DIGITAL WATERMARKING IN MEDICAL IMAGES

A Thesis Submitted for the Degree of Doctor of Philosophy

By

Jasni Mohamad Zain

School of Information Systems, Computing and Mathematics
Brunel University

November 2005

To my Parents,

Mohamad Zain Said &

Sabariah Yaacob

To my Husband,

Mohamed Fauzi Abdul Rani

To my Children,

Luqman,

Hilmi,

Hannah,

Syahid &

Tawfeq

Abstract

This thesis addresses authenticity and integrity of medical images using watermarking. Hospital Information Systems (HIS), Radiology Information Systems (RIS) and Picture Archiving and Communication Systems (PACS) now form the information infrastructure for today's healthcare as these provide new ways to store, access and distribute medical data that also involve some security risk. Watermarking can be seen as an additional tool for security measures.

As the medical tradition is very strict with the quality of biomedical images, the watermarking method must be reversible or if not, region of Interest (ROI) needs to be defined and left intact. Watermarking should also serve as an integrity control and should be able to authenticate the medical image.

Three watermarking techniques were proposed. First, Strict Authentication Watermarking (SAW) embeds the digital signature of the image in the ROI and the image can be reverted back to its original value bit by bit if required. Second, Strict Authentication Watermarking with JPEG Compression (SAW-JPEG) uses the same principal as SAW, but is able to survive some degree of JPEG compression. Third, Authentication Watermarking with Tamper Detection and Recovery (AW-TDR) is able to localise tampering, whilst simultaneously reconstructing the original image.

Table of Contents

List of Figures

List of Tables

Abbreviations

CDMA	Code Division Multiple Access
DCT	Discrete Cosine Transform
DE	Digital Envelope
DICOM	Digital Imaging and Communications in Medicine
DWT	Discrete Wavelet Transform
ECC	Error Correction Code
EPR	Electronic Patients Record
HVS	Human Visual System
LSB	Least Significant Bit
JPEG	Joint Picture Expert Group
JPEG-LS	JPEG Lossless Scheme
LUT	Look Up Table
MAC	Message Authentication Code
MD5	Message Digest by Ron Rivest
MSB	Most Significant Bit
NEMA	National Electrical Manufacturers' Association
PN	Pseudorandom Noise
RLE	Run Length Encoding
RSA	Rivest, Shamir and Adleman public key encryption
SHA	Secure Hash Algorithm
TIFF	Tag Image File Format
VPN	Virtual Private Network
VQ	Vector Quantization
VW2D	Variable-Watermark Two-Dimensional Algorithm

Acknowledgements

Alhamdulillahirabbil 'alamin.

I would like to thank my supervisors, Dr Malcolm Clarke, Prof. Ray Paul and Dr Lynne Baldwin for their invaluable advice and assistance throughout the course of this research.

Declaration

The following papers have been published, or submitted for publication, as a direct result of this research.

J. M. Zain and M. Clarke, "Fragile Image Watermarking with One-to-One Block Mapping ", accepted for presentation in MMU International Symposium on Information and Communication Technologies, Petaling Jaya, Malaysia, 24-25th November 2005.

Jasni M Zain and Malcolm Clarke, "LSB Reversible Watermarking Surviving JPEG Compression", in The 27th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, Shanghai, China, 1-4 September 2005.

J. M. Zain, L.P. Baldwin and M. Clarke, (in submission) "Reversible Watermarking for Authentication of Medical Image", Journal of Advancing Information and Management Studies.

Jasni Zain and Malcolm Clarke, "Issues in watermarking medical images", PREP2005, University of Lancaster, April 2005.

Jasni Zain and Malcolm Clarke, "Security In Telemedicine: Issues in Watermarking Medical Images", 3rd International Conference Sciences of Electronic, Technologies of Information and Telecommunications (SETIT 2005), Susa, Tunisia, 27-31 March 2005.

Jasni Zain, "Security in Telemedicine: Watermarking medical images ", Medical Error and Technologies Research Workshop. London, 3 November 2004.

J.M Zain, L. P. Baldwin, M. Clarke, " Reversible watermarking for authentication of DICOM images", in The 26th Annual International Conference of the IEEE Engineering in Medicine and Biology Society, San Francisco, California, 1-4 September 2004.

J. M. Zain, M. Clarke, L. P. Baldwin, "The effect of reversible LSB manipulation to the quality of image", in PREP2004, University of Hertfordshire, April 2004.

J Zain and R S H Istepanian, "Digital Watermarking in Wireless Telemedical Environment", in Proceedings of PREP2003, Exeter University, Exeter, April 2003.

J M Zain, "Globalization and Telecommunication Technologies", in MRG 2nd annual Conference, Manchester, September 2003.

J M Zain, "Threats and Challenges in Securing Telemedicine System", in MRG 2nd annual Conference, Manchester, September 2003.

Chapter 1

Security of Medical Images

1.1 Introduction to Information Security

One of the major concerns throughout the world today is to make high quality healthcare available to all. Traditionally, part of the difficulty in achieving equitable access to healthcare has been that the provider and the recipient must be physically present in the same place. Recent advances in information and communication technologies have increased the number of ways in which healthcare can be delivered to reduce these difficulties.

Telemedicine, the area where medicine and information and communications technology (ICT) meet, is probably the part of this revolution that could have the greatest impact on healthcare delivery. The prefix 'tele' derives from the Greek 'at a distance', and therefore, more simply telemedicine is medicine at a distance. The information infrastructure of modern healthcare is based on digital information management. While the recent advances in information and communication technologies provide new means to access, handle and move medical information, they also compromise their security due to their ease of manipulation and replication. All patients records, electronic or not, linked to medical secrecy, must be confidential. The

digital handling of the EPR (Electronic Patient Record) on a network requires a systematic content validation that is aimed at quality control: actuality (precise interest of the information at a given instant) and reliability (authentication of the origin and integrity).

