing a blind detection system.

1.5 Application of watermarking

Watermarking application can be found in different areas [1–3] and can be categorized as follows:

Watermarking for copy protection

Copy protection is an important watermarking application. It prevents unauthorized copying of the media. The watermarked information can directly control the digital devices for copy protection [11].

Fingerprinting for transaction tracing

The purpose of this application is to convey information about the legal recipient in order to identify single distributed copies of data. Fingerprinting application requires robustness against attacks. It enables the owner of the intellectual property to trace customers who have broken their license agreement.

Watermarking for copyright protection

A digital watermark contains the owner identification and the content identification. The watermark can be used for the protection of intellectual property and proof of ownership [12].

Watermarking for image authentication

A fragile watermark is very sensitive to any kind of modifications. In other words, if a portion of the image is altered, the watermark should detect the modified area. Fragile watermarks can be used to check the authenticity of the image.

Medical safety

The purpose of medical safety is to increase the confidentiality of medical information by embedding the date and the patient's name in medical images.

Broadcast monitoring

Watermarks can be embedded in any kind of data to be widely broadcasted on a network and help the automated identification of broadcasted programs. Commercials and Tv products are examples in which broadcast monitoring can be applied to ensure that the multimedia data are not illegally distributed.

1.6 Overview of digital image watermarking

Most watermarking research and publications are focused on 2D images [13–16]. The reason might be that there is a large demand for image watermarking products. The goal of digital image watermarking is to embed a watermark into an image such that it can be extracted later for proof of ownership. Digital watermarking technology provides law enforcement officials with a forensic tool and an effective means of tracing and catching pirates.

Different watermarking techniques based on the embedding domain have been proposed. The watermark can be embedded directly in the spatial domain or in some transform space using common transforms, such as discrete Fourier transform (DFT) [5, 17], discrete Cosine transform (DCT) [5, 13, 21], discrete Wavelet transform (DWT) [14, 15, 18–20], and fast Hadamard transform (FHT) [22–24]. In transform-based schemes the image is transformed prior to watermark embedding and the watermark is hidden in the transformed coefficients representing the image. The watermarked image is obtained using an inverse transformation.

Transform domains have been extensively studied in image compression and many research results can be applied to digital watermarking. For example, when a typical image is mapped into the frequency domain, the energy is concentrated in low-index terms which are very large comparing

to the high-index terms. A digital image is demonstrated by the low frequency components. Those low frequency components represent the over all shape of the image, outline of features in the image, and the luminance and the contrast characteristics. High frequencies represent sharp edges. For example 95% of the energy found in the lowest 5% of the special frequencies of the two dimensional DCT domain.

The watermark should not be placed perceptually in insignificant region of the image or its spectrum since many common signal and geometric processes affect these components. For example lossy compression is an operation that usually eliminates perceptually non-salient components of an image. If we wish to protect the watermark algorithm from such operation we must place the watermark in significant region of the cover. In fact, the loss usually occurs in high frequency components. Therefore, the best solution for such attack is to place the watermark in low frequency components.

1.7 Spread spectrum watermarking

I.J.Cox et al. [8,25] propose an invisible robust watermarking technique. They insert the watermark into the spectral components of the image using DCT domain. The general idea of spread spectrum watermarking system is spread a narrow-band signal as a watermark over a much wider important frequency bands which are obtained from the transformed cover image. The watermark in each band is small and undetectable. On the other hand, the receiver with knowledge of spreading function should be able to extract and sum up the watermark.

In [8] the watermark is embedded in the first n lowest frequency components or the first highest magnitude components $V = \{v_i\}_1^n$ of the full image DCT in order to provide high level of robustness

to JPEG compression. The watermark consists of a sequence of real numbers $W = \{w_i\}_1^n$ is computed where each w_i is chosen according to $N(0,1)$ where $N(0,1)$ denotes a normal distribution with mean 0 and variance 1. The watermark is embedded into an image using formula $\acute{v}_i = v_i(1 + \alpha w_i)$ where α is the watermark strength factor=0.1. Watermark detection is performed using the following similarity measure:

$$sim(W, \acute{W}) = \frac{W, \acute{W}}{\sqrt{\acute{W}, \acute{W}}}$$

The $\acute{W}$ is the extracted watermark, which is calculated as:

$$\{\acute{w}_i\}_1^n = \{(\frac{\acute{v_i}}{v_i} - 1) \diagup \alpha\}_1^n$$

Where $\acute{v_i}$ components are extracted from the received watermarked image, and v_i component extracted from the original cover image. The watermark is present if the extracted $sim(W, \acute{W})$ is greater than threshold.

Cox spread the watermark across 1000 lowest frequency. Robustness tests showed that the watermark is robust to common attacks. Retrieval of the watermark unambiguously identifies the owner and the watermark. The watermarking technique has the disadvantage that it needs the original image for its extraction. It is also not clear whether the watermark is robust to photocopying. Figure 1.4 give the process of the insertion and extraction process.

1.8 Singular value decomposition techniques

The Singular Value Decomposition (SVD) is a widely used technique to decompose a matrix into several component matrices, exposing many of the useful and interesting properties of the original matrix. (SVD) is developed for a variety of applications. The main properties of (SVD) from

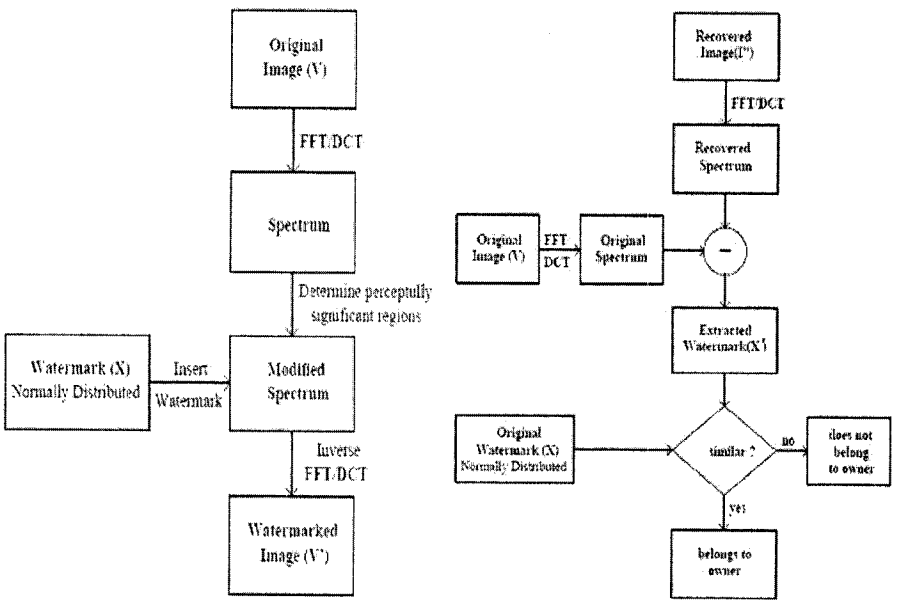


Figure 1.4: Spread spectrum watermarking embedding and extraction model [8].

the viewpoint of image processing applications are: The singular values (SVs) of an image have very good stability, i.e., when a small perturbation is added to an image its (SVs) do not change significantly; (SVs) represent intrinsic algebraic image properties.

The interest in the (SVD), from the point of view of watermarking is its ability to decompose the cover and the watermark images in to three matrices. Let A be an arbitrary real $m \times n$ matrix. There are two orthogonal matrices U and V, $U^T U = I$, $V^T V = I$ and a diagonal matrix Σ; such that:

$$A = U \Sigma V^T$$

In this case, U is $m \times m$ and V is $n \times n$, so that Σ is rectangular with the same dimensions as A. The diagonal entries of Σ, called the singular values (SVs) of A. The columns of U and V are

called left and right singular vectors for A. Each SV specifies the luminance of the image layer and the corresponding pair of singular vectors specify the geometry of the image.

New invertible digital image watermarking method based on singular value decomposition was proposed [9]. This method performs well both in resolving rightful ownership and in resisting common attacks. The watermarking embedding and extraction algorithms can be summarized as follows: in watermark embedding process, the singular value decomposition of an $N \times N$ cover image A is computed to obtain two orthogonal matrices U and V and one diagonal matrix S, other non-square images can be processed in exactly the same way.

$$A \Rightarrow USV^T$$

The watermark W is added to the matrix S, followed by singular value decomposition to the new matrix

$$S + \alpha W \Rightarrow U_w S_w V_w^T$$

Where the positive constant α is the scale factor which controls the strength of the watermark to be inserted. The watermarked image A_w is obtained by:

$$A_w \Rightarrow US_wV^T$$

In watermark detection algorithm, they simply reverse the above steps given U_w, S, and V_w matrices which are saved in the secret key during embedding process of the watermarked image and possibly distorted image $A_w^\star$.

$$A_w^\star \Rightarrow U^\star S_w^\star V^{\star T}$$

$$D^\star \Rightarrow U_w S_w^\star V_w^T$$

$$W^\star \Rightarrow 1/\alpha(D^\star - S)$$

To study the robustness of (SVD) watermarking method they compared the results with the Spread Spectrum Communication method proposed by Cox [8]. The results show that the (SVD) [9] method is much more robust by testing it against six different attacks: adding noise, low pass filtering, JPEG compression, scaling, image cropping and rotation.

1.9 Wavelet techniques

Wavelet technique prevents watermark removal by JPEG-2000 lossy compression. One dimensional DWT converts an input sequence into a low pass sub-band and high pass sub-band. A two dimensional DWT is constructed from single level decomposition first to the columns and then to the rows to give four sub-bands as shown in Figure 1.5 and Figure 1.6. In the first level decomposition, the lowest frequency band is found in the top-left corner LL. At the same resolution level, the block HL contains information about the highest horizontal and lowest vertical frequency band. Similarly, the block LH contains information about the lowest horizontal and the highest vertical frequency band, and block HH contains information about the highest horizontal and the highest vertical frequency band. The same process is repeated for higher levels.

Recently many watermarking techniques use wavelet transform in watermarking. Some of the schemes that were reviewed will be discussed briefly. In [19], the authors used the idea that embedding the watermark in the low frequency area increase the robustness with respect with image distortion that have low pass characteristics like filtering, lossy compression, geometrical distortions. On the other hand, oblivious schemes with low-frequency watermarks are more sensitive to modifications of the histogram, such as contrast, brightness adjustment, gamma correction, histogram equalization, and cropping. Watermarks inserted into middle and high frequencies are typically

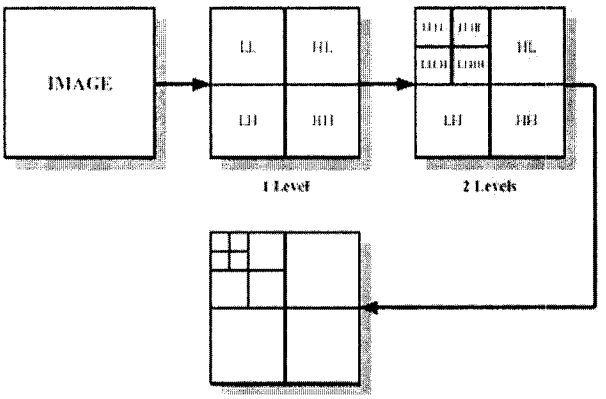


Figure 1.5: 3-Level of 2D Discrete Wavelet Decomposition model.

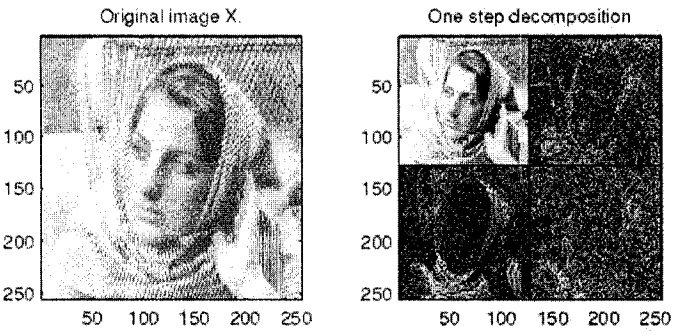


Figure 1.6: One-level of 2D Discrete Wavelet Decomposition for an image.

less robust to low-pass filtering lossy compression and small geometric deformations of the image, but are extremely robust with respect to noise adding. It is understandable that the advantages and disadvantages of low and middle-to-high frequency watermarks are complementary. It appears that by embedding two watermarks into one image could achieve extremely high robustness properties with respect to image processing operations. The above reasoning leads to propose many techniques to embedding multiple watermarks into the low frequency and high frequency bands of Discrete Wavelet Transform.

In [19], two level decomposition is applied to the cover image, followed by embedding the watermarks into the second level LL and HH band respectively. The watermarked image is obtained using the following relationship. $\hat{V}_{ij} = V_{ij} + \beta W(i,j)$, where $\hat{V}_{ij} = (i,j)th$ watermark embedded DWT coefficient, $V_{ij} = (i,j)th$ DWT coefficient of value V, and β is a scaling factor which determines the strength of the watermark.

For the watermark extraction algorithm, two level of DWT decomposition is applied to the suspected and the original watermarked images to recover the LL and HH bands. Subtraction of the suspected and original bands is performed to recover the watermark bits in both LL and HH bands. The output is then divided by the watermark strength factor β. The operation can be summarized as $\hat{W}(i,j) = (\hat{V}(i,j) - V(i,j))/\beta$.

Another watermarking scheme proposed in [15] used SVD and DWT to embed the SVs of the watermark image in all frequencies of the discrete wavelet transformed cover image. This method consists of decomposing the cover image into four transformed sub-bands (LL, LH, HL, and HH), then the SVD is applied to each band, followed by modifying the singular values of the transformed sub-bands with the singular values of the visual watermark. This modification in all frequencies provides more robustness to different attacks. It is important to note that the wavelet coefficients with the highest magnitude are found in the LL sub-band, and those with the lowest coefficients are found in the HH sub-band. Correspondingly, the singular values with the highest magnitudes are in the LL sub-band, and the singular values with the lowest magnitudes are in the HH sub-band, Therefore, two scaling factor are used. The first scaling factor is used for the LL sub-band and the second scaling factor is used for all other sub-bands such that, the first scaling factor is greater than the second one. Experimental results show that the watermarks inserted in the lowest frequencies

(LL sub-band) are resistant to one group of attacks, and the watermarks embedded in highest frequencies (HH sub-band) are resistant to another group of attacks. If the same watermark is embedded in 4 blocks, it would be extremely difficult to remove or destroy the watermark from all frequencies. In some cases, embedding in the HL and LH sub-bands is also resistant to certain attacks. Two examples of those attacks are histogram equalization and gamma correction.

1.10 Thesis overview and contributions

The organization of this thesis is as follows:

- The first Chapter contains a brief review of essential concepts and definitions which we will refer to throughout the thesis, and presents a short summary of material relevant to watermarking systems, digital image watermarking and embedding and extraction models.

- In Chapter 2, we introduce a robust invisible image watermarking mechanism using nonnegative matrix factorization and singular value decomposition approaches [36]. The proposed watermarking scheme improves the performance of the data embedding system effectively, and provides resistance to a wide range of attacks. Illustrating experimental results show the effectiveness and the much improved performance of the proposed watermarking approach in comparison with existing techniques, and also demonstrate its robustness against a variety of intentional attacks and normal visual processes [37, 38].

- In Chapter 3, we present a hashing technique for 3D models using spectral graph theory and entropic spanning trees [39]. The main idea is to partition a 3D triangle mesh into an ensemble of sub-meshes, then apply eigen-decomposition to the Laplace-Beltrami matrix of

each sub-mesh, followed by computing the hash value of each sub-mesh. This hash value is defined in terms of spectral coefficients and Tsallis entropy estimate. The experimental results on a variety of 3D models demonstrate the effectiveness of the proposed technique in terms of robustness against the most common attacks including Gaussian noise, mesh smoothing, mesh compression, scaling, rotation as well as combinations of these attacks.

- In the **Conclusions** Chapter, we summarize the contributions of this thesis, and we propose several future research directions that are directly or indirectly related to the work performed in this thesis.

CHAPTER 2

Spectral Image Watermarking Scheme

We introduce a robust invisible image watermarking mechanism using nonnegative matrix factorization and singular value decomposition approaches. The proposed watermarking scheme improves the performance of the data embedding system effectively, and provides resistance to a wide range of attacks. Illustrating experimental results show the effectiveness and the much improved performance of the proposed watermarking approach in comparison with existing techniques, and also demonstrate its robustness against a variety of intentional attacks and normal visual processes.

2.1 Introduction

The rapid development of electronic commerce and the increased demand for online services has triggered the need for multimedia protection. Although digital data has many advantages over analog data, the unrestricted duplication of copyrighted materials and illegal recordings of digital content caused that most of service providers be reluctant to offer services in digital format. Data embedding in digital multimedia could help in providing proof of origin and distribution of content [26]. Watermarking is a process of adding a structure called a watermark to the original data object [1, 26–28]. Such a watermark can be used to prevent online piracy and illegal recordings of

digital content, and to enforce copyright compliance in the use of digital media [26].

A basic digital image watermarking scheme consists of a cover image, a watermark structure, an embedding algorithm, and an extraction algorithm. A variety of watermarking techniques have been proposed for multimedia protection, and in particular for digital images [4,8,29]. These techniques can be divided into two main categories according to the embedding domain of the cover image: spatial domain methods and transform domain methods [4]. The spatial domain methods are the earliest and simplest watermarking techniques but have a low information hiding capacity, and also the watermark can be easily erased by lossy image compression. On the other hand, the transform domain approaches insert the watermark into the transform coefficients of the image cover, yielding more information embedding and more robustness against watermarking attacks.

Recently, a singular value decomposition (SVD)-based watermarking technique and its variants have been proposed [9,16,21]. The main idea of these approaches is to find the SVD of a cover image or the SVD of each block of the cover image, and then modify the singular values to embed the watermark. In [21], a hybrid non-blind watermarking scheme based on the discrete wavelet transform (DWT), the discrete cosine transform (DCT) and the SVD was proposed. This method consists of decomposing the cover image into four transformed sub-bands, then the SVD is applied to each sub-band, followed by modifying the singular values of the transformed sub-bands with the singular values of the visual watermark. This modification in all frequencies provides more robustness against different attacks [21]. Also, a watermarking technique based on the independent component analysis (ICA) method [30] was recently introduced in [31,32], and it uses ICA to project the image into a basis with its components as statistically independent as possible. The watermark is then embedded into the basis. The ICA-based scheme, however, tends to create block artifacts.