Attacks on security are best characterised by viewing the function of the computer system as a provision of information. In general, normal communication is represented as a flow of information from source to destination.

There are four categories of attacks:

- Interruption: An attack on availability. Information is destroyed or becomes unavailable or unusable.
- Interception: An attack on confidentiality. An unauthorised party gains access to information.
- Modification: An attack on integrity. An unauthorised party not only gains access to, but also tampers with information.
- Fabrication: An attack on authenticity. An unauthorised party inserts counterfeit objects into the system.

These attacks can be divided further into two categories, according to the nature of the attacks:

- Active Attacks: These attacks involve modification of the data stream or the creation of a false stream and can be subdivided into four categories:
  1. Masquerade: One entity pretends to be a different entity.
  2. Replay: The passive capture of a data unit and its subsequent retransmission to produce an unauthorised effect.
  3. Modification of messages: Some portion of a legitimate message is altered, or messages are delayed or recorded to produce an unauthorised effect.
  4. Denial of service: One prevents or inhibits the normal use or management of communications facilities.

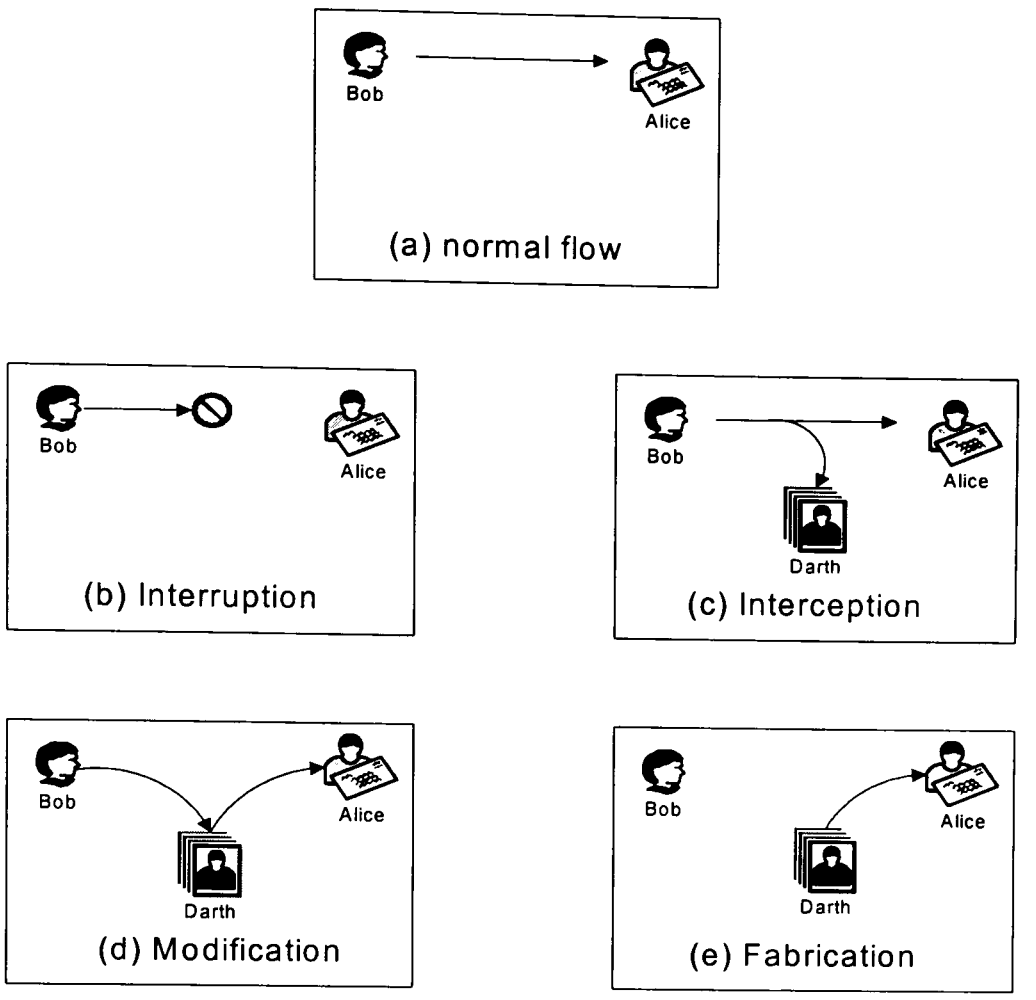


Figure 1.1 Security attacks

- Passive Attacks: These attacks involve eavesdropping on, or monitoring of, transmission and can be subdivided into two categories:
  1. Release of message contents: An unauthorised party obtains information that is being transmitted.
  2. Traffic analysis: An unauthorised party obtains information useful in guessing the nature of communication by observing the pattern of masked message transmissions.

This research will deal with modification and fabrication attacks on medical images. We will look at how to authenticate medical images using watermarking.

1.2 Image Authentication and Motivation

Image authentication can assure receivers that the received image is from the authorized source and that the image content is identical to the one sent. It is becoming easier and easier to tamper with digital image in ways that are difficult to detect. For example, Figure 1.2 shows two nearly identical images using readily available software (e.g.

Adobe Photoshop). The cyst was removed from the image by using the healing brush tool. It is difficult if not impossible to tell which picture is the original and which has been tampered with. If this image were a critical piece of evidence in a legal case or police investigation, this form of tampering might pose a serious problem.

The problem of authenticating messages has been studied in cryptography (Stinson 1995). Specifically, we are interested in methods for answering the following questions:

i. Has the image been altered in any way what so ever?
ii. What parts of the image have been altered?
iii. Can an altered image be restored?