Another SVD-block based watermarking scheme was proposed in [16], and it embeds the watermark in two layers. In the first layer, the cover image is divided into small blocks and the singular values of the watermark are embedded in those blocks. In the second layer, the cover image is used as a single block to embed the whole watermark. One major weakness of SVD is that it produces low rank bases which do not respect the nonnegativity of the cover image. Nonnegative matrix factorization (NMF) was introduced in [33, 34] to overcome this limitation without significantly increasing the error of the associated approximation, and it has been shown to be an effective tool in many engineering applications including data mining, and spectroscopy [35].

In this chapter we introduce a robust method for digital watermarking and secure copyright protection of digital images. The proposed watermarking scheme is based on NMF and SVD approaches, and it improves the performance of the data embedding system effectively. It is also resistant to a variety of intentional attacks and normal visual processes. A preliminary work on this watermarking technique was presented in [36].

The remainder of this chapter is organized as follows. In Section 2, we briefly review some related work that is closely related to our proposed scheme. In Section 3, we introduce the proposed approach and we describe in more details the fundamental steps of the watermark embedding and extraction algorithms. In Section 4, we present some experimental results to demonstrate the much improved performance of the proposed method in comparison with existing techniques, and also to show its robustness against the most common attacks. Finally, we conclude in Section 5.

2.2 Related Work

In this section, we will review four representative methods for digital image watermarking that are closely related to our proposed scheme. We briefly show their mathematical foundations and algorithmic methodologies.

2.2.1 SVD watermarking scheme

The SVD of a cover image C of size $m \times m$ is given by $C = U\Sigma V'$, where U is an orthogonal matrix ($U'U = I$), $\Sigma = \text{diag}(\lambda_i)$ is a diagonal matrix of singular values λ_i, $i = 1, \ldots, m$, arranged in decreasing order, and V is an orthogonal matrix ($V'V = I$) as depicted in Fig. 2.1. The columns of U are the left singular vectors, whereas the columns of V are the right singular vectors of the cover image. The matrix V' denotes the transpose of V.

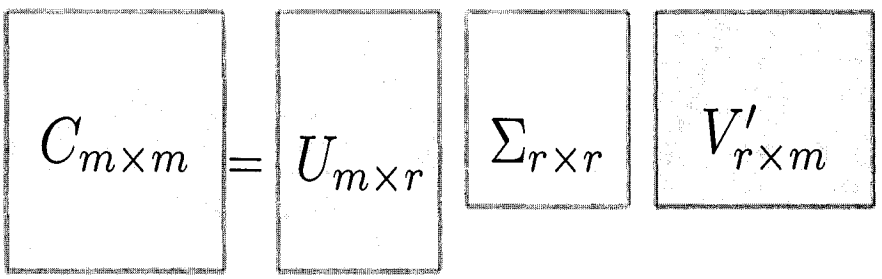


Figure 2.1: Illustration of the SVD approximation.

Let W be a visual watermark of size $w \times w$ with $w \leq m$. The SVD of the watermark is $W = U_w \Sigma_w V'_w$, where Σ_w is a diagonal matrix of the singular values λ_{wi} of the visual watermark. The SVD watermark embedding algorithm is based on the following linear transformation

$$\lambda_i^d = \lambda_i + \alpha \lambda_{wi}, \quad i = 1, \ldots, m \tag{1}$$

where λ_i^d denotes the distorted SVs of the watermarked image, and α is a constant scaling factor. Hence the watermarked image is given by $M = U\Sigma^d V'$, where $\Sigma^d = \text{diag}(\lambda_i^d)$. The linear transformation given by Eq. (1) is invertible, and hence we may extract the singular values of the visual watermark as follows

$$\hat{\lambda}_{wi} = \frac{\lambda_i^d - \lambda_i}{\alpha}$$

Consequently, the extracted watermark $\widehat{W}$ is given by $\widehat{W} = U_w \widehat{\Sigma}_w V_w'$, where $\widehat{\Sigma}_w = \text{diag}(\hat{\lambda}_{wi})$

2.2.2 Block-based SVD watermarking scheme

The cover image C is divided into blocks of size $\ell \times \ell$, and the SVD of each block L is given by $L = U_\ell \Sigma_\ell V_\ell'$. The singular values of the visual watermark W are embedded into each block of the cover image by modifying the largest singular value of each block. The scaling factor α_i used for each block embedding is chosen relative to the SVs of the watermark image and the largest singular value of the block and it is multiplied with a constant percentage c. The block-based SVD watermark embedding algorithm is based on the following linear transformation

$$\lambda_i^d = \lambda_{\max} + \alpha_i \lambda_{wi}, \quad i = 1, 2, ..., \tag{2}$$

where $\lambda_{\max}$ denotes the largest SV of a block in the cover image, λ_{wi} denotes the SVs of the visual watermark, and λ_i^d denotes the distorted SVs of a given block of the watermarked image. Hence, we may extract the singular values of the visual watermark as follows

$$\hat{\lambda}_{wi} = \frac{\lambda_i^d - \lambda_{\max}}{\alpha_i}$$

Therefore, the extracted watermark $\widehat{W}$ is given by $\widehat{W} = U_w \widehat{\Sigma}_w V_w'$, where $\widehat{\Sigma}_w = \text{diag}(\hat{\lambda}_{wi})$

2.2.3 DWT-SVD watermarking scheme

The cover image C of size $m \times m$ is decomposed into four subbands, the approximation coefficient LL, and the detailed coefficients HL, LH, HH. We then apply SVD to each subband of the cover image $C^k = U_c^k \Sigma_c^k V_c'^k$, $k = 1, 2, 3, 4$, where k denotes the bands, and λ_i^k, $i = 1, \ldots, m$, are the singular values of Σ_c^k. The SVD of the watermark W of size $\frac{m}{2} \times \frac{m}{2}$ is $W = U_w \Sigma_w V_w'$, where Σ_w is a diagonal matrix of the singular values λ_{wi} of the visual watermark. The singular values of the cover image in each subband are modified with the singular values of the visual watermark. For the LL subband

$$\lambda_i^{dk} = \lambda_i^k + \alpha_L \lambda_{wi}, \quad k = 1 \tag{3}$$

and for other subbands

$$\lambda_i^{dk} = \lambda_i^k + \alpha_H \lambda_{wi}, \quad k = 2, 3, 4 \tag{4}$$

where λ_i^{dk} denotes the distorted SVs of a subband of the watermarked image, α_L is a constant scaling factor to control the perception of watermark in LL subband, and α_H is the one for HL, LH, HH subbands. We then obtain four sets of modified DWT coefficients $M^k = U_c^k \Sigma_d^k V_c'^k$, where $\Sigma_d^k = \text{diag}(\lambda_i^{dk})$, $k = 1, 2, 3, 4$. Finally, by applying the inverse DWT using the four sets of modified DWT coefficients, we can produce the watermarked image.

The algorithm is invertible and the watermark can be extracted from the watermarked (and possibly attacked) image. It can be done by decomposing the watermarked image into four subbands, then applying SVD to each subband. The singular values are computed from the LL subband as

$$\hat{\lambda}_{wi} = (\lambda_i^{dk} - \lambda_i^k)/\alpha_L, \quad k = 1$$

and from the HL, LH, and HH subbands as

$$\hat{\lambda}_{wi} = (\lambda_i^{dk} - \lambda_i^k)/\alpha_H, \quad k = 2, 3, 4$$

Consequently, the four extracted watermarks are given by $\widehat{W}^k = U_w \widehat{\Sigma}_w^k V_w'$, where $\widehat{\Sigma}_w^k = \text{diag}(\hat{\lambda}_{wi})$, $k = 1, 2, 3, 4$.

2.2.4 ICA watermarking scheme

Given a cover image C of size $m \times m$ and a visual watermark W of size $w \times w$ with $w \leq m$, we first compute the components of the cover image x_i^C and the watermark x_i^W by dividing them into $k \times k$ blocks, $i = 1, \ldots, m^2/k^2$. The independent components of the cover image $y_i^C = P^C x_i^C$ and the watermark $y_i^W = P^W x_i^W$ are computed, using ICA projections P^C and P^W. The independent component of the cover image which has the lowest energy is updated by the one of the watermark

$$y_i^D = y_i^C + \alpha y_i^W \tag{5}$$

where y_i^D denotes the distorted independent components of the watermarked image, and α is a scaling factor to control the perception of the watermark. Finally the watermarked image M restored from the components $x_i^D = P_C^{-1} y_i^D$. The mixing matrices P^C and P^W are stored as the keys.

To extract the watermark from the watermarked image, we first compute the components x_i^D of the image by dividing it into $k \times k$ blocks. We then compute the independent components of the watermarked image $y_i^D = P^C x_i^D$, using the P^C mixing matrix. The independent components of watermark are computed as

$$\hat{y}_i^W = (y_i^D - y_i^C)/\alpha$$

The extracted watermark $\widehat{W}$ is then restored from the components $\hat{x}_i^W = P_W^{-1} \hat{y}_i^W$.

2.3 Proposed Method

2.3.1 Motivation

The purpose of SVD is to approximate an image by using fewer entries, and the use of the rank r ($r < m$) of the cover image C leads to the removal of the redundant information. The SVD of the cover image C may be approximated by a sum of rank-one matrices

$$C = \sum_{i=1}^{r} \lambda_i \boldsymbol{u}_i \boldsymbol{v}_i',$$

where $\boldsymbol{u}_i$ and $\boldsymbol{v}_i$ are the the left and right singular vectors respectively.

Reconstruction techniques must balance the logical attractiveness of measurement functions against their physical feasibility. In imaging and spectrometry for instance, all measurements at the optical-electronic are strictly positive photon counts. Unfortunately, the SVD decomposition does not generate strictly positive basis functions. With a view to avoiding this problem, we decompose the image into the product of two nonnegative matrices to generate a nonnegative measurement basis. This approach is described next.

2.3.2 Nonnegative matrix factorization

One major drawback of SVD is that the basis vectors may have both positive and negative components, and the data are represented as linear combinations of these vectors with positive and negative coefficients. In many applications, the negative components contradict physical realities. To address this problem, the NMF approach was proposed to search for a representative basis with only nonnegative vectors [33,34]. The NMF approach can be formulated as follows. Given a cover image C of size $m \times m$, we can approximately factorize C into the product of two nonnegative

matrices B and H with sizes $m \times r$ and $r \times m$ respectively, that is $C = BH$, where $r \leq m$ (see Fig. 2.2). The nonnegative matrix B contains the NMF basis vectors, and the nonnegative weight matrix H contains the associated coefficients (nonnegative weights). To measure the quality of the approximation factorization $C = BH$, a cost function between C and BH needs to be optimized subject to non-negativity constraints on B and H. This is done by minimizing the $\mathcal{I}$-information divergence given by

$$\mathcal{I}(C\|BH) = \sum_{ij} \left(C_{ij} \log \frac{C_{ij}}{(BH)_{ij}} - C_{ij} + (BH)_{ij} \right),$$

which yields the following multiplicative update rules [34]

$$H_{kj} \leftarrow H_{kj} \frac{\sum_i B_{ik} C_{ij}/(BH)_{ij}}{\sum_i B_{ik}},$$

$$B_{ik} \leftarrow B_{ik} \frac{\sum_j H_{kj} C_{ij}/(BH)_{ij}}{\sum_j H_{kj}},$$

where the matrices B and H are initialized as nonnegative random matrices, and the updates are done alternatively, that is after updating one row of H, we need to update the corresponding column of B. In other words, we should not update the whole matrix H first followed by an update of the matrix B. The NMF algorithm is therefore an iterative optimization algorithm, which modifies at each iteration the nonnegative basis functions (i.e. columns of B) and encodings (i.e. H_{kj}) until convergence.

It is worth pointing out that with respect to image quality, digital watermarking is closely related to image compression. Fig. 2.3 depicts an example of image compression via nonnegative matrix factorization.

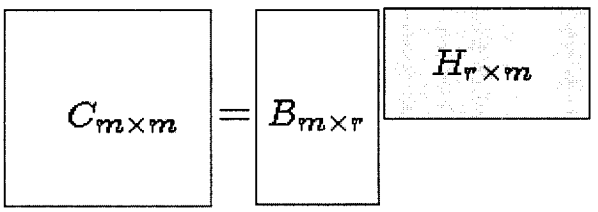


Figure 2.2: Illustration of the NMF approximation.

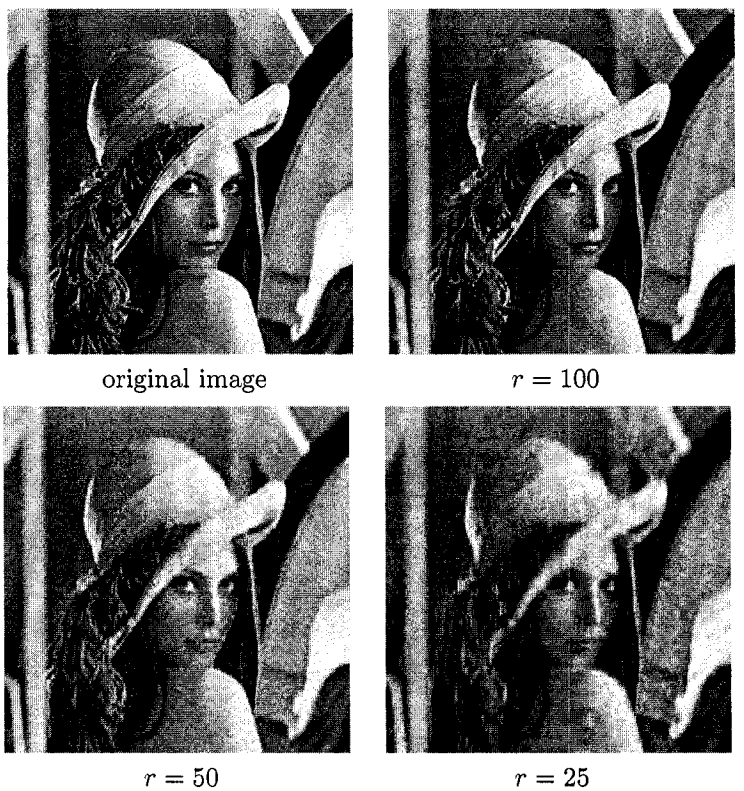


Figure 2.3: Image compression using NMF with different numbers of basis functions.

2.3.3 Proposed NMF watermarking scheme

The proposed watermarking scheme is based on NMF and SVD factorization approaches. The key idea of this proposed scheme is to apply NMF to the watermark image, and also to each block of

the cover image. The embedding and extraction algorithms are illustrated in the block diagrams of Fig. 2.4 and Fig. 2.5.

Watermark embedding algorithm:

1) Divide the cover image into blocks of size $\ell \times \ell$.

2) To each block L, apply NMF: $L = B_\ell H_\ell$, followed by an SVD to the weight matrix: $H_\ell = U_\ell \Sigma_\ell V_\ell'$.

3) Apply NMF to the watermark: $W = B_w H_w$, followed by an SVD to the weight matrix: $H_w = U_w \Sigma_w V_w'$, where $\Sigma_w = \text{diag}(\lambda_{wi})$.

4) Modify $\lambda_{\max}$ according to $\lambda_i^d = \lambda_{\max} + \alpha \lambda_{wi}$, where $\lambda_{\max}$ denotes the largest SV of H_ℓ, α is a scaling factor, and λ_i^d denotes the distorted SV of H_ℓ.

5) Use all the distorted blocks $L^d = B_\ell H_\ell^d$, where $H_\ell^d = U_\ell \Sigma_\ell^d V_\ell'$ and $\Sigma_\ell^d = \text{diag}(\lambda_i^d)$, to produce the watermarked image.

Watermark extraction algorithm:

1) Divide the watermarked image into blocks

2) To each block K of the watermarked image, apply NMF: $K = B_k H_k$, followed by an SVD to the weight matrix: $H_k = U_k \Sigma_k V_k'$.

3) Extract the singular values from each block using $\hat{\lambda}_{wi} = (\lambda_i^d - \lambda_{\max})/\alpha$, where λ_i^d are the SVs of H_k, and $\lambda_{\max}$ is the largest SV of H_ℓ which is saved in each block of the secret key (cover image).

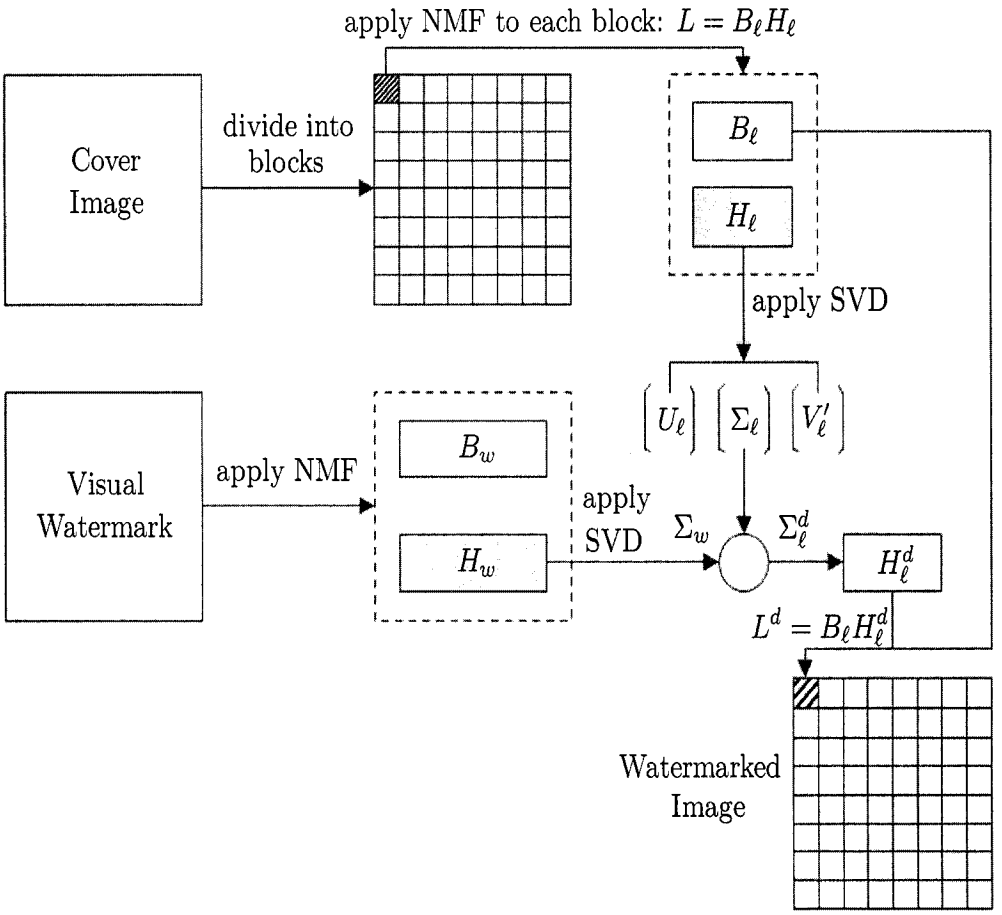


Figure 2.4: Watermark embedding algorithm.