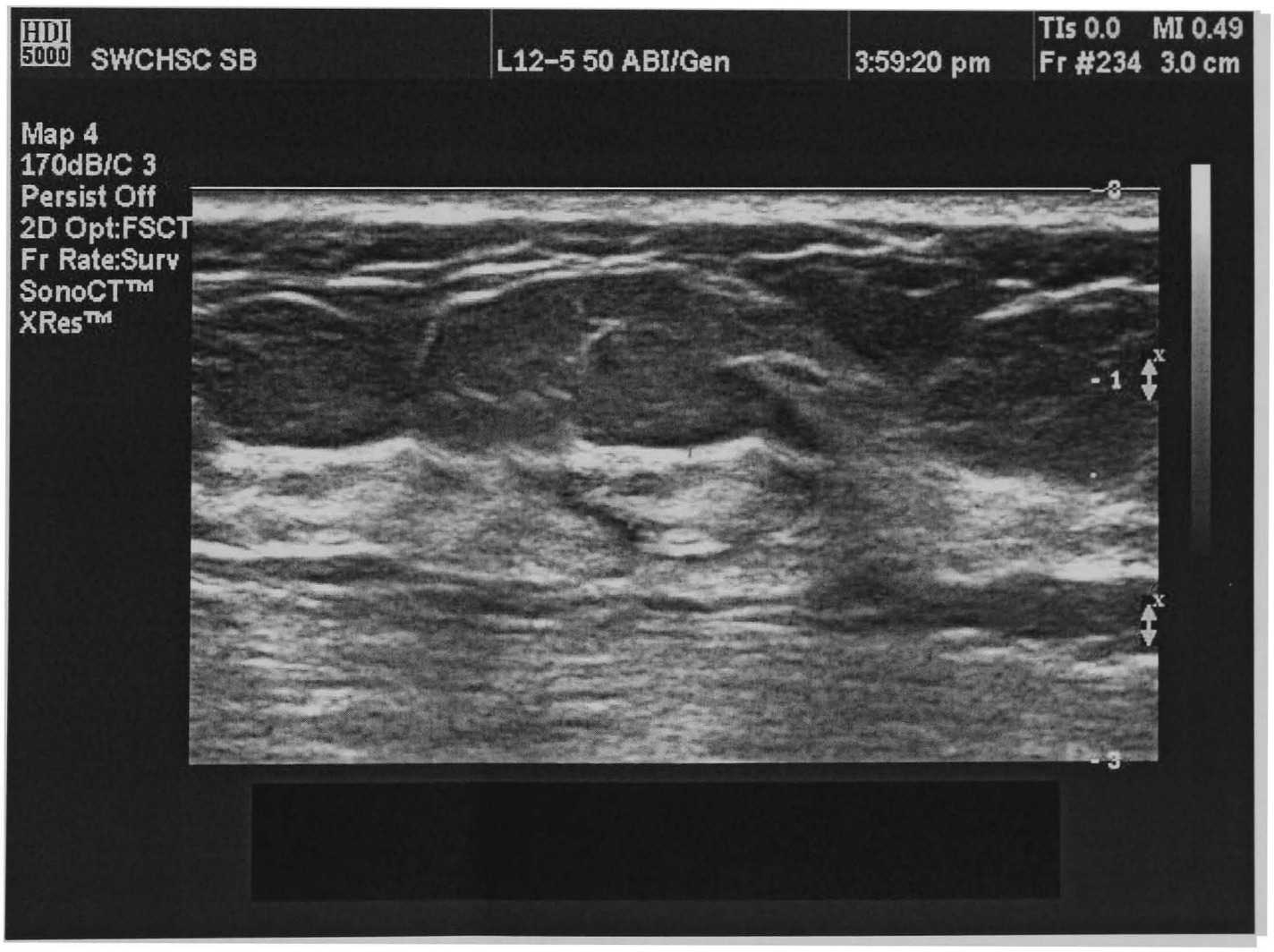
To implement such methods for medical images, the following questions are relevant:

iv. Do medical images have the same properties as other images?
v. What are the requirements needed to make watermarking suitable for medical images?

Many non-watermarking methods exist for answering these questions. Two common cryptographic approaches are the creation of a hash function and digital signature. The classical scenario for image authentication can be described as follows (Stallings 2003): A sender S wants to transmit a digital image I to a receiver R. When the image I^r is eventually delivered to R, by means of a network facility or any other media capable of storing digital data, an effective authentication scheme must ensure with high probability that:

- The Image I^r received by R is exactly the same as I the sender S has sent. (Integrity verification)
- The receiver R can verify the alleged source of the image I^r, where R can determine if S has actually sent I^r, or if a pirate has forged it. (Alleged Source Verification)

(a) An ultrasound image of a cystic breast tissue



(b) An ultrasound image of a normal breast tissue

Figure 1.2 Ease of modifying images

- R can demonstrate that I^r was actually sent by S, and S cannot deny having sent I^r. (non-repudiation Property).

Image authentication techniques are usually based on two kinds of tools: digital signature and watermarking. A digital signature is non-repudiation, encrypted version of the message digest extracted from the data. It is usually stored as a separate file, which can be attached to the data to prove integrity and originality. Watermarking techniques consider the image as a communication channel. The embedded watermark, usually imperceptible, may contain either a specific producer ID or some content-related codes that are used for authentication.

Digital watermarking (Cox et al. 2002, Langelaar et al. 2000) offers a promising alternative to digital signatures in image authentication applications. The use of watermarks instead of digital signatures typically records additional functionality by exploiting inherent properties of image content. The main advantage of digital watermarking is that the authentication information is directly embedded into the image data. As a result, the authentication information survives even when the host image undergoes format conversions. In contrast, a digital signature appended in the header of an image file may be easily stripped off, for instance, when the file is opened and saved in a different format. The digital watermark's capability for isolating manipulated image regions is another advantage. This functionality is known as the tamper localisation property. It is worth mentioning that both digital signatures and authentication watermarks are useful only for establishing the source of the image and detecting manipulations occurring after the signature/watermark has been inserted. Neither technique by itself is capable of certifying that a signal represents an original unaltered scene, unless supported by additional mechanisms (Friedman 1993). In this respect, digital watermarking differs from forensic image analysis (Federation Bureau of Investigation 2000).

Most watermarking techniques modify, and hence distort, the host signal in order to insert authentication information. In many applications, loss of image fidelity is not prohibitive as long as the original and modified images are perceptually equivalent. On

the other hand, in medical, military and legal imaging applications, where the need for authentication is often paramount, there are typically stringent constraints on data fidelity that prohibit any permanent signal distortion in the watermarking process. For instance, artifacts in a patient's diagnostic image may cause errors in diagnosis and treatment with possible life-threatening consequences. Likewise, in military applications, satellite and aerial photographs are often enlarged, enhanced or further processed by computer vision algorithms. Unless the loss of fidelity is either carefully limited or eliminated altogether, the corresponding artifacts may be amplified by the post-processing operations. In these applications, the permanent loss of signal fidelity due to digital watermarking can be remedied by lossless data embedding (also referred as reversible, invertible or distortion-free data embedding) techniques. These techniques, like their lossy counterparts, insert information bits by modifying the host signal, thus induce an embedding distortion. Nevertheless, they also enable the removal of such distortions and the exact/lossless restoration of the original host signal after extraction of the embedded information. Lossless data embedding methods can be employed for digital image authentication. Lossless (reversible) authentication watermarks provide a complete framework; the authentication property of the watermark protects the integrity of the image, whereas the quality is preserved by the reversibility of the watermarking process.

1.3 Current Security Methods for Medical Images

- Data Encryption

Encryption is the most useful approach to assure data security during its transmission through public communication networks. Image data scrambled by a sender cannot be understood by anyone other than an intended party and assures data security during transmission, but not before or after.

  - Virtual Private Network (VPN) is the most common application of data encryption techniques to ensure data security during its transmission through public communication networks.

- DICOM Security or DICOM standard part 15 has been released to provide a standardised method (selection of security standards, encryption algorithms and parameters) for secure medical image communication but has yet to be implemented by the industrial and medical community.

- Data Embedding

Data embedding can be a form of steganography that conceals patient information and the digital signature in the image so that the visual quality of the image is not perceptually affected. It provides a permanent assurance of image data security no matter when and how the image has been manipulated. But there is no standard embedding method and it is also difficult to implement for a variety of medical image modalities. Watermarking researchers in the medical field have also incorporated hashing to produce image digests and use them as watermarks (Cao et al. 2003, Guo and Zhuang 2003, Zhou et al. 2001).

1.4 Watermarking and Steganography

Watermarking, that is the technique of placing and transmitting a small amount of data imperceptibly in the host or cover data has many applications including broadcast monitoring, owner identification, proof of ownership, and content authentication. Paper watermarks are used regularly as an authentication (anti-counterfeiting) measure in valuable documents, such as bank notes, cheques and visa stamps. For instance, the authenticity of a bank note is confirmed by the existence of a visible watermark pattern when the note is held to the light. Paper watermarks are designed to be i) easily detectable, ii) hard to reproduce, and iii) invisible or unobtrusive in normal use of the document. Digital watermarks inherit many of the paper watermarks features and properties: they are digital patterns superimposed on digital signals; the patterns should be easily detectable, yet be very hard to reproduce without specific knowledge (cryptographic keys); the watermark should be invisible or unobtrusive during normal use of the digital signal.

However, steganography or data hiding has a long history and the use of paper watermarks for copy protection can be traced back to the thirteenth century (Murray 1996). The earliest forms of information hiding can actually be considered to be highly crude forms of private-key cryptography; the "key" in this case being the knowledge of the method being employed (security through obscurity). Steganography books are filled with examples of such methods used throughout history. Greek messengers had messages tattooed into their shaved head, concealing the message when their hair finally grew back. Wax tables were scraped down to bare wood where a message was scratched. Once the tables were re-waxed, the hidden message was secure (Petitcolas 2000). Over time these primitive cryptographic techniques improved, increasing speed, capacity and security of the transmitted message.

Today, crypto-graphical techniques have reached a level of sophistication such that properly encrypted communications can be assumed secure well beyond the useful life of the information transmitted. In fact, it is projected that the most powerful algorithms using multi kilobit key lengths could not be comprised through brute force, even if all the computing power worldwide for the next 20 years was focused on the attack. Of course the possibility exists that vulnerabilities could be found, or computing power breakthroughs could occur, but for most users in most applications, current cryptographic techniques are generally sufficient.

Why then pursue the field of information hiding? Several good reasons exist, the first being that "security through obscurity" is not necessarily a bad thing, provided that it is not the only security mechanism employed. Steganography for instance allows us to hide encrypted messages in mediums less likely to attract attention. A garble of random characters being transmitted between two users may tip off a watchful third party that sensitive information is being transmitted; whereas baby pictures with some additional noise present may not. The underlying information in the pictures is still encrypted, but attracts far less attention by being distributed in the picture than it would otherwise.

Nowadays, there exist watermarking methods for virtually every kind of digital media: text documents (Su et al. 1998, Brassil et al. 1999), images (Tsai et al. 2004, Zhang et al. 2003, Paquet et al. 2003), video (Sun and Chang 2003, Okada et al. 2002), audio (Li

and Xue 2003, Yan et al. 2004), even for 3D polygonal models (Kwon et al. 2003, Benedens and Busch 2000), maps (Barni et al. 2001) and computer programs (Monden et al. 2000). Interestingly, watermarking technology is not limited to digital media, but is also applicable to chemical data like protein structures, for example (Eggers et al. 2001).

1.5 Research Objectives

There are three research objectives:

1. To investigate methods for authentication watermarking.
2. To develop techniques for authentication appropriate for a chosen medical image modality.
3. To investigate and evaluate any such technique on the chosen medical image modality.

1.6 Research Strategy and Method

The sources of information for the present work came from three different subject areas. Firstly, information security in general; secondly, hiding information, known as steganography; and thirdly medical imaging.

The research concentrates on authenticity and integrity of medical images, and investigates current techniques of authentication with an emphasis on those most suitable for medical images. A few issues need to be clarified before choosing tools and techniques for this research. The first issue to consider is whether complete authentication or content authentication is required as an entity.