4) Construct the watermark image $\widehat{W} = B_w \widehat{H}_w$, where $\widehat{H}_w = U_w \widehat{\Sigma}_w V'_w$. Note that U_w and V_w are saved in the secret key during the embedding stage, and $\widehat{\Sigma}_w = \text{diag}(\hat{\lambda}_{wi})$ is obtained from step **3)**.

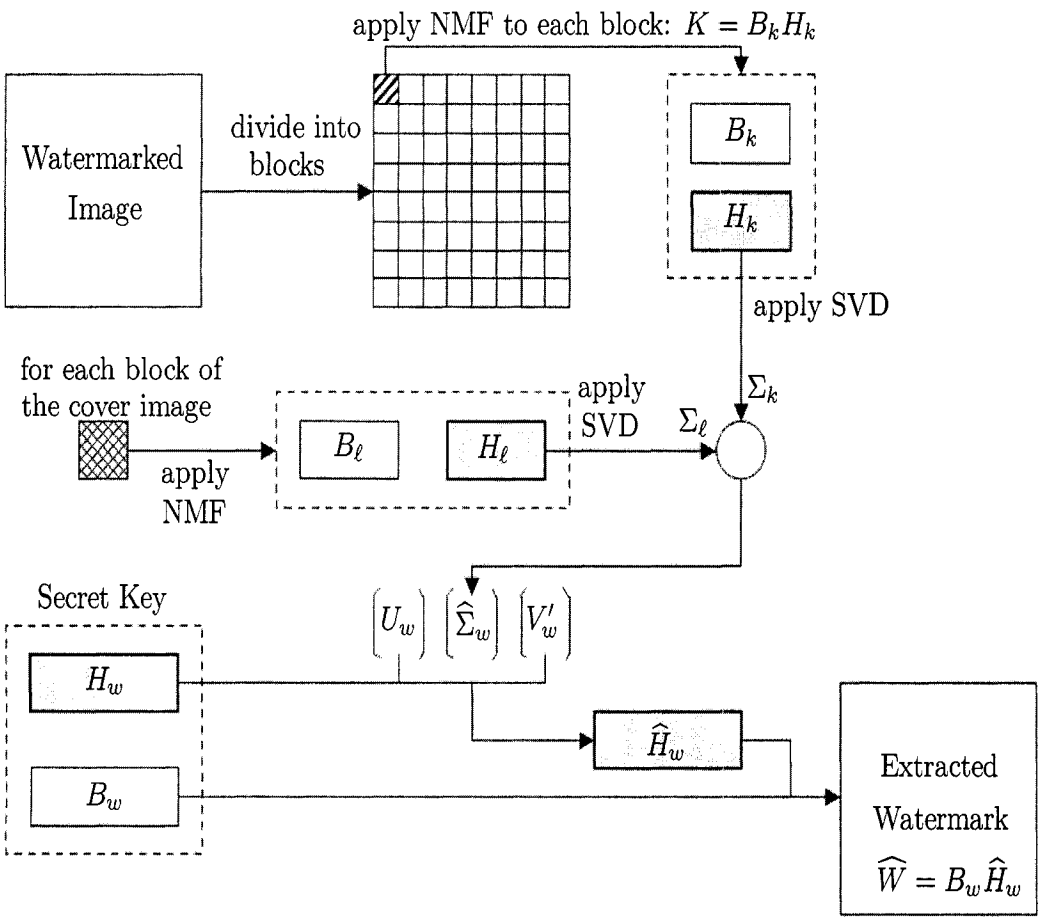


Figure 2.5: Watermark extraction algorithm.

2.4 Experimental Results

To test the performance and to demonstrate the effectiveness of the proposed watermarking scheme in comparison with the related watermarking techniques discussed in Section 2, we performed a number of experiments under different attacks using the Stirmark 3.1 benchmark [40] to attack the watermarked image. Such attacks include JPEG compression, Gaussian noise, impulsive noise,

multiplicative noise, median filtering, and sharpening. Fig. 2.6 shows a 256×256 cover image 'Lena', a 256×256 visual watermark 'Cameraman', and five extracted watermarks using the SVD scheme, the block-based SVD scheme, the hybrid DWT-SVD scheme, the ICA scheme, and the proposed watermarking method. The block size was set to 8×8, and the scaling factor α (also referred to as watermark strength or watermark embedding intensity) was chosen to satisfy two conditions: (i) the invisibility property, and (ii) the highest correlation coefficient between the visual watermark and the extracted watermark before applying any attacks. For fair comparison of the proposed scheme with the related approaches, we used the same watermark embedding strength $\alpha = 0.08$. Also, we set the constant percentage c to 0.1 for the block-based SVD scheme. The correlation coefficient ρ between two images $A = (A_{ij})$ and $B = (B_{ij})$ of size $m \times n$ is defined as

$$\rho = \frac{\sum_{i=1}^{m} \sum_{j=1}^{n} (A_{ij} - \bar{A})(B_{ij} - \bar{B})}{\sqrt{\left(\sum_{i=1}^{m} \sum_{j=1}^{n} (A_{ij} - \bar{A})^2\right)\left(\sum_{i=1}^{m} \sum_{j=1}^{n} (B_{ij} - \bar{B})^2\right)}}$$

where $\bar{A}$ and $\bar{B}$ are the mean values of A and B respectively.

Fig. 2.6(c)-(g) show the extracted watermarks before applying any attack to the watermarked image. Note that in this experiment the correlation coefficients between the cover image and the watermarked image are all equal to 0.9998, and the correlation coefficients between the visual watermark and the extracted one are all close to 1.0.

We tested our proposed scheme against a wide range of attacks including JPEG compression, Gaussian noise, impulsive noise, multiplicative noise, median filtering, and sharpening. Fig. 2.7 presents the watermarked images affected with these attacks. The corresponding extracted watermarks for the proposed scheme, block-based SVD, SVD-based, DWT-SVD, and ICA techniques are shown in Fig. 2.8 through Fig. 2.12 respectively. The visual comparison between the extracted watermarks of these five watermarking techniques clearly shows that the proposed scheme gives the

best results against all the listed attacks. In particular, Fig. 2.8 shows that the proposed scheme is more resistant to lossy JPEG compression, whereas the ICA-based method as depicted in Fig. 2.12 tends to produce block artifacts.

The correlation coefficient results are depicted in Fig. 2.13, where the performance of our proposed approach over other schemes is clearly demonstrated. We also tested our proposed algorithm using different block sizes as depicted in Fig. 2.14. We achieve more robustness against most of the attacks by using a block size 32×32.

Fig. 2.15 to Fig. 2.23 show the experimental results for a 256×256 cover image 'Clown', and a 256×256 visual watermark 'Sand-Clock'.

2.5 Conclusions

In this chapter we presented a novel watermarking technique for embedding visual watermarks into digital images using a combination of NMF and SVD approaches. The proposed scheme improves the performance of the data embedding system effectively, and it is resistant to a variety of intentional attacks and normal visual processes. Our experimental evaluations clearly show that the proposed watermarking technique outperforms the current SVD-based and ICA watermarking schemes, and also it provides a good balance between robustness and invisibility of the watermark. Our future goal is to extend the proposed watermarking scheme to 3D mesh models by partitioning a 3D mesh into smaller sub-meshes, and then applying NMF to the the probability transition matrix of each sub-mesh.

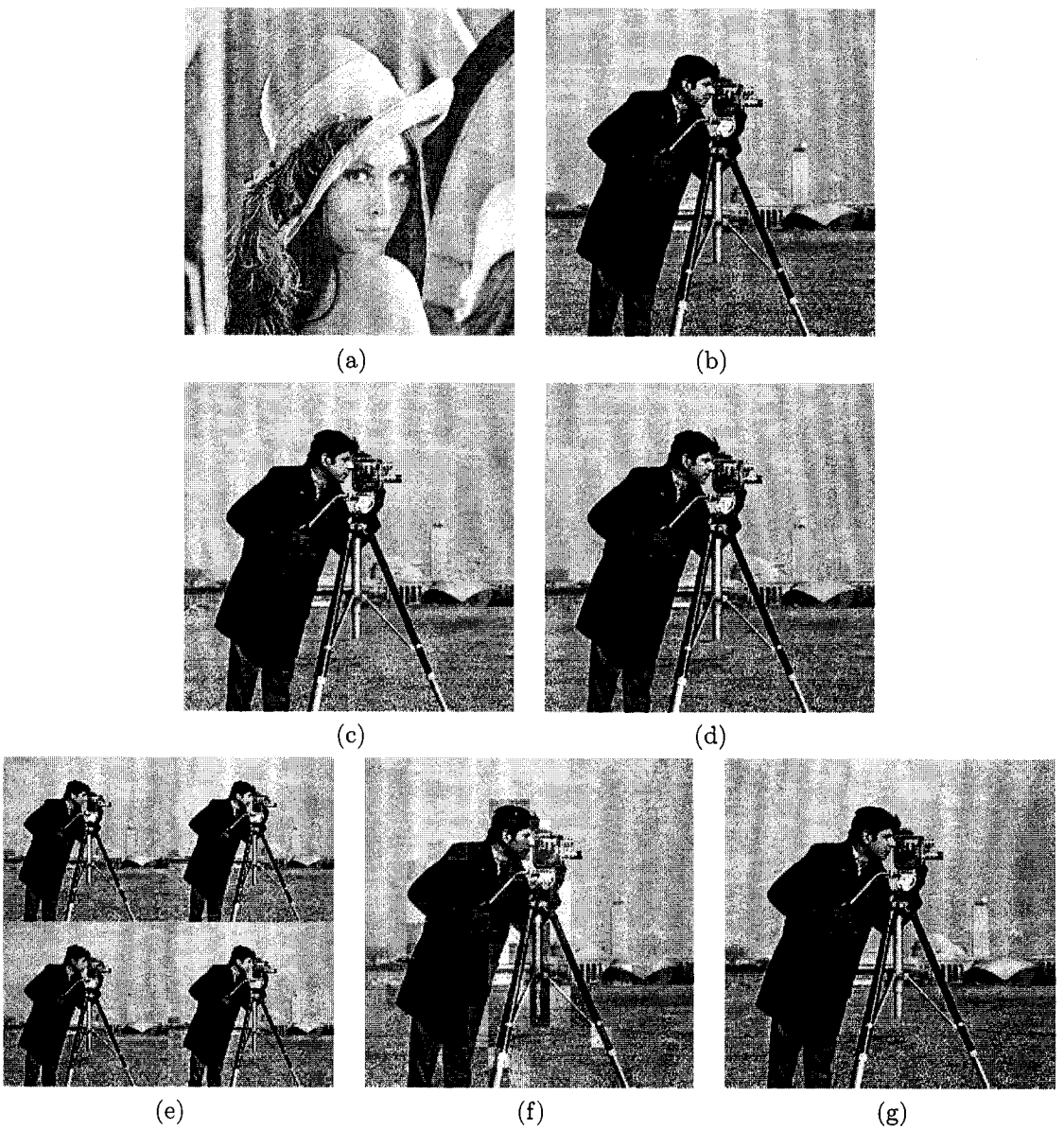


Figure 2.6: First row: (a) Cover image, (b) Visual watermark. Second row: Extracted watermark using (c) SVD-based scheme, (d) block-based SVD scheme. Third row: Extracted watermark using (e) DWT-SVD scheme, (f) ICA scheme, and (g) the proposed method.

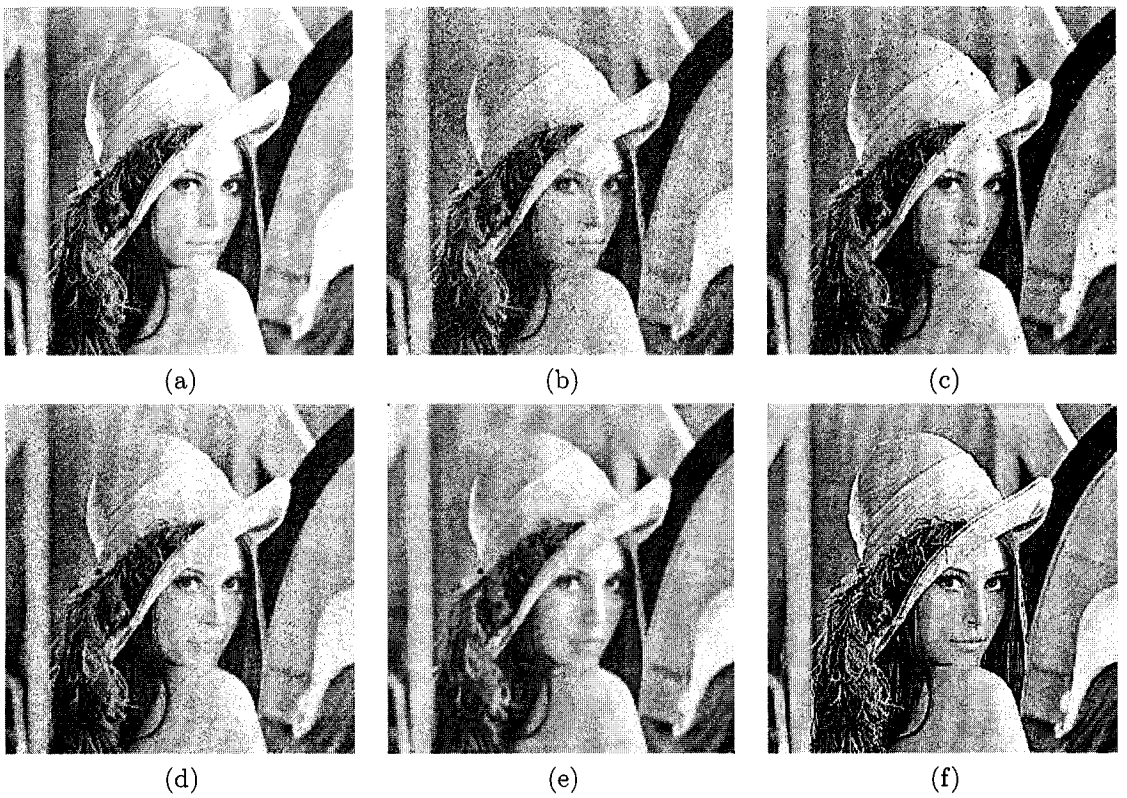
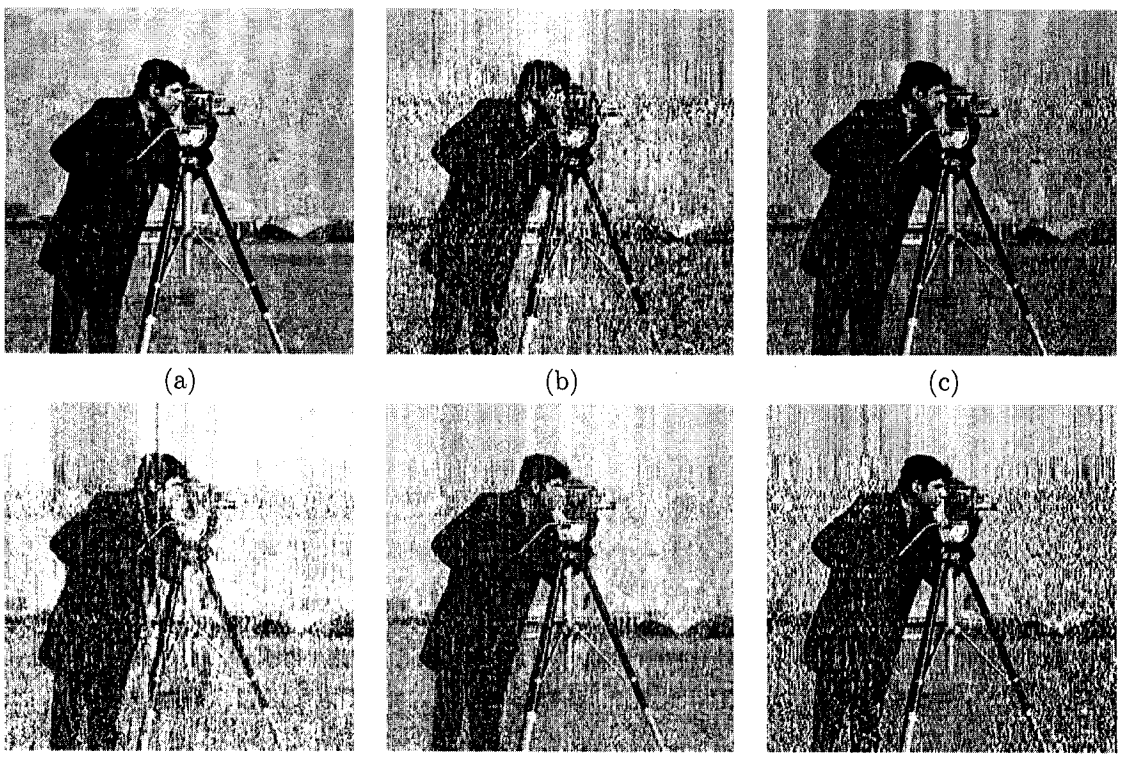


Figure 2.7: Attacked watermarked images: (a) JPEG compression 50:1, (b) Gaussian noise 0.006, (c) impulsive noise 0.01, (d) multiplicative noise 0.01, (e) median filtering, (f) sharpening 0.5.