Complete authentication refers to techniques that consider the image and do not allow any manipulations or transformation (Wu and Liu 1998, Yeung and Mintzer 1997). Many existing message authentication techniques can be applied directly. For example, a digital signature might be placed in the LSB of the uncompressed data or the header of the compressed data. Manipulations will be detected if the hash value of the altered message does not match the digital signature. In practice, fragile watermarks or traditional digital signatures may be used for complete authentication.

Content authentication refers to a different objective that is applicable to multimedia data. The meaning of multimedia data is based on its content instead rather than specific bit content. In some applications, manipulations on the bit streams without changing the meaning of content are considered as acceptable. Compression is an example. Digital Imaging and Communication in Medicine (DICOM) standard has included JPEG (lossy and lossless), JPEG-LS and RLE (known as TIFF) compressions in their standard. JPEG2000 has also been considered in the report (National Electrical Manufacturers Association (NEMA) 2002).

The second issue is whether the watermarks are reversible or permanent. Ideally the medical image should be unaltered through the process of watermarking (Macq and Dewey 1999, Giakoumaki et al. 2003, Yang and Bao 2003) and the watermarking should be reversible so that the original image can be restored. However, it may be argued that if the change is imperceptible and has no impact on diagnosis then it is acceptable and may be compared with compression, which is accepted.

The third issue is if we decide to have watermarks for content authentication, whether compression should be distinguished from other manipulations. Previous watermarks are either too fragile for compression or too flexible to detect malicious manipulations. The performance of an authenticator should be simultaneously evaluated by two parameters: the probability of false alarm and the probability of missing manipulations. Fragile watermarks, which have a low probability of missing manipulations, usually fail to survive compression so that their probability of false alarm is very high. Previous researchers have attempted to modify the fragile watermark to make it robust to compression (Wolfgang and Delp 1996, Zhu et al. 1996). However, such modifications then failed to distinguish both compression and tampering. In general, watermarks made robust to most manipulations are usually then too robust to detect malicious manipulations.

Watermarking capacity is determined by invisibility and robustness requirements. The relationship between capacity, invisibility and robustness is shown in Figure 1.3.

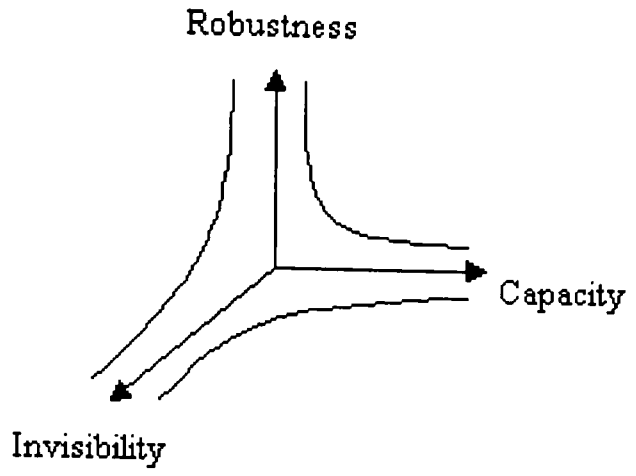


Figure 1.3 Watermarking properties

It is seen that if one parameter is fixed, then the other two parameters are inversely proportional. For instance, a specific application may determine how many message bits are needed, copyright protection may need to embed about 10 bytes and authentication may need from 100-1000 bits for a 265 X 256 image. After the amount to embed is decided, there exists a trade-off between visual quality and robustness. Robustness refers to the extraction of embedded bits with a probability of error equal to or approaching zero. Visual quality represents the quality of watermarked image. In general, if we want to make our message bits more robust against attack, then a longer codeword will be necessary to provide better error resistance. However, degradation in visual quality can be expected.

1.7 Dissertation Outline

This thesis is divided into 7 chapters and organised as follows:

Chapter 1: This chapter introduces the problem area and outlines approaches to be explored. In this chapter watermarking is introduced as the technique used in the research.

Chapter 2: This chapter presents the area of message authentication. The digital signature is discussed as a possible watermark. A review of current image authentication is presented and various embedding techniques are discussed.

Chapter 3: This chapter presents medical image watermarking. Issues and properties of medical image watermarking are discussed. Some approaches in

watermarking medical images and the issues of tamper localisation and restoration are presented.

Chapter 4: This chapter proposes two strict authentication watermarking techniques SAW and SAW-JPEG. SAW embeds the digital signature of the medical image in the region of non-interest. SAW-JPEG is an enhanced SAW and is made to be robust to some degree of JPEG compression.

Chapter 5: This chapter proposes another authentication watermarking with tamper detection and recovery AW-TDR.

Chapter 6: This chapter discusses the results obtained from Chapter 4 and Chapter 5 and evaluates the proposed techniques.

Chapter 7: This chapter presents a summary of the thesis and conclusions to the work.

Chapter 2

Literature Review

2.1 Introduction

This chapter introduces the area of image authentication, the techniques available and introduces the area of image watermarking. The chapter is structured as follows:

- Section 2.2 introduces hash functions and how it is used to secure message transmission and describes the method. The use of private key and public key is also discussed.
- Section 2.3 describes a digital signature algorithm and the purpose of having one person who can produce a signature that can checked by everybody else by using private and public key. This section also describes how a digital signature algorithm is used for image authentication.
- Section 2.4 presents an image authentication scheme using content-based, feature-based, edge-based, mean-based and relation-based methods.
- Section 2.5 gives definition of fragile and semi-fragile watermarking and provides a review of methods available.
- Section 2.6 lists the requirements for watermarking-based authentication system.
- Section 2.7 describes the main components for a watermarking system.

- Section 2.8 will show some of the most frequent attacks that an image authentication system has to overcome
- Section 2.9 presents embedding techniques for watermarking.

2.2 Hashed Message Authentication Code (HMAC)

A hash function such as MD5 (Rivest April 1992) and SHA-1 (National Institute of Standards and Technology 1995) produces a one-way message digest, a fingerprint of a file, message, or other block of data. The hash based message authentication code (HMAC) encrypts the hash value of the message with a secret key shared by the sender and receiver. This technique assumes that two communicating parties, A and B share the same secret key K_{AB}. When A has a message M to send to B, it calculates the message authentication code as a function of the message and the key: $MAC_M = H(K_{AB}, M)$.

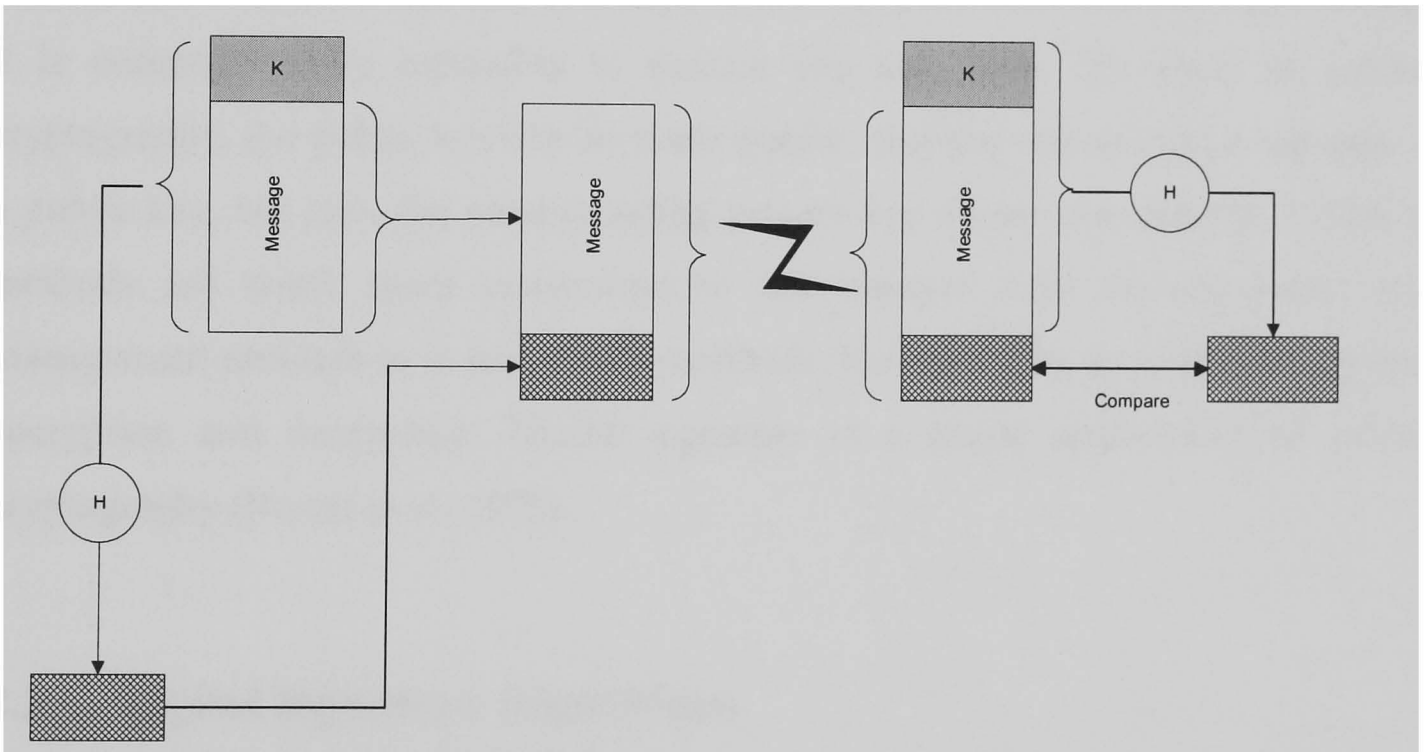


Figure 2.1 HMAC (Adapted from Network Security Essentials page 58)

The message and the MAC code are transmitted to the intended recipient. The recipient performs the same calculation on the received message, using the same secret key, to generate a new message authentication code. The received code is compared to the calculated code. If we assume that only the receiver and the sender know the identity of the secret key, and if the received code matches the calculated code, then

1. The receiver is assured that the message has not been altered. If an attacker alters the message, the received code will not match the calculated code.
2. The receiver is assured that the message is from the alleged sender as no one else could prepare a message with a proper code.

Modern cryptography can use either private-key or the public-key key method (Garfinkel and Spafford 1996). Private-key cryptography (symmetric cryptography) uses the same key for data encryption and decryption. It requires both the sender and the receiver to agree on a key before they can exchange message securely. Although computation speed for obtaining the private-key is acceptable, the management of the keys is difficult.

Public-key cryptography (asymmetric cryptography) uses two different keys (a key pair) for encryption and decryption. The keys in the key pair are mathematically related, but it is computationally infeasible to deduce one key from the other. In public-key cryptography, the public key can be made public. Anyone can encrypt a message using a public key, but only the corresponding private-key owner can decrypt it. Public-key methods are much more convenient to use because they do not share the key management problem as in private-key methods. However they require a longer time for encryption and decryption. Digital signature is a major application of public-key cryptography (Rivest et al. 1978).

2.3 Digital Signature Algorithms

The basic idea of a digital signature is that a signature on a message can be created by only one person, but checked by anyone. It can thus perform the sort of function in the electronic world that ordinary signatures do in the world of paper. The asymmetric encryption algorithms published in the late 1970s, such as RSA, in conjunction with the secure hash functions, are digital signature algorithms, which allow the sender to associate its unforgeable imprint with the digital image, so that the receiver can check its integrity and its source. Non-repudiation is also guaranteed.