Proposed Scheme



(a) (b) (c)

(d) (e) (f)

Figure 2.8: Extracted watermark: (a) JPEG compression, (b) Gaussian noise, (c) impulsive noise, (d) multiplicative noise, (e) median filtering, (f) sharpening.

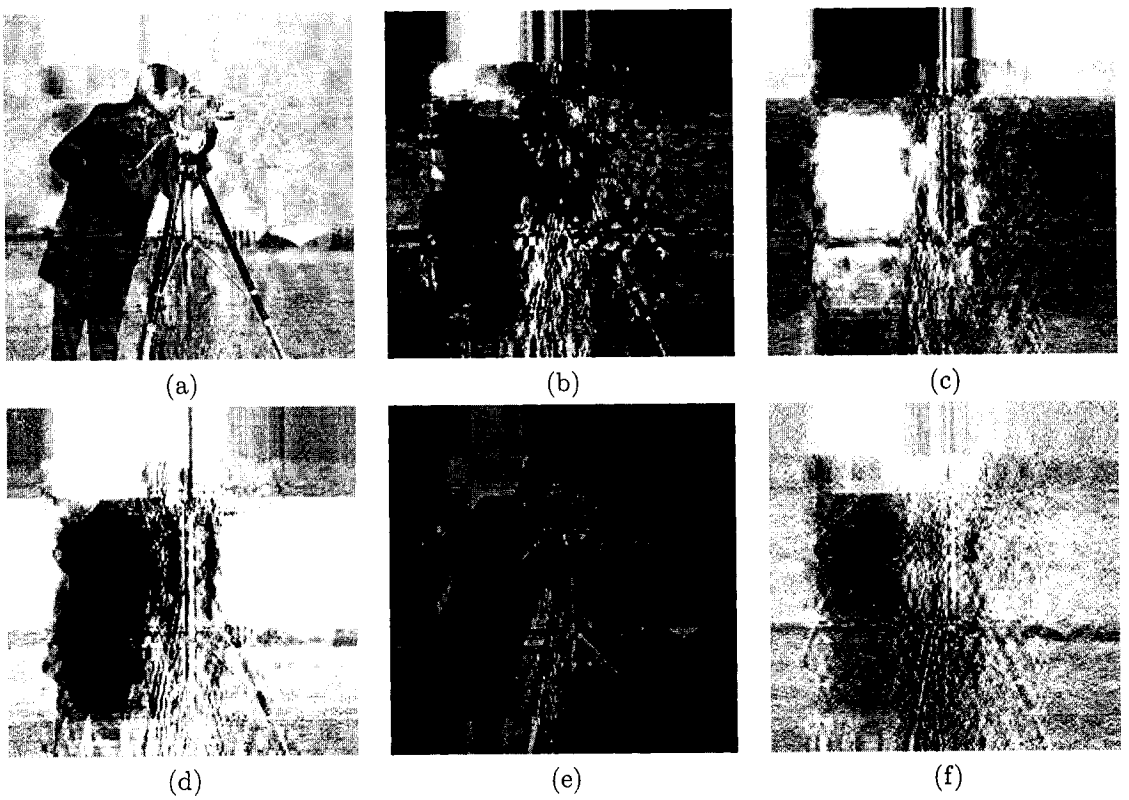


Figure 2.9: Extracted watermark: (a) JPEG compression, (b) Gaussian noise, (c) impulsive noise, (d) multiplicative noise, (e) median filtering, (f) sharpening.

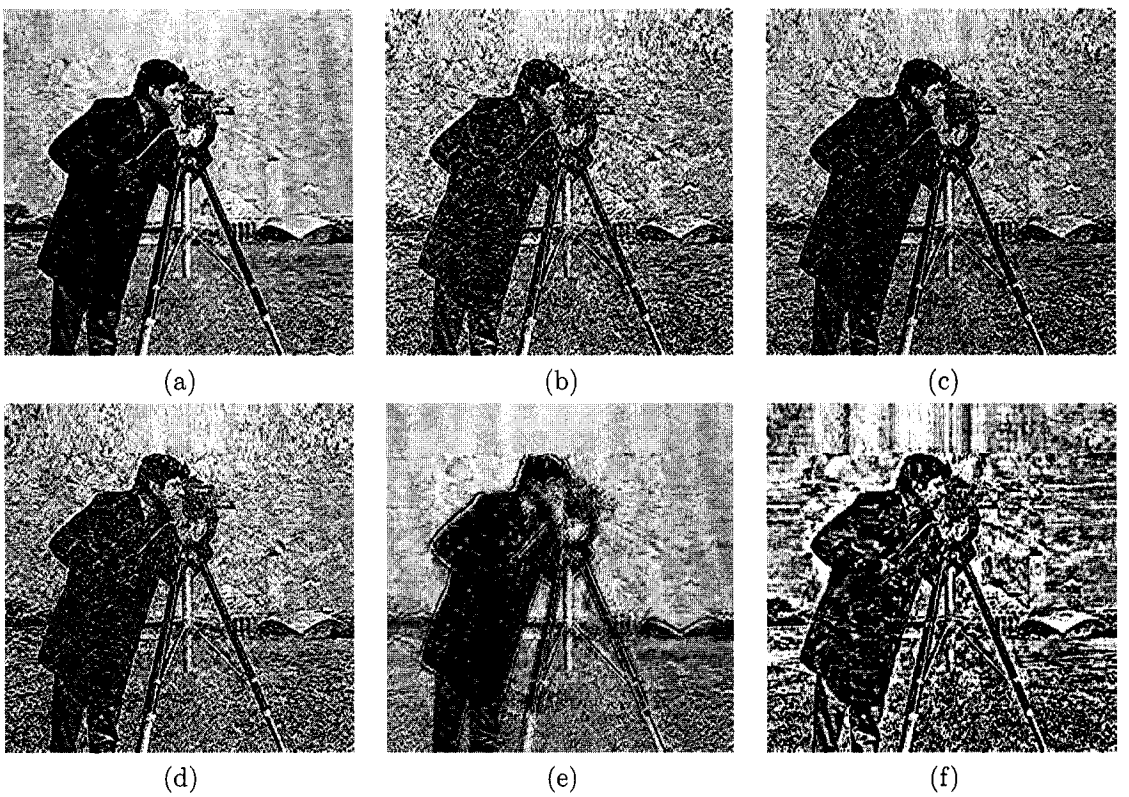
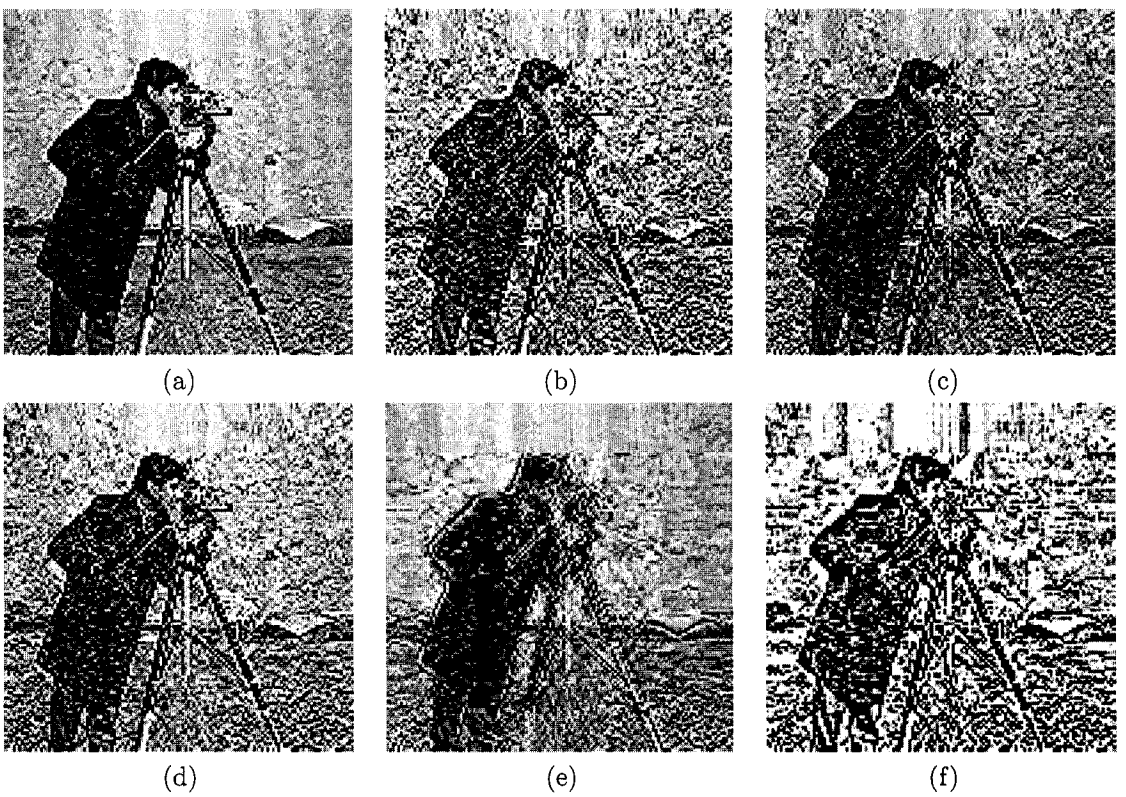


Figure 2.10: Extracted watermark: (a) JPEG compression, (b) Gaussian noise, (c) impulsive noise, (d) multiplicative noise, (e) median filtering, (f) sharpening.



(a) (b) (c)

(d) (e) (f)

Figure 2.11: Extracted watermark: (a) JPEG compression, (b) Gaussian noise, (c) impulsive noise, (d) multiplicative noise, (e) median filtering, (f) sharpening.

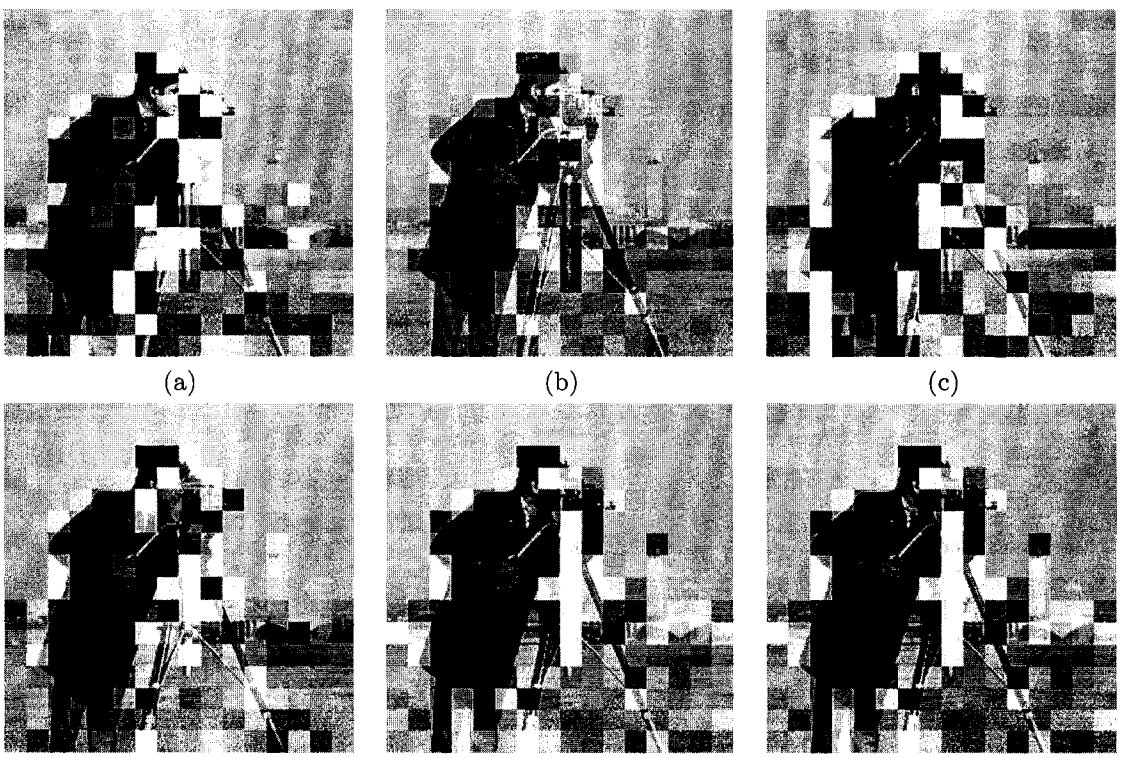


Figure 2.12: Extracted watermark: (a) JPEG compression, (b) Gaussian noise, (c) impulsive noise, (d) multiplicative noise, (e) median filtering, (f) sharpening.

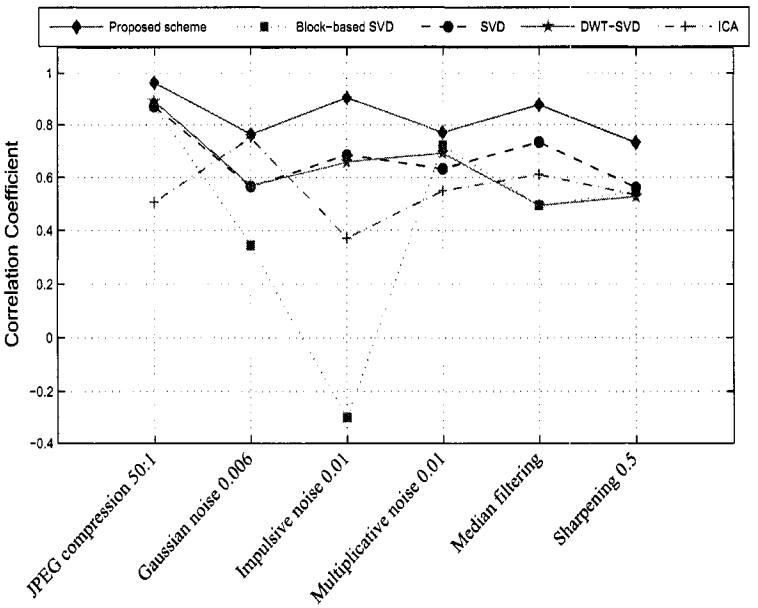


Figure 2.13: Correlation coefficient comparison results.

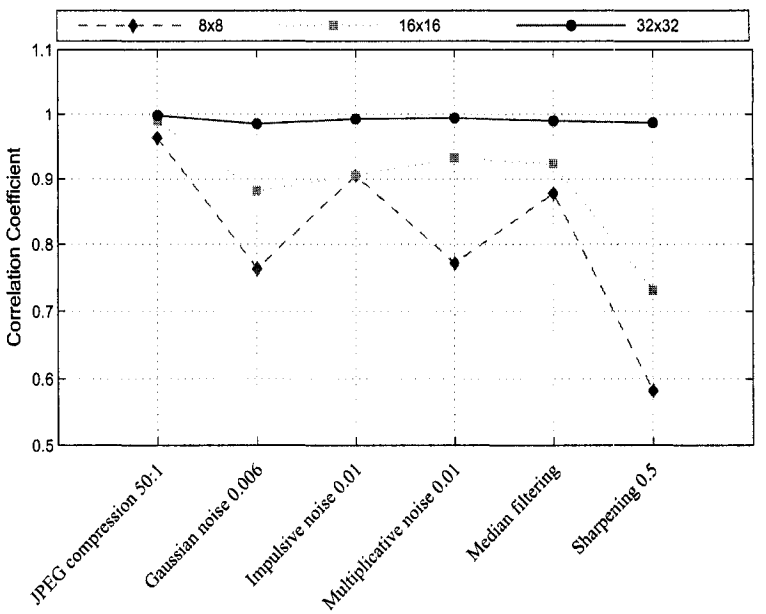


Figure 2.14: Correlation coefficient results using three different block sizes.

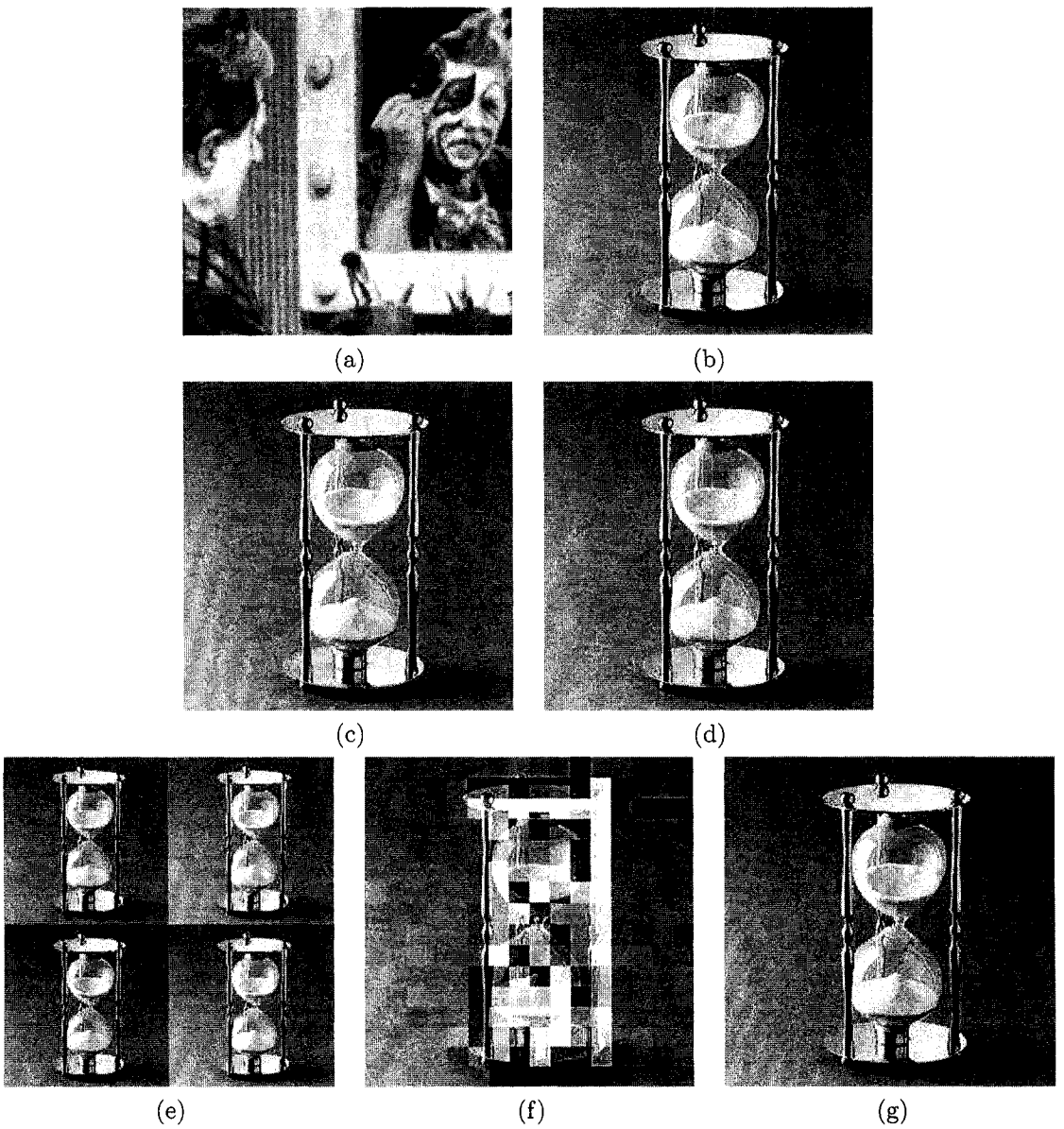


Figure 2.15: First row: (a) Cover image, (b) Visual watermark. Second row: Extracted watermark using (c) SVD-based scheme, (d) block-based SVD scheme. Third row: Extracted watermark using (e) DWT-SVD scheme, (f) ICA scheme, and (g) the proposed method.