The asymmetric encryption involves the use of two separate keys: a public key made public for others to decrypt a received message, and a private key known only to its owner to encrypt the original. When A has a message M to send to B, it calculates the digital signature sig M as a function of the hashed message H(M) and the private key Kprivate: sigM = F(Kprivate, H(M)). The message plus digital signature are transmitted to the intended receiver. The receiver performs the same hash calculation on the received message to generate a hashed message. The receiver also decrypts the received signature sigM, using public key Kpublic, to get the received hashed message. The received hashed message is compared to the hashed message. If we assume that only the sender knows the identity of the secret key, and if the received hashed message is identical to the new hashed message, then

- The receiver is assured that the message has not been altered. If an attacker alters the message but does not alter the code, then the receiver's calculation of the hashed message will differ from the new hashed message. Because the attacker is assumed not to know the private key, the attacker cannot alter the code to correspond to the alterations in the message.
- The receiver is assured that the message is from the alleged sender. Because no one knows the private key, no one else could prepare a message with a proper digital signature.

Image authentication is projected as a procedure of guaranteeing that the image content has not been altered, or at least that the visual (or semantic) characteristics of the image are maintained after incidental manipulations such as JPEG compression. In other words, one of the objectives of image authentication is to verify the integrity of the image. For many applications such as medical archiving, news reporting and political events, the capability of detecting manipulations of digital images is often required.

To address both the integrity and legitimacy issues, a wide variety of techniques have been proposed for image authentication. Depending on the ways chosen to convey the authentication data, these techniques can be divided into two categories: labelling-based techniques (e.g., the method proposed by Friedman 1993) and watermarking-based

techniques (e.g., method proposed by Walton 1995). The main difference between these two categories of techniques is that labelling-based techniques create the authentication data in a separate file while watermarking-based authentication can be accomplished without the overhead of a separate file.

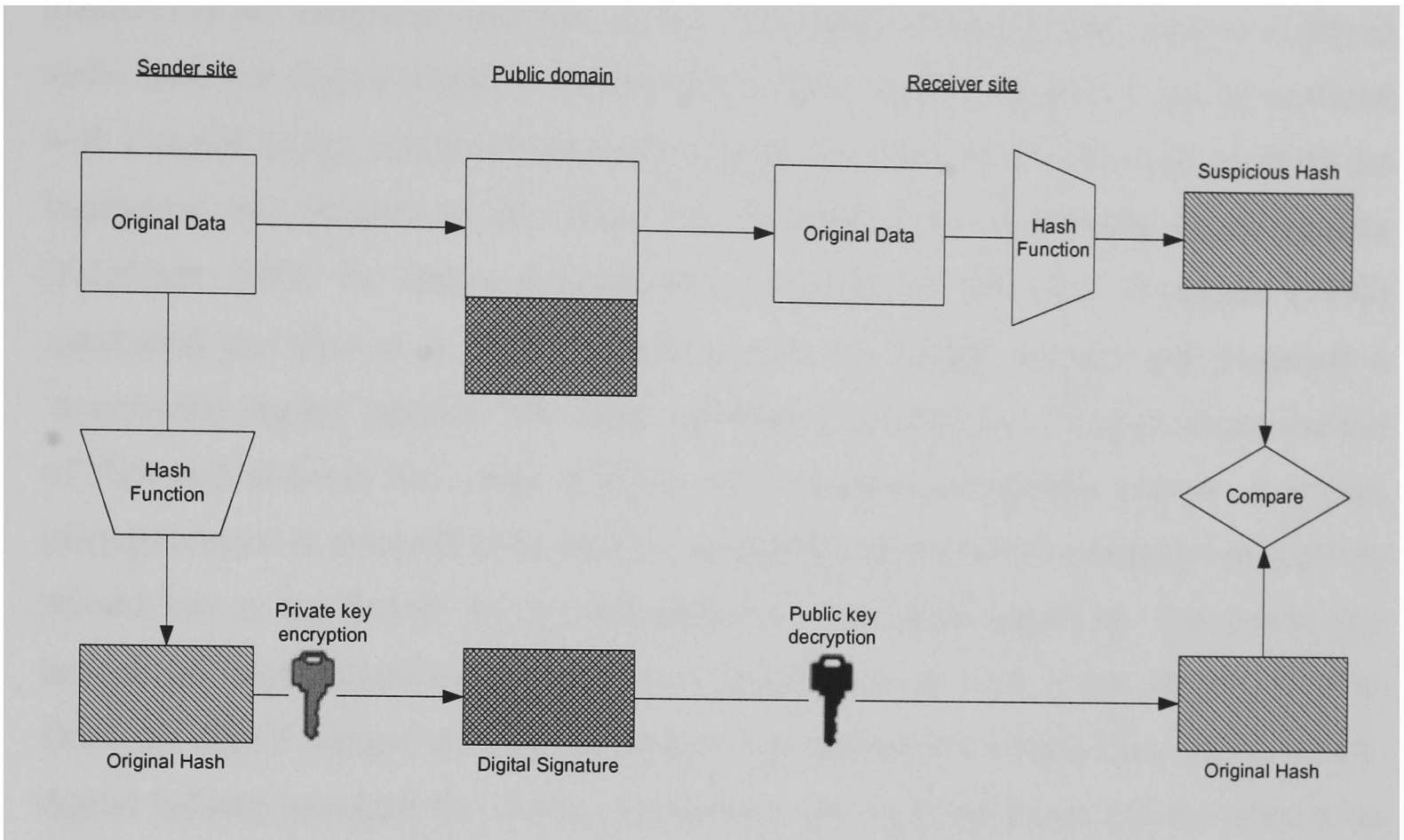


Figure 2.2. Basic model of a digital signature

The digital signature-based image authentication is based on the concept of a digital signature, which is derived from a cryptographic technique called public key cryptosystem (Rivest et al. 1978, Diffie and Hellman 1976). Figure 2.2 shows the basic model of a digital signature. The sender first uses a hash function, such as MD5 (Rivest 1992), to hash the content of the original data to a small file called digest. MD5 was the most widely used secure hash algorithm until the last few years that the security of a 128-bit hash code has become questionable (Dobbertin 1996) and in summer 2004 was broken by Chinese researchers (Wang et al. 2004, Hawkes et al. 2005). Then the digest is encrypted with the sender's private key. The encrypted digest can form a unique 'signature' because only the sender has the knowledge of the private key. The signature is then sent to the receiver along with the original information. The receiver can use the sender's public key to decrypt the signature, and obtain the original digest. The received information can also be hashed by using the same hash function at the sender's side. If

the decrypted digest matches the newly created digest, the legitimacy and the integrity of the message are therefore authenticated.