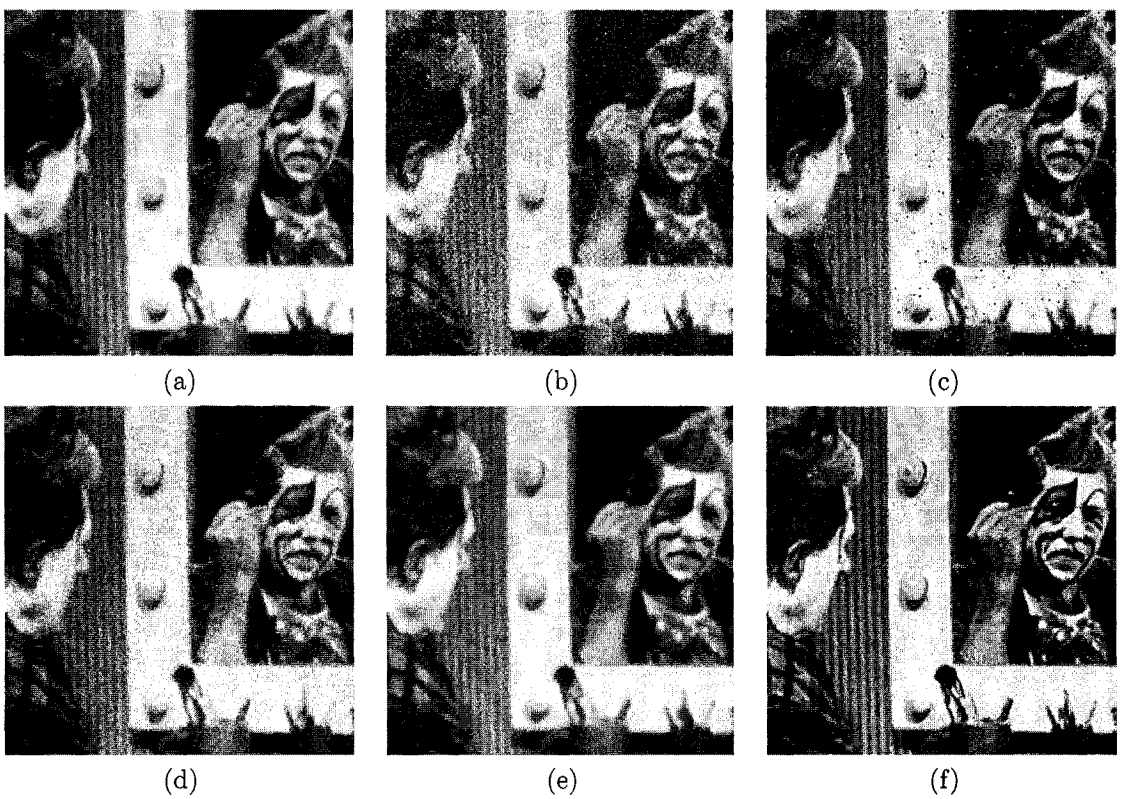
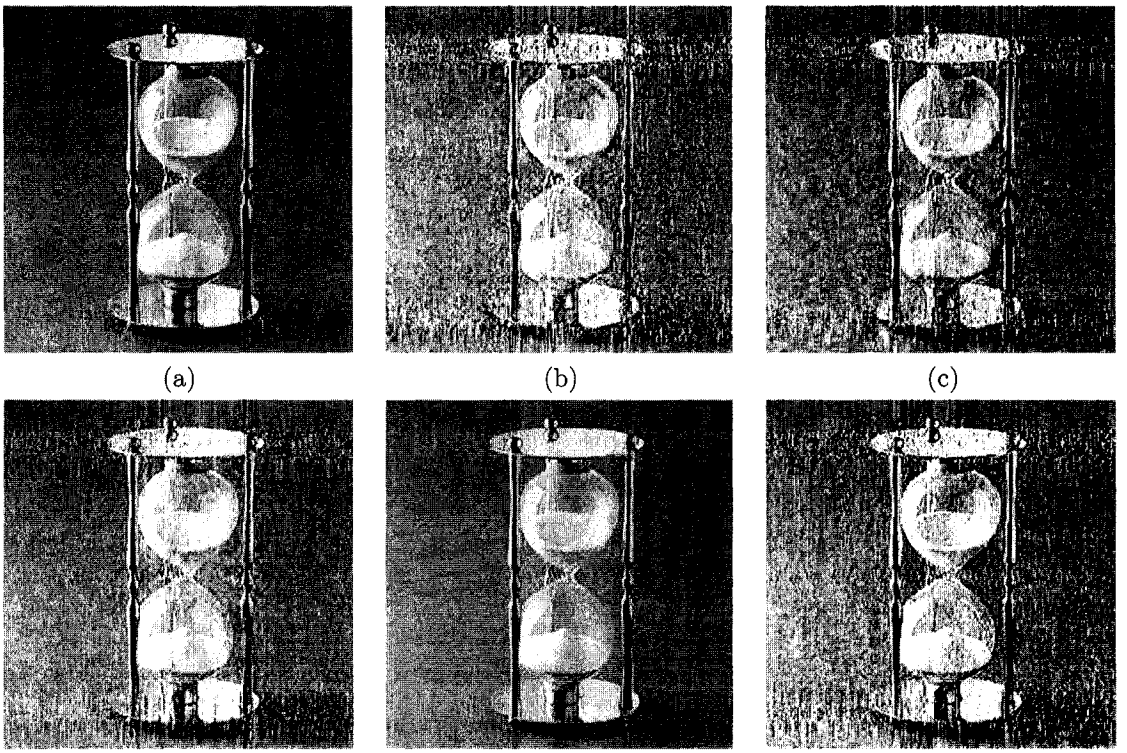


Figure 2.16: Attacked watermarked images: (a) JPEG compression 50:1, (b) Gaussian noise 0.006, (c) impulsive noise 0.01, (d) multiplicative noise 0.01, (e) median filtering, (f) sharpening 0.5.

Proposed Scheme



(a) (b) (c)

(d) (e) (f)

Figure 2.17: Extracted watermark: (a) JPEG compression, (b) Gaussian noise, (c) impulsive noise, (d) multiplicative noise, (e) median filtering, (f) sharpening.

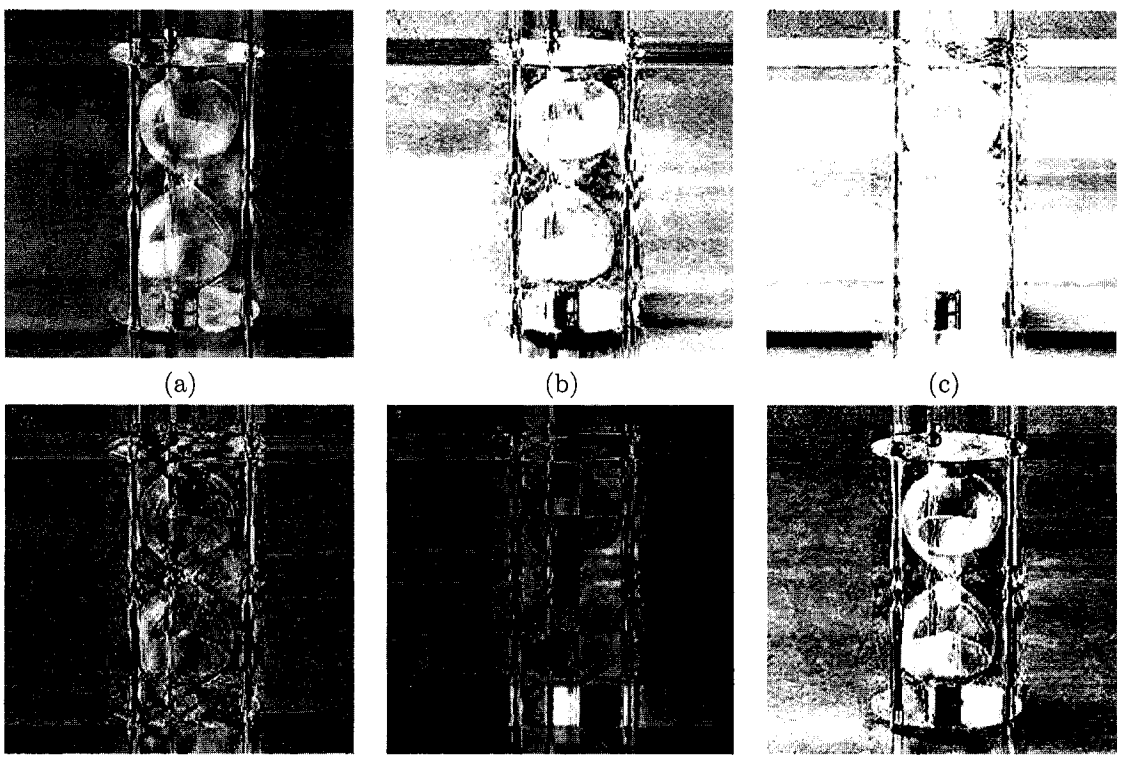
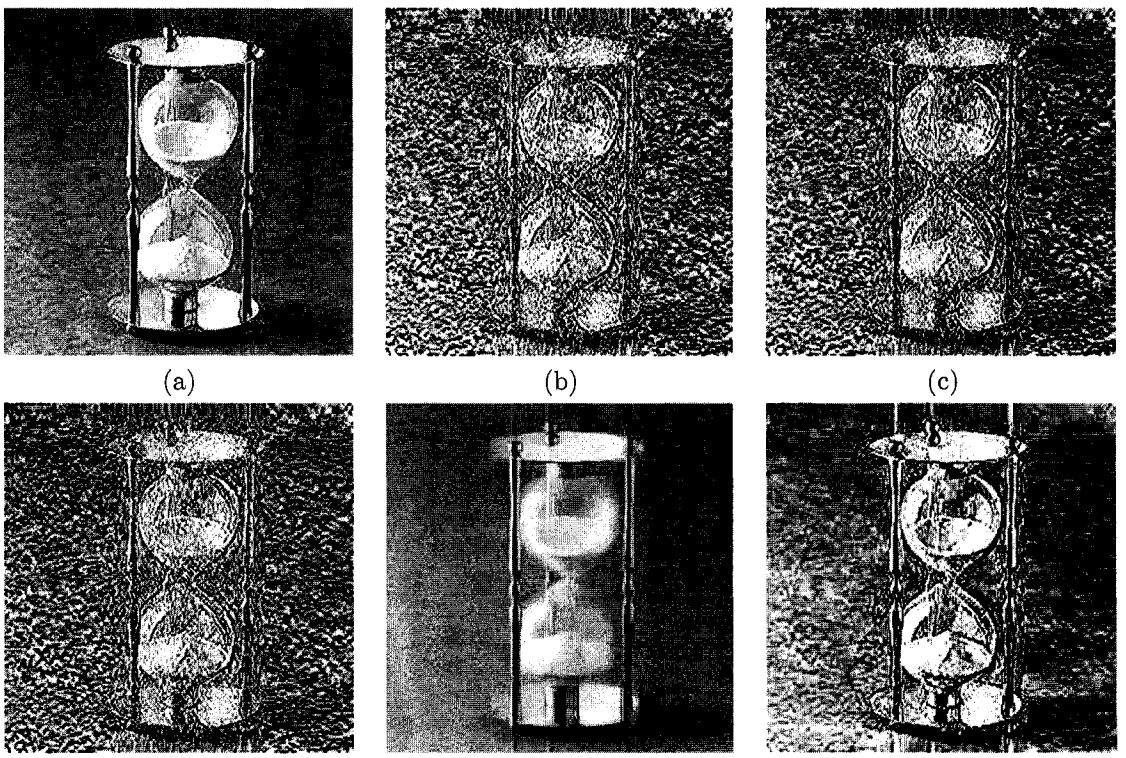


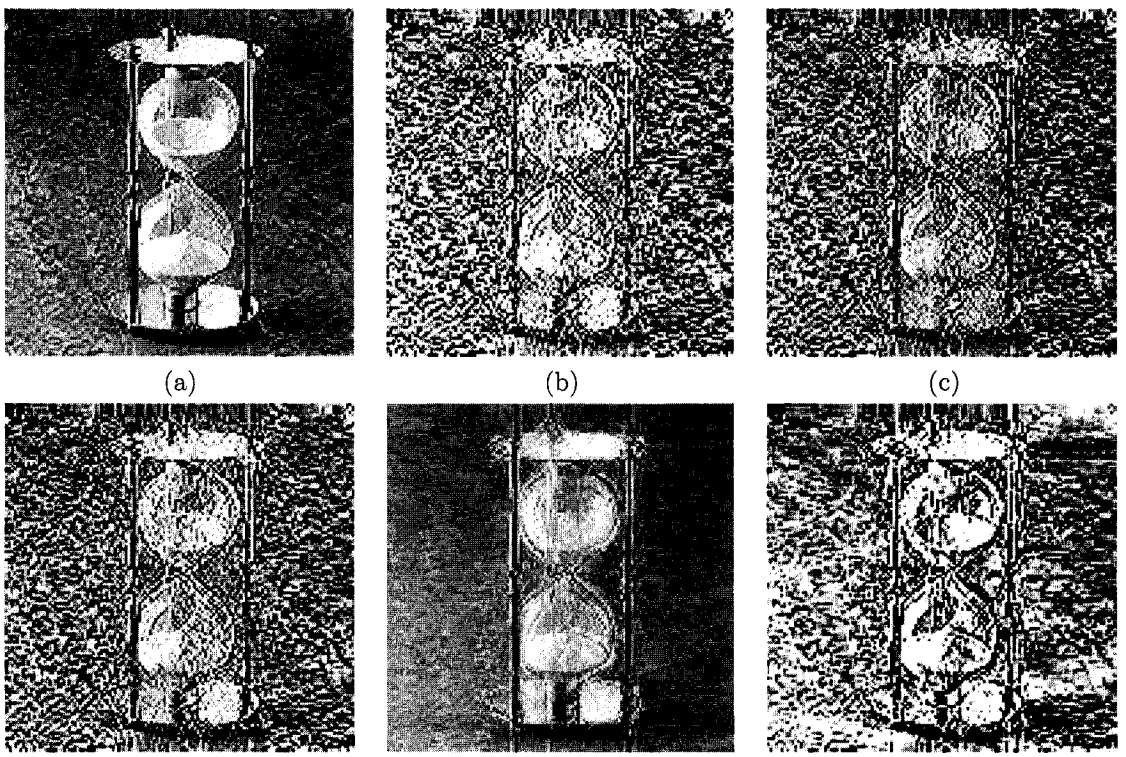
Figure 2.18: Extracted watermark: (a) JPEG compression, (b) Gaussian noise, (c) impulsive noise, (d) multiplicative noise, (e) median filtering, (f) sharpening.



(a) (b) (c)

(d) (e) (f)

Figure 2.19: Extracted watermark: (a) JPEG compression, (b) Gaussian noise, (c) impulsive noise, (d) multiplicative noise, (e) median filtering, (f) sharpening.



(a) (b) (c)

(d) (e) (f)

Figure 2.20: Extracted watermark: (a) JPEG compression, (b) Gaussian noise, (c) impulsive noise, (d) multiplicative noise, (e) median filtering, (f) sharpening.

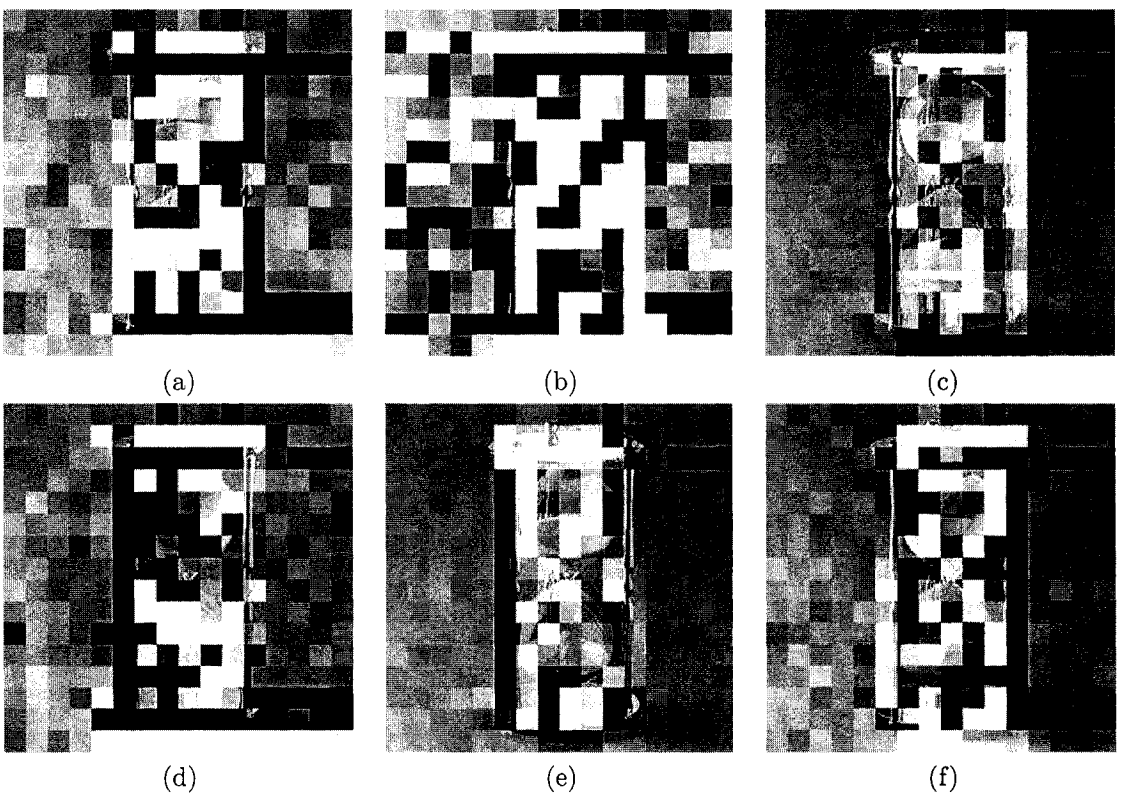


Figure 2.21: Extracted watermark: (a) JPEG compression, (b) Gaussian noise, (c) impulsive noise, (d) multiplicative noise, (e) median filtering, (f) sharpening.

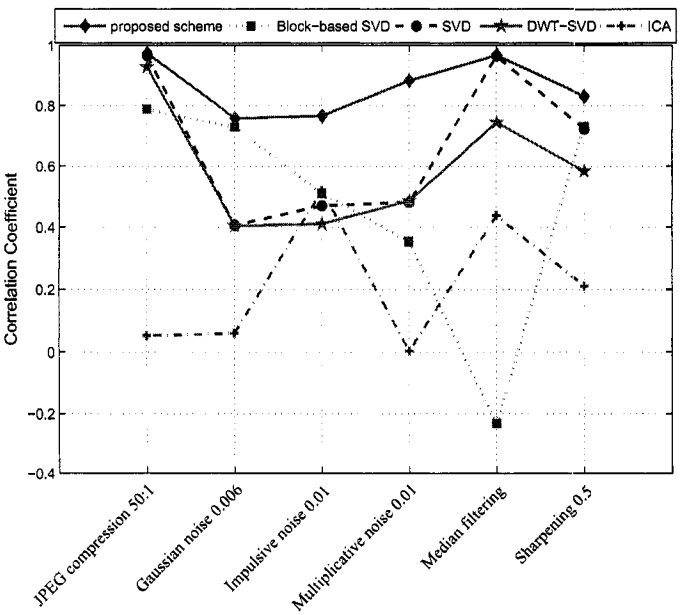


Figure 2.22: Correlation coefficient comparison results.

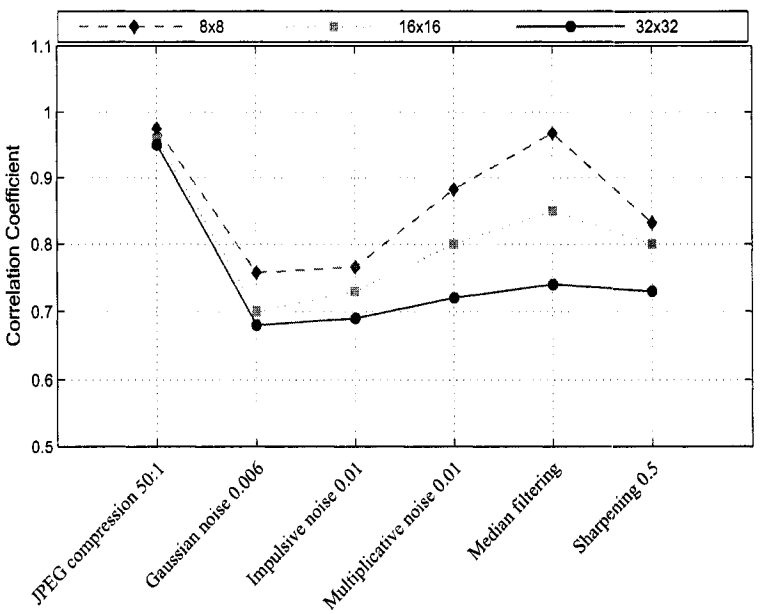


Figure 2.23: Correlation coefficient results using three different block sizes.

CHAPTER 3

Spectral Mesh Fingerprinting

In this Chapter, we present a hashing technique for 3D models using spectral graph theory and entropic spanning trees. The main idea is to partition a 3D triangle mesh into an ensemble of sub-meshes, then apply eigen-decomposition to the Laplace-Beltrami matrix of each sub-mesh, followed by computing the hash value of each sub-mesh. This hash value is defined in terms of spectral coefficients and Tsallis entropy estimate. The experimental results on a variety of 3D models demonstrate the effectiveness of the proposed technique in terms of robustness against the most common attacks including Gaussian noise, mesh smoothing, mesh compression, scaling, rotation as well as combinations of these attacks.

3.1 Introduction

The increasing use of 3D models in multimedia applications and the wide demand of online services have opened the doors for users to modify the digital content without leaving any perceptual traces. To tackle this problem, cryptographic hash functions could help in ensuring the authentication and the integrity of data. Cryptographic hash functions play an important role in modern cryptography [41]. Hash functions take an input of arbitrary length to produce an output of fixed length

referred to as hash.

The authenticity of the data can be verified by recalculating the hash value from the underlying data and comparing it to the attached hash value. Recently, Venkatesan *et al.* [42] introduced a method for robust image hashing. This technique uses randomized signal processing strategies for a non-reversible compression of images into random binary strings, and is shown to be robust against image changes that are due to compression, geometric distortions, and other attacks. Another robust image hashing technique was proposed in [43], which presents a framework for perceptual image hashing using feature points. Using on the characteristics of the visual system, significant image features are extracted by using a wavelet-based feature detection algorithm [44]. This hash algorithm withstands standard benchmark attacks and common signal processing operations. In [45], a novel algorithm for generating an image hash based on Fourier transform features and controlled randomization was proposed. This scheme shows its resiliency to content-preserving modifications.

The problem of 3D mesh hashing is relatively new compared to 2D hashing and has received less attention partly because the technology that has been used for image and video analysis cannot be easily adapted to 3D objects. Also, a large number of attacks can be applied to 3D meshes. In [46], the mesh Laplacian matrix was used to encode the 3D shape into a more compact representation by retaining the smallest eigenvalues and associated eigenvectors which contain the highest concentration of the shape information. In [47], an geometric hashing method for object recognition was presented. This method identifies objects in the presence of noise and partial occlusion. In [48], a public authentication of 3D mesh models was presented. The signature is embedded within the 3D mesh model for authentication purposes, and a new hash value is produced and compared with the value decrypted from the retrieved signature.

The primary motivation behind the proposed method is to encode the geometric and topological information of a 3D object into a hash value that may be used for a variety of rightful ownership protection purposes including authentication and integrity. Our approach partitions a 3D triangle mesh into sub-meshes and produces a hash value for each sub-mesh. To gain further insight into the proposed technique, we performed extensive numerical experiments to demonstrate the potential and the much improved performance of the proposed scheme in 3D object authentication.

The layout of this chapter is as follows. Section 2 is devoted to the problem formulation, followed by a brief background material about Laplace-Beltrami matrix and entropic spanning trees. Section 3 describes the algorithmic steps of the proposed approach. In section 4, we present experimental results to show the performance of the proposed method and its robustness against the most common attacks. Finally, we conclude in section 5.

3.2 Problem Formulation

In computer graphics and geometric-aided design, 3D objects are usually represented as triangle meshes. A triangle mesh $\mathbb{M}$ may be defined as $\mathbb{M} = (\mathcal{V}, \mathcal{E})$ or $\mathbb{M} = (\mathcal{V}, \mathcal{T})$, where $\mathcal{V} = \{\boldsymbol{v}_1, \ldots, \boldsymbol{v}_m\}$ is the set of vertices, $\mathcal{E} = \{e_{ij}\}$ is the set of edges, and $\mathcal{T} = \{\boldsymbol{t}_1, \ldots, \boldsymbol{t}_n\}$ is the set of triangles. Each edge $e_{ij} = [\boldsymbol{v}_i, \boldsymbol{v}_j]$ connects a pair of vertices $\{\boldsymbol{v}_i, \boldsymbol{v}_j\}$. Two distinct vertices $\boldsymbol{v}_i, \boldsymbol{v}_j \in \mathcal{V}$ are adjacent if they are connected by an edge, i.e. $e_{ij} \in \mathcal{E}$. The neighborhood of a vertex $\boldsymbol{v}_i$ is the set $\boldsymbol{v}_i^\star = \{\boldsymbol{v}_j \in \mathcal{V} : \boldsymbol{v}_i \sim \boldsymbol{v}_j\}$, and the degree d_i of a vertex $\boldsymbol{v}_i$ is simply the cardinality of $\boldsymbol{v}_i^\star$. We also denote by $\mathcal{T}(\boldsymbol{v}_i^\star)$ the set of triangles of $\boldsymbol{v}_i^\star$.

The objective of 3D hashing is to design a robust hash function that produces a unique identifier for a 3D model, while satisfying three main requirements [41]. First, given a 3D model $\mathbb{M}$ and a

hash function H, the computation of the hash value $h = H(\mathbb{M})$ must be easy. Second, Given h, it is hard to find a 3D model $\widetilde{\mathbb{M}}$ such that $h = H(\widetilde{\mathbb{M}})$. Third, two different 3D models should not produce the same hash value.

3.2.1 Laplace-Beltrami matrix of a triangle mesh

Let $\boldsymbol{v}_i \in \mathcal{V}$, the Laplace-Beltrami operator $\Delta_m \boldsymbol{v}_i$ is defined as

$$\Delta_m \boldsymbol{v}_i = \frac{3}{\eta} \sum_{\boldsymbol{v}_j \in \boldsymbol{v}_i^*} (\cot \alpha_{ij} + \cot \beta_{ij})(\boldsymbol{v}_j - \boldsymbol{v}_i),$$

where α_{ij} and β_{ij} are the angles $\angle \boldsymbol{v}_i \boldsymbol{v}_{j-1} \boldsymbol{v}_j$ and $\angle \boldsymbol{v}_i \boldsymbol{v}_{j+1} \boldsymbol{v}_j$ respectively (see Fig. 3.1), and η is the sum of all areas of neighboring triangles defined as $\eta = \sum_{\boldsymbol{t}_j \in \mathcal{T}(\boldsymbol{v}_i^*)} area(\boldsymbol{t}_j)$.

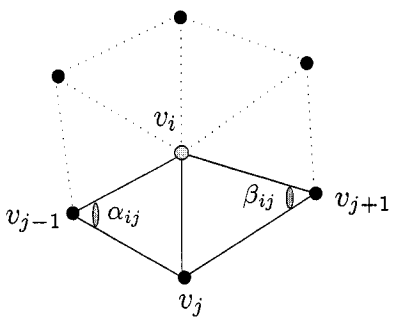


Figure 3.1: Illustration of Laplace-Beltrami angles α_{ij} and β_{ij}

We may define the Laplace-Beltrami matrix as

$$L = D - W = \begin{cases} d_i - \omega_{ii} & \text{if } \boldsymbol{v}_i = \boldsymbol{v}_j \\ -\omega_{ij} & \text{if } \boldsymbol{v}_i \sim \boldsymbol{v}_j \\ 0 & \text{otherwise} \end{cases}$$

where $W = (w_{ij})$ denotes the weighted adjacency matrix with $w_{ij} = 3(\cot \alpha_{ij} + \cot \beta_{ij})/\eta$, and $D = \text{diag}\{d_i : \boldsymbol{v}_i \in \mathcal{V}\}$ is the degree matrix with diagonal entries $d_i = \sum_{\boldsymbol{v}_j \in \boldsymbol{v}_i^*} \omega_{ij}$.

Fig. 3.2 illustrates an example of a 3D triangle mesh and its sparse Laplace-Beltrami matrix.

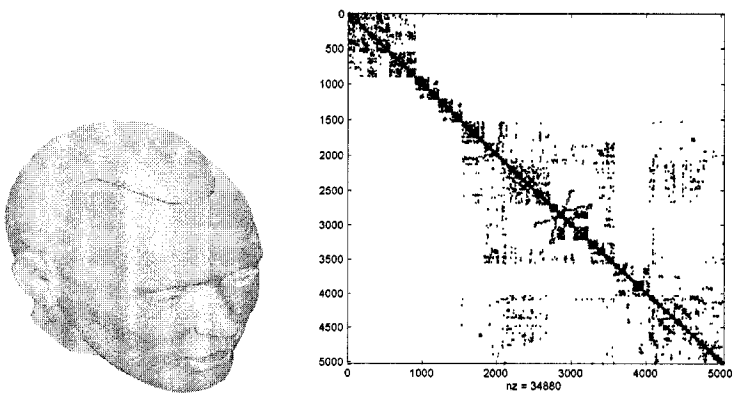


Figure 3.2: 3D triangle mesh and its Laplace-Beltrami matrix.

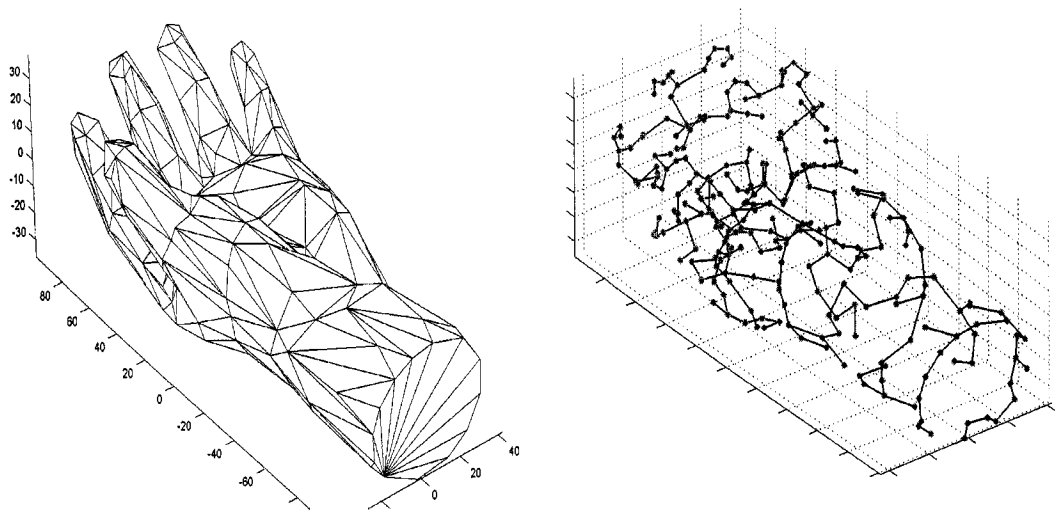
3.2.2 Entropic spanning tree

A spanning tree is a connected acyclic graph that passes through all the elements of the vertex set $\mathcal{V}$, and it is specified by an ordered list of edges e_{ij} connecting certain pairs $(\boldsymbol{v}_i, \boldsymbol{v}_j), i \neq j$, along with a list of edge adjacency relations [51].

Recently, there has been a concerted research effort in statistical physics to explore the properties of Tsallis entropy, leading to a statistical mechanics that satisfies many of the properties of the standard theory [49]. When a system is composed of two statistically independent subsystems, then Shannon or Rényi entropy of the composite system is just the sum of entropies of the individual systems, and hence the correlations between the subsystems are not accounted for. Tsallis entropy, however, does take into account these correlations due to its pseudo-additivity property [49].
We may define an estimator $\widehat{H}_\alpha$ of Tsallis entropy as follows

$$\widehat{H}_\alpha(\mathcal{V}) = \frac{1}{1-\alpha}\left[\frac{L^\star(\mathcal{V})}{\beta m^\alpha} - 1\right], \tag{1}$$

where $L^{\star}(\mathcal{V})$ is the total length of the minimal spanning tree [51], α is referred to as an entropic index, and β is a constant playing a role of bias correction [50]. We employ Kruskal's algorithm [51] to compute the minimal spanning tree.

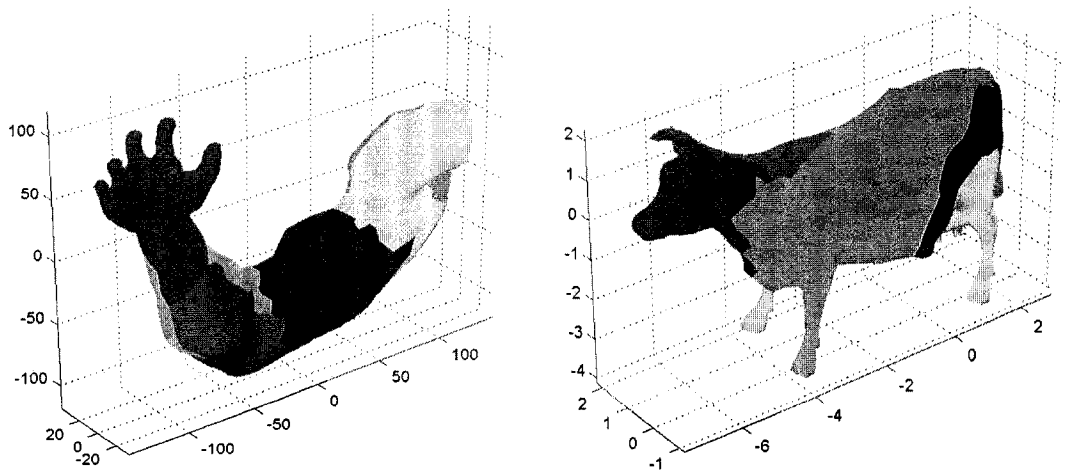


(a) (b)

Figure 3.3: Illustration of an MST. (a) Hand model, (b) the MST.

3.3 Proposed Method

The proposed algorithm is based on the concepts of entropic spanning trees and the eigen-decomposition of the Laplace-Beltrami matrix. However, calculating of the eigenvalues and the eigenvectors of a typically large $m \times m$ Laplace-Beltrami matrix is prohibitively expensive $\mathcal{O}(m^3)$. To circumvent this limitation, we first partition the original 3D mesh into sub-meshes and then apply eigen-analysis to each sub-mesh. To this end, we implemented a 3D mesh partitioning algorithm based on MeTiS [52–54]. Fig. 3.4 shows an example of 3D models partitioned into eight sub-meshes.



(a) (b)

Figure 3.4: 3D mesh partitioning: each sub-mesh is colored randomly. (a) Arm model, (b) Cow model.

The algorithmic steps of the proposed 3D hashing approach may be summarized as follows:

1) Partition a 3D object $\mathbb{M}$ into s sub-meshes: $\mathbb{M} = \cup_{k=1}^{s} \mathbb{M}_k$, where the cardinality of the vertex set $\mathcal{V}_k$ of each sub-mesh $\mathbb{M}_k$ is equal to $|\mathcal{V}_k| = m_k$.

1.1) Apply Kruskal's algorithm to each $\mathcal{V}_k$ in order to compute the entropy value ξ_k of each sub-mesh

$$\xi_k \simeq 2\left[\frac{L^{\star}(\mathcal{V}_k)}{\sqrt{m_k}} - 1\right], \quad k = 1, \ldots, s$$

where the Tsallis entropic index α is set to $1/2$.

1.2) Apply eigen-decomposition to the Laplace-Beltrami matrix L_k of each sub-mesh, that is $L_k = B_k \Lambda_k B_k^T$, where $B_k = (\boldsymbol{b}_1, \boldsymbol{b}_2, \ldots, \boldsymbol{b}_{m_k})$ is an orthogonal matrix whose columns are eigenvectors which form a spectral basis, and $\Lambda_k = \mathrm{diag}(\lambda_i : i = 1, \ldots m_k)$ is a diagonal

matrix of eigenvalues arranged in decreasing order of magnitude.

1.3) Retain the r ($r < m_k$) significant spectral basis vectors which account for most of the energy.

1.4) Compute the hash value μ_k of each sub-mesh according to the formula given by $\mu_k = \sum_{i=1}^{r} \lambda_i^{\xi_k} \|\boldsymbol{b}_i\|^2$

2) Stack the hash values of all the sub-meshes into a single vector $\boldsymbol{h} = (\mu_1, \mu_2 \ldots, \mu_s)$, which we refer to as the hash vector.

3.4 Experimental Results

We tested the performance of the proposed hashing method using a variety of 3D models. Fig. 3.5 shows a sample of six 3D objects.

Fig. 3.6 depicts examples of the 3D camel and 3D cow partitioned meshes.

Fig. 3.7(a) through Fig. 3.7(h), and Fig. 3.8(a) through Fig. 3.8(h) depict the minimal spanning trees for each sub-mesh of the 3D camel and 3D cow models, as well as their corresponding hash values μ_k, where $k = 1, \ldots, 8$.

To test the robustness of the proposed hashing algorithm, we applied several attacks to the 3D models including scaling, rotation, mesh smoothing, mesh simplification, Gaussian noise, and Gaussian noise combined with compression. We evaluated the performance of the proposed scheme by computing the normalized correlation ρ between the resulting hash vectors as follows

$$\rho = \frac{|\boldsymbol{h}_1 \cdot \boldsymbol{h}_2|}{\|\boldsymbol{h}_1\| \, \|\boldsymbol{h}_2\|}, \tag{2}$$

where $\boldsymbol{h}_1$ and $\boldsymbol{h}_2$ are the hash vectors before and after the attack respectively. The correlation results shown in Table 3.1 clearly demonstrate the good performance of the proposed method in

terms of robustness against the attacks. Moreover, this good performance is in fact consistent with all the 3D models used for experimentation.

Table 3.1: Normalized hash correlation results with different 3D models

Attacks	3D Models					
	Camel	Cow	Shark	Triceratops	Baby	Arm
Mesh Scaling with X*2	0.9674	0.9908	0.7539	0.7755	0.5907	0.7704
Mesh Scaling with Y*2	0.9625	0.9910	0.7565	0.7864	0.6017	0.7784
Mesh Scaling with Z*2	0.9553	0.9221	0.7433	0.7637	0.5916	0.7673
Rotation around X 45°	0.9534	0.9880	0.7486	0.7807	0.5864	0.7812
Rotation around Y 45°	0.9534	0.9880	0.7486	0.7807	0.5864	0.7812
Rotation around Z 45°	0.9534	0.9880	0.7486	0.7807	0.5864	0.7812
Mesh Smoothing (10 iterations)	0.9531	0.9886	0.7343	0.7765	0.5922	0.7643
Mesh Simplification (70%)	0.7965	0.8448	0.7247	0.7641	0.8862	0.7621
Gaussian Noise ($\sigma = 0.25$)	0.9572	0.9893	0.7606	0.7931	0.6153	0.7886
Gaussian Noise + Compression (25%)	0.9617	0.9902	0.7450	0.7868	0.6198	0.7843

Uniqueness is an important factor that needs to be take into consideration when dealing with hash functions. As mentioned earlier, the hash value produced by our proposed method should always be unique. Therefore, we compared the hash vectors between different 3D models using the normalized correlation coefficient to check whether the proposed hash vector fulfills the requirement of uniqueness. The results are listed in Table 3.2. It is apparent that the proposed method shows very good performance in terms of the ability to distinguish different 3D models and to produce different hash values.

Table 3.2: Normalized correlation between different 3D model hashes

	Camel	Cow	Shark	Triceratops	Baby	arm
Camel	1	0.7987	0.7458	0.7668	0.8112	0.7834
Cow	0.7987	1	0.7842	0.8590	0.8855	0.7921
Shark	0.7458	0.7842	1	0.7816	0.5129	0.7172
Triceratops	0.7668	0.8590	0.7816	1	0.7179	0.7145
Baby	0.8112	0.8855	0.5129	0.7179	1	0.8328
Arm	0.7834	0.7921	0.7172	0.7145	0.8328	1

3.5 Conclusions

In this Chapter, we proposed a robust hashing scheme for 3D models. The approach consists of partitioning a 3D model into sub-meshes, followed by applying eigen-decomposition to the Laplace-Beltrami matrix of each sub-mesh in order to obtain the hash values of all the sub-meshes. The performance of the proposed method was evaluated through extensive experiments which clearly showed satisfactory resiliency against a variety of attacks.

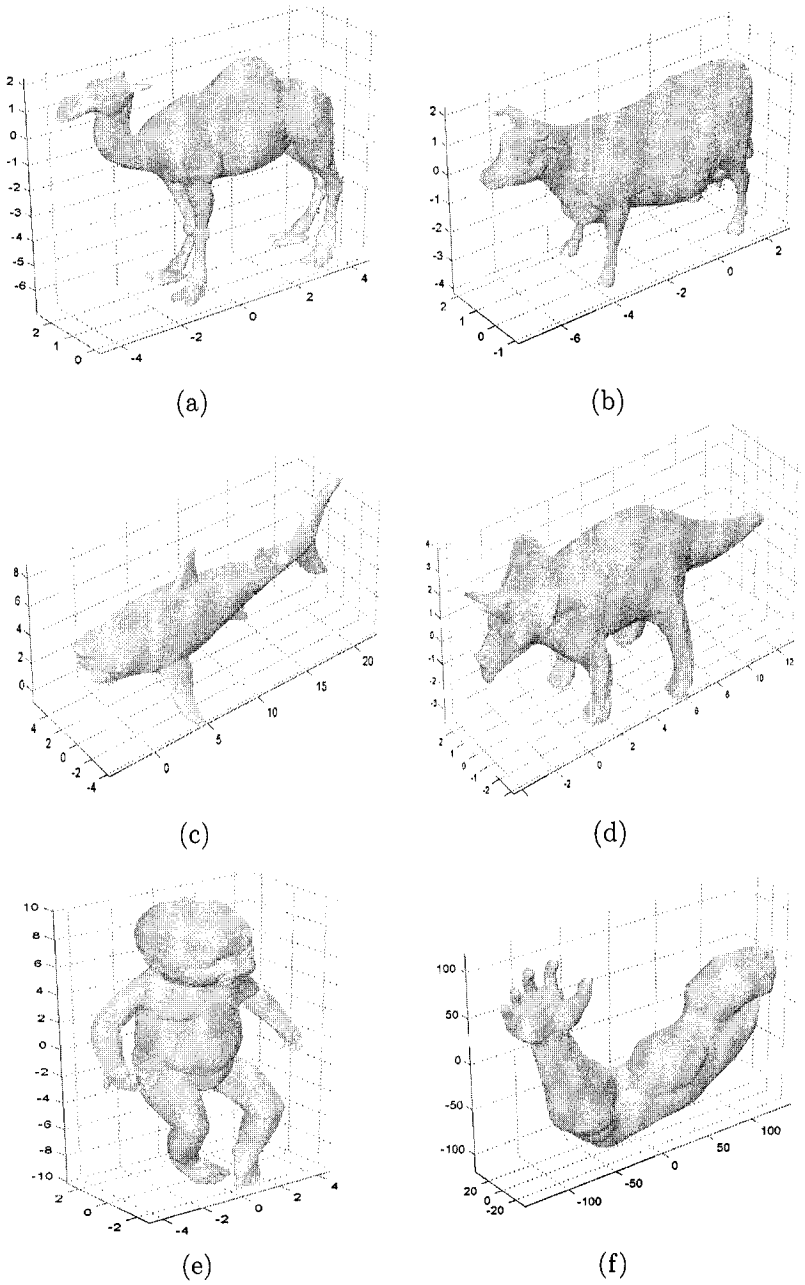
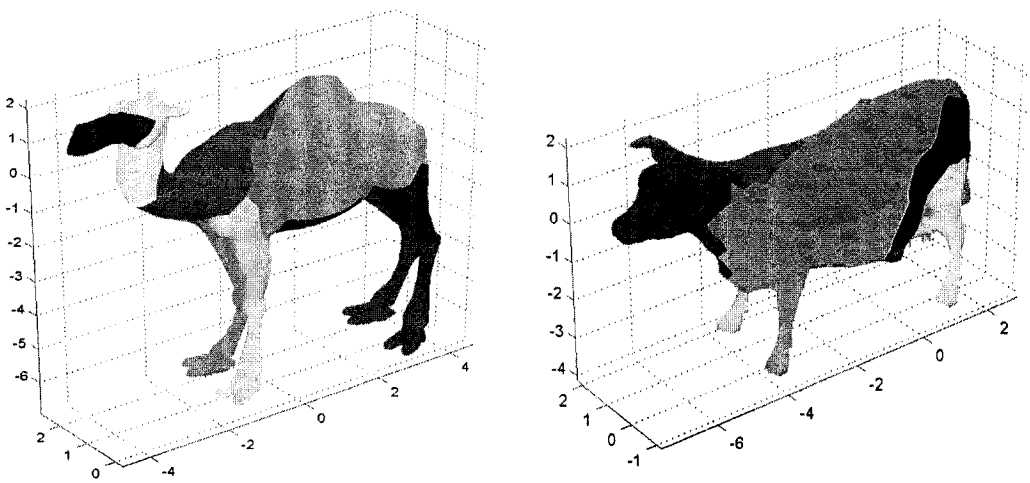
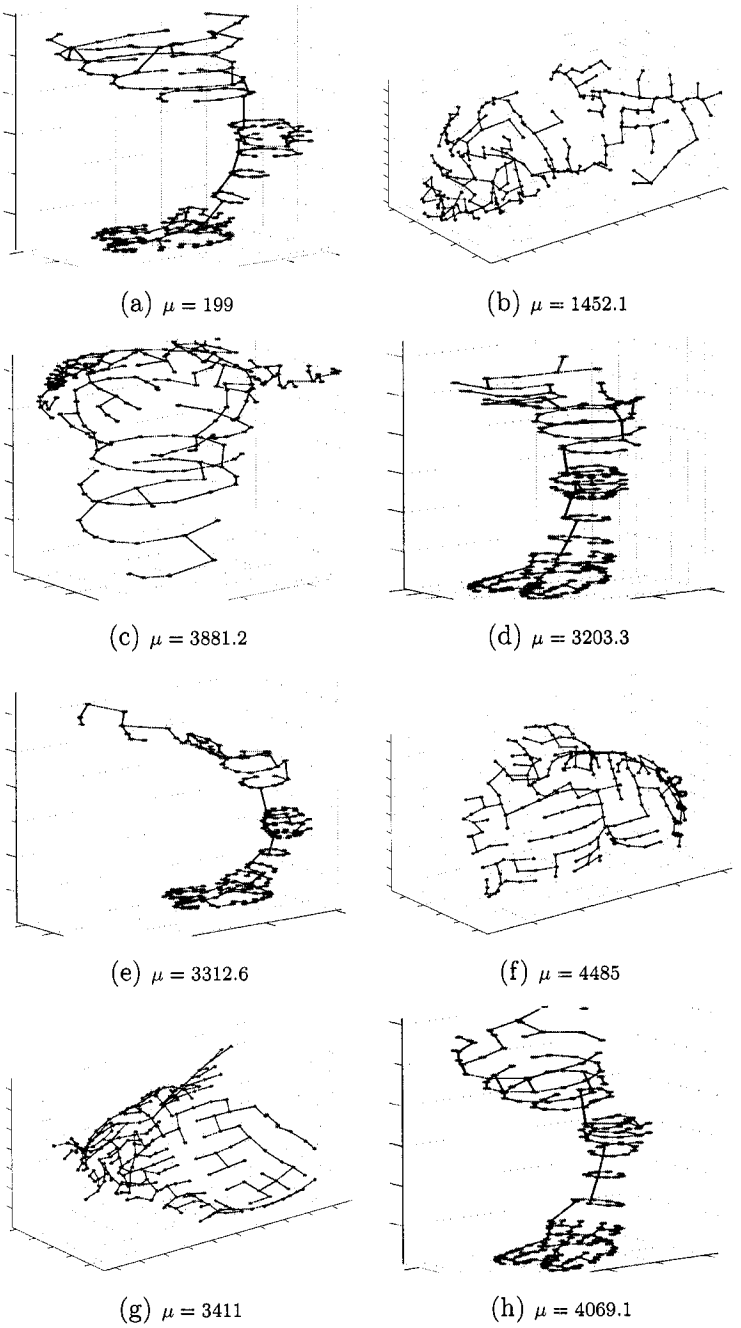


Figure 3.5: 3D models used for experimentation: (a) Camel, (b) Cow, (c) Shark, (d) Triceratops, (e) Baby, (f) Arm.



(a) (b)

Figure 3.6: Partitioned 3D models. (a) camel, (b) cow



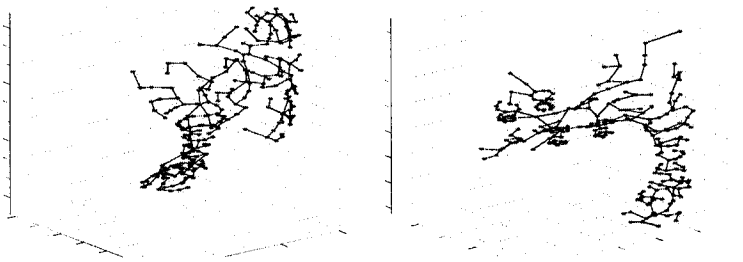
(a) $\mu = 199$ (b) $\mu = 1452.1$

(c) $\mu = 3881.2$ (d) $\mu = 3203.3$

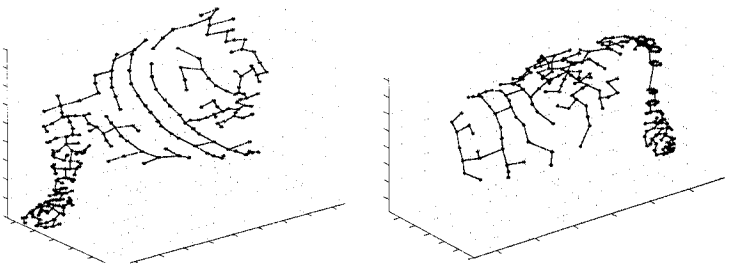
(e) $\mu = 3312.6$ (f) $\mu = 4485$

(g) $\mu = 3411$ (h) $\mu = 4069.1$

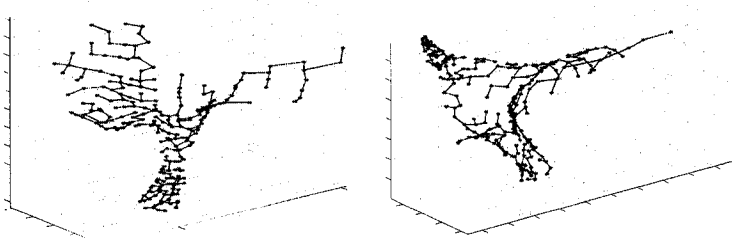
Figure 3.7: MST of the 3D camel sub-meshes and their corresponding hash values : (a) Head, (b) Neck, (c)-(d) Front feet, (e) Hump, (f) Shoulders, (g)-(h) Back.



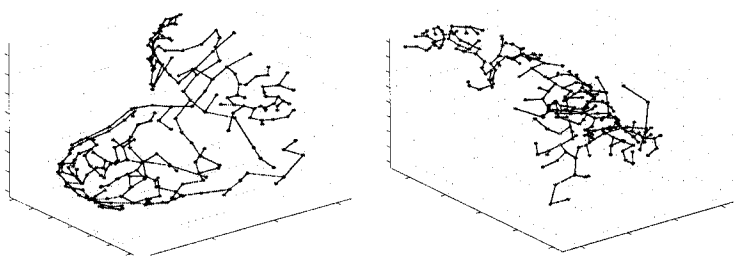
(a) $\mu_1 = 4650.6$ (b) $\mu_2 = 4422$



(c) $\mu_3 = 5771$ (d) $\mu_4 = 6829.3$

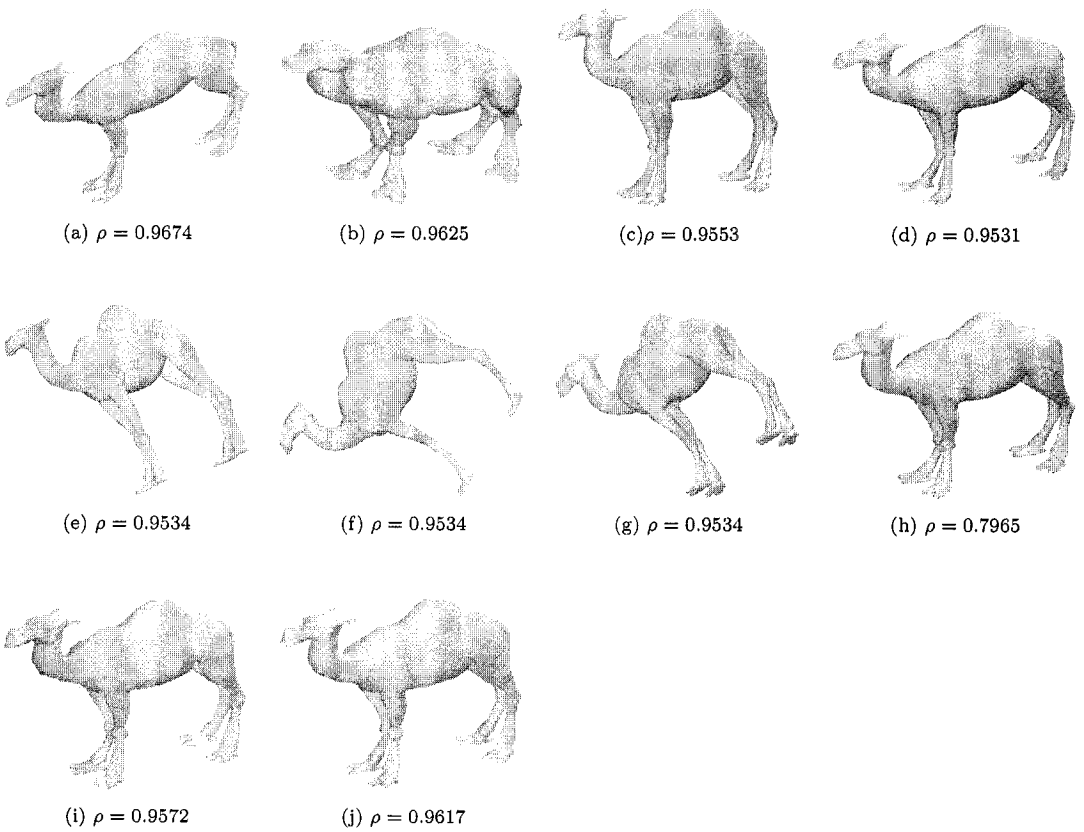


(e) $\mu_5 = 4264.9$ (f) $\mu_6 = 3408.7$



(g) $\mu_7 = 1453.4$ (h) $\mu_8 = 1696.4$

Figure 3.8: Minimal spanning trees of the 3D cow sub-meshes and their corresponding hash values: (a)-(b) back feet, (c)-(e) front feet, (d) back-tail, (f) neck, (g)-(h) head-horn.



(a) $\rho = 0.9674$ (b) $\rho = 0.9625$ (c)$\rho = 0.9553$ (d) $\rho = 0.9531$

(e) $\rho = 0.9534$ (f) $\rho = 0.9534$ (g) $\rho = 0.9534$ (h) $\rho = 0.7965$

(i) $\rho = 0.9572$ (j) $\rho = 0.9617$

Figure 3.9: Illustration of the 3D camel model with different attacks. (a) scaling with X-axis, (b) scaling with Y-axis, (c) scaling with Z-axis, (d) mesh smoothing 10 iterations, (e) rotating around X-axis 45^o, (f) rotating around Y-axis 45^o, (g) rotating around Z-axis 45^o, (h) simplification 70% , (i) Gaussian noise $\sigma = 0.25$, (j) Gaussian noise combined with compression 25%.

CHAPTER 4

Conclusions and Future Work

This thesis has presented robust watermarking schemes for multimedia protection as well as 3D mesh fingerprinting. We have demonstrated the performance of the proposed algorithms through extensive experiments, and we compared our techniques with existing methods. A variety of images are used in the experiments to show the effectiveness of the proposed schemes. We have achieved to balance between the imperceptibility of the watermarked image and its robustness against intentional and geometric attacks. In addition, we have developed a 3D mesh fingerprinting technique.

In the next Section, the contributions made in each of the previous chapters and the concluding results drawn from the associated research work are presented. Suggestions for future research directions related to this thesis are provided in Section 4.2.

4.1 Contributions of the thesis

4.1.1 Image watermarking using SVD and NMF transforms

We proposed a watermarking scheme using singular value decomposition transform and nonnegative matrix factorization. We tested our proposed watermarking technique through extensive

experiments. The experimental results clearly demonstrate the much improved performance of the proposed watermarking approach in terms of watermark imperceptibility and robustness against several attacks.

4.1.2 3D mesh fingerprinting

We proposed an new methodology for 3D object fingerprinting. The key idea is to partition a 3D model into sub-meshes, followed by applying the eigen-decomposition to each sub-mesh and obtain the fingerprint values of all sub-meshes. The performance of the proposed method was evaluated through extensive experiments which clearly showed excellent resiliency against multiple attacks.

4.2 Future research directions

Several interesting research directions motivated by this thesis are discussed next. In addition to designing robust watermarking schemes for multimedia protection, we intend to accomplish the following projects in the near future:

4.2.1 Image watermarking using fast Hadamard, MPDFRF and wavelet transforms

The watermarking scheme introduced in Chapter 2 provides robustness against several attacks. The performance of this approach, however, is not resilient when applying some geometric attacks such as salt and pepper noise, gamma correction, histogram equalization, and sharpening to the watermarked image. To improve the proposed technique, we present a watermarking scheme that uses MPDFRFT, fast Hadamard transform(FHT), and DWT. Based on a variety of studies, 2D

Hadamard transform has been used with great success for image compression and image watermarking. The elements of the basis vectors of the Hadamard transform take only the binary values +1 and -1. Therefore, the FHT is well suited for digital image processing applications where computational simplicity is required.

4.2.2 3D image watermarking scheme using nonnegative transition matrix factorization and wavelet transform

Inspired by the good experimental results we obtained when applying the image watermarking algorithm using nonnegative matrix factorization and wavelet transform. We would like to extend the proposed algorithm and develop a robust watermarking scheme for 3D models. The key idea is to apply the transition matrix to the 3D mesh and decompose it into four wavelet sub-bands and then apply NMF to the blocks of each sub-band.

4.2.3 Spectral 3D mesh watermarking

3D mesh compression technique can play a good role in improving the watermarking system. The 3D compression technique we proposed has led us to find a better way to embed the watermark. A 3D model can be partitioned into smaller sub-meshes, then apply the umbrella compression technique to each sub mesh, followed by embedding a watermark in the spectral coefficients of the compressed 3D meshes.

List of References

[1] I. Cox, M. L. Miller, and J. A. Bloom, *Digital watermarking*, Morgan Kaufmann Publishers Inc., 2001.

[2] M. Arnold, M. Schmucker, and S. D. Wolthusen, *Techniques and Applications of Digital Watermarking and Content Protection*, Artech House, 2003.

[3] A. Nikolaidis, S. Tsekeridou, A. Tefas, and V. Solachidis, "A survey on watermarking application scenarios and related attacks," *proc. IEEE International Conference on Image Processing*, vol. 3, pp. 991- 994, October 2001.

[4] F. Hartung and M. Kutter, "Multimedia watermarking techniques," *Proceedings of the IEEE*, vol. 87, no. 7, pp. 1079-1107, 1999.

[5] S. P. Mohanty, K. R. Ramakrishnan, and M. S. Kankanhalli, "A DCT domain visible watermarking technique for images," *proc. IEEE International Conference on Multimedia and Expo*, pp. 1029- 1032, 2000.

[6] S. Craver, N. Memon, B. L. Yeo, and M. M. Yeung, "Resolving rightful ownerships with

invisible watermarking techniques: Limitations, attacks, and implications," *IEEE Journal on Selected Areas in Communications,* vol. 16, no. 4, pp. 573- 586, May 1998.

[7] W. Yongdong, "On the security of an SVD-based ownership watermarking," *IEEE Transaction on Multimedia,* vol. 7, no. 4, pp.624- 627, August 2005.

[8] I. J. Cox, J. Killian, T. Leighton, and T. Shamoon "Secure spread spectrum watermarking for multimedia," *IEEE Trans. Image Processing* , vol. 6, no. 12, pp. 1673-1687, 1997.

[9] R. Liu and T. Tan, "A SVD-based watermarking scheme for protecting rightful ownership," *IEEE Trans. Multimedia*, vol. 4, no. 1, pp. 121-128, 2002.

[10] M. Barni, F. Bartolini, V. Cappelini and A. Piva, "A DCT-domain system for robust image watermarking," *Signal Processing,* vol.66, pp. 357-372, 1998.

[11] M. Kutter and F. Hartung. Introduction to Watermarking Techniques, Chapter 5 of "Information hiding: Techniques for Steganography and digital watermarking," S. Katzenbeisser and F. A. P. Peticolas(eds.), Norwood, MA: Artech house, pp. 97-120, 2000.

[12] J. Dittmann and F. Nack, "Copyright - copywrong," *IEEE Transactions on Multimedia*, vol. 7, pp. 14-17, 2000.

[13] V. Fotopoulos and A. N. Skodras, "A subband DCT approach to image watermarking," *Proc. European Signal Processing Conference*, Finland, 2000.

[14] Y. Wang, J. F. Doherty, R. E. Van Dyck, "A Wavelet-Based Watermarking Algorithm-for Ownership Verification of Digital Images," *IEEE Transactions on Image Processing*, vol. 11, no. 2, pp. 77-88, Feb 2002.

[15] E. Ganic and A. M. Eskicioglu, "Robust embedding of visual watermarks using DWT-SVD," *Journal of Electronic Imaging*, October-December 2005.

[16] E. Ganic, N. Zubair, and M. Eskicioglu, "An optimal watermarking scheme based on singular value decomposition," *Proc. Comm., Network, and Information Security*, pp. 89-90, December 2003.

[17] J. Kusyk and A. M. Eskicioglu, "A Semi-blind logo watermarking scheme for color images by comparison and modification by comparison and modification of DFT coefficients," *Proc. Multimedia Systems and Applications Conference*, Boston, MA, October 2005.

[18] P. Tao and A. M. Eskicioglu, "A robust multiple watermarking scheme in the DWT domain," *Proc. Optics East Symposium, Internet Multimedia Management*, 2004.

[19] R. Mehul and R. Priti, "Discrete wavelet transform based multiple watermarking scheme," *proc. IEEE Region 10 Technical Conference on Convergent Technologies for the Asia-Pacific, Bangalore*, India, 2003.

[20] O. G. Pla, E. T. Lin, and E. J. Delp, "A Wavelet Watermarking Algorithm Based on a Tree Structure," *proc. International Conference on Security, Steganography, and Watermarking of Multimedia Contents*, vol. 5306, pp. 571-580, San Jose, Jan. 2004.

[21] A. Sverdlov, S. Dexter, and A. M. Eskicioglu, " Robust DCT-SVD domain image watermarking for copyright protection : embedding data in all frequencies," *Proc. Euro. Signal Processing Conference,* Turkey, 2005.

[22] S. Gilani and A. Skodras,"Watermarking by multiresolution Hadamard transform," *proc. Electronic Imaging and Visual Arts*, pp. 73- 77, Florence, Italy, 2001.

[23] E. E. Abdallah, A. Ben Hamza, and P. Bhattacharya, "An improved image watermarking scheme using fast Hadamard and discrete wavelet transforms," *Journal of Electronic Imaging*, 2007.

[24] E. E. Abdallah, A. Ben Hamza, and P. Bhattacharya, "A robust block-based image watermarking scheme using fast Hadamard transform and singular value decomposition," *Proc. Interntional Conference on Pattern Recognition*, vol. 3, pp. 673-676, 2006.

[25] I. J. Cox, J. Kilian, T. Leighton, and T. Shamoon, "Secure Spread Spectrum Watermarking of Images, Audio and Video," *Proc. IEEE International Conf. on Image Processing*, pp. 243-246, Switzerland, September 1996.

[26] M.D. Swanson, M. Kobayashi, A.H. Tewfik, "Data hiding for multimedia personalization, interaction, and protection," *Proceedings of the IEEE*, vol. 86, no. 6, pp. 1064-1087, 1998.

[27] B.B. Zhu, M.D. Swanson, A.H. Tewfik, "When seeing isn't believing," *IEEE Signal Processing Magazine*, vol. 21, no. 2, pp. 40-49, 2004.

[28] E. Abdallah, A. Ben Hamza, P. Bhattacharya, "Spectral graph-theoretic approach to 3D mesh watermarking," *Proc. ACM Graphics Interface Conference*, Montréal, Canada, 2007.

[29] N. Memon and P. Wong, "Digital watermarks: protecting multimedia content," *Communications of the ACM*, vol. 47, no. 7, pp. 35-43, 1998.

[30] A. Hyvarinen, J. Karhunen and E. Oja, *Independent Component Analysis*, Wiley, 2001.

[31] F.J. Gonzalez-Serrano, H.Y. Molina-Bulla, and J.J. Murillo-Fuentes, 'Independent component

analysis applied to digital image watermarking," *Proc. IEEE International Conference on Acoustics Speech and Signal Processing*, pp. 1997-2000, 2001.

[32] J.J. Murillo-Fuentes, "Independent component analysis in the blind watermarking of digital images," *Neurocomputing*, vol. 70, no. 16-18, pp. 2881-2890, 2007.

[33] D. Lee and H. Seung, "Learning the parts of objects by nonnegative matrix factorization," *Nature*, vol. 401, pp. 788-791, 1999.

[34] D. Lee and H. Seung, "Algorithms for nonnegative matrix factorization," *Advances in Neural Information Processing Systems*, vol. 13, pp. 556-562, 2001.

[35] A. Ben Hamza and D.J. Brady, "Reconstruction of reflectance spectra using robust nonnegative matrix factorization," *IEEE Transactions on Signal Processing*, vol. 54, no. 9, pp. 3637-3642, 2006.

[36] M. Ghaderpanah, A. Ben Hamza, "A nonnegative matrix factorization scheme for digital image watermarking," *Proc. IEEE Int. Conference on Multimedia & Expo*, Toronto, Canada, 2006.

[37] M. Ghaderpanah, A. Ben Hamza, "NMF-based watermarking scheme for multimedia protection," *Proc. IEEE Int. Symposium on Industrial Electronics*, Montréal, Canada, 2006.

[38] M. Ghaderpanah, A. Ben Hamza, "Secure copyright protection of digital images using nonnegative matrix factorization," *Proc. IEEE Biennial Symposium on Communications*, Kingstone, Canada, 2006.

[39] M. Ghaderpanah, A.Abbas, and A. Ben Hamza, "Entropic hashing of 3D objects using Laplace-Beltrami operator," *Proc. IEEE Int. Conference on Image Processing*, San Diego, USA, 2008.

[40] Stirmark 4.0, http://www.petitcolas.net/fabien/watermarking/stirmark/

[41] A. J. Menezes , P.C van Oorschot, and S.A. Vanstone, *Handbook of Applied Cryptography*, CRC Press, 1996.

[42] R. Venkatesan, S. M. Koon, M.H. Jakubowski, and P. Moulin, "Robust image hashing," *Proc. IEEE Int. Conf. Image Processing*, Vancouver, Canada, 2000.

[43] V. Monga and B.L. Evans, "Perceptual image hashing via feature points: performance evaluation and tradeoffs," *IEEE Trans. Image Processing*, vol. 15, no. 11, pp. 3452-3465, 2006.

[44] A. Meixner and A. Uhl, "Analysis of a wavelet based robust hash algorithm," *Proc. SPIE Security, Steganography Watermarking of Multimedia Contents*, San Jose, CA, 2004.

[45] A. Swaminathan, Y. Mao and M. Wu, " Robust and secure image hashing," *IEEE Trans. on Information Forensics and Security*, vol. 1, no. 2, pp. 215-230, 2006.

[46] Z. Karni and C. Gotsman, "Spectral compression of mesh geometry," *Proc. ACM SIGGRAPH*, pp. 279-286, 2000.

[47] B. Lamiroy and P. Gros, "Rapid object indexing and recognition using enhanced geometric hashing," *Proc. ECCV*, pp. 59-70, 1996.

[48] H. Wu and Y. Cheung, "Public authentication of 3D mesh models," *Proc. IEEE/WIC/ACM Int. Conf. Web Intelligence*, pp. 940-946, Hong Kong, 2006.

[49] C. Tsallis, "Possible generalization of Boltzmann-Gibbs statistics," *Jour. Statistical Physics*, vol. 52, pp. 479-487, 1988.

[50] A.O Hero, B. Ma, O. Michel, and J. Gorman, "Applications of entropic spanning graphs," *IEEE Signal Processing Magazine*, vol. 19, no. 5, pp. 85-95, 2002.

[51] B.Y. Wu and K.M.Chao, *Spanning Trees and Optimization Problems*, Chapman & Hall/CRC, 2004.

[52] G. Karypis and V. Kumar, "MeTiS: A software package for partitioning unstructured graphs, partitioning meshes, and computing fill-reducing orderings of sparse matrices," *Version 4.0, University of Minnesota*, 1998.

[53] G.L. Miller, S.H. Teng, W. Thurston, And S.A. Vavasis. "Automatic mesh partitioning," *Institute for Mathematics and Its Applications*, vol. 56, 1993.

[54] G.L. Miller, S.H. Teng, W. Thurston, and S.A. Vavasis "Geometric separators for finite-element meshes," *SIAM Jour. on Scientific Computing*, vol. 19, no. 2, pp. 364-386, 1998.