There are two points worth noting in the process of a digital signature. First, the plaintext is not limited to text file. In fact, any types of digital data, such as digitised audio data and digital image. Therefore the original data in Figure 2.2 can be replaced with a digital image, and the process of a digital signature can then be used to verify the legitimacy and integrity of the image. The concept of the trustworthy digital camera (Friedman 1993) for image authentication is based on this idea. Friedman (1993) associated the idea of a digital signature with the digital camera and proposed a 'trustworthy digital camera'. The proposed digital camera uses a digital sensor instead of film and delivers the image directly in a computer-compatible format. A secure microprocessor is assumed to be built in the digital camera and is programmed with the private key at the factory for the encryption of the digital signature. The public key needed for later authentication appears on the camera as well as the image's border. Once the digital camera captures the image, it produces two output files. One is an all-digital industry-standard file format representing the captured image and the other is an encrypted digital signature generated by applying the camera's unique private key to a hash of the captured image file. The digital image file and the digital signature can later be distributed freely and safely.

Image authentication is accomplished by calculating the hash of the received image, and by using the public key to decode the digital signature to reveal the original hash; the two hash values are compared. If these two hash values match, the image is considered to be authentic. If these two hash values are different, the integrity of this image is questionable. It should be noted that the hash algorithms such as SHA-256 are sensitive to single bit changes. This is strict authentication. However in the process of lossy compression, although the image is essentially retained, individual pixel values may be changed. Strict authentication will determine the image is no longer authentic and does not provide a useful check. A different check is required.

2.4 Other Image Authentication Schemes

- **Content-based authentication**

Image manipulation such as lossy compression, changes individual pixel values and so strict authentication (hash value calculated from all bit values in the image) will fail. In these cases a method must be sought to determine features in the image that will be invariant through the compression-decompression process. Edge information, DCT coefficients, colour, and intensity histograms are regarded as potential invariant features.

In Schneider and Chang's (Schneider and Chang 1996) method, the intensity histogram is employed as the invariant feature in the implementation of the content-based image authentication scheme. To be effective, the image is divided into blocks of variable sizes and the intensity histogram of each block is computed separately and is used as the authentication code. To tolerate incidental modifications, the Euclidean distance between intensity histograms was used as a measure of the content of the image. They pointed out that using a reduced distance function could increase the maximum permissible compression ratio up to 14:1 if the block average intensity is used for detecting image content manipulation.

- **Feature-based method**

Bhattacharjee and Kutter (1998) proposed another algorithm to extract a smaller size feature of an image. Their feature extraction algorithm is based on the so-called scale interaction model. Instead of using Gabor wavelets, they adopted Mexican-Hat wavelets as the filter for detecting the feature points. The algorithm for detecting feature points is depicted as follows.

- Define the feature-detection function, $P_{ij}(.)$ as:

$$\mathrm{P}_{ij}(\vec{x}) = \left| M_i(\vec{x}) - \gamma . M_j(\vec{x}) \right| \tag{2.1}$$

where $M_i(\vec{x})$ and $M_j(\vec{x})$ represent the responses of Mexican-Hat wavelets at the image location $\vec{x}$ for scales I and j respectively. For the image A, the wavelet response $M_i(\vec{x})$ is given by:

$$M_i(\vec{x}) = \langle (2^{-i}\psi(2^{-1}\cdot\vec{x})); A\rangle \tag{2.2}$$

where $\langle .;. \rangle$ denotes the convolution of its operands. The normalising constant γ is given by $\gamma=2^{-(i-j)}$, the operator $|\cdot|$ returns the absolute value of its parameter, and the $\psi(\vec{x})$ represents the response of the Mexican-Hat mother wavelet, and is defined as:

$$\psi(\vec{x}) = (2-|\vec{x}|^2)\exp(-\frac{x^2}{2}) \tag{2.3}$$

- Determine points of local maximum of $P_{ij}(.)$. These points correspond to the set of potential feature points
- Accept a point of local maximum in $P_{ij}(.)$ as a feature-point if the variance of the image pixels in the neighbourhood of the point is higher than a threshold. This criterion eliminates a suspicious local maximum in featureless regions of the image.

The column positions and row positions of the resulting feature points are concatenated to form a string of digits, and then encrypted to generate the image signature. In order to determine whether an image A is authentic with another known image B, the feature set S_A of A is computed. The feature set S_A is then compared with the feature set S_B of B that is decrypted from the signature of B. The following rules are adopted to authenticate the image A.

- Verify that each feature location is present both in S_B and in S_A.
- Verify that no feature location is present in S_A but absent in S_B.
- Two feature points with coordinates $\vec{x}$ and $\vec{y}$ are said to match if:

$|\vec{x}-\vec{y}|<2$

- **Edge-based method**

The edges in an image are the boundaries or contours where significant changes occur in some physical aspects of an image. Edges are a strong content feature for an image. However, coding edge values and positions can carry a large overhead. One way to resolve this problem is to use a binary map to represent only the edges. For example, Li et al (2003) used a binary map to encode the edges of an image in their image authentication scheme. However it is known that edges will be modified when high compression ratios are used. Consequently, the success of using edges as the authentication code is greatly dependent on the capacity of the authentication system to discriminate the differences the edges produced by content-preserving manipulations from those content-changing manipulations.

- **Mean-based method**

The local mean is a simple and practical image feature to represent the content of an image. Lou and Liu (2000) proposed such an algorithm to generate a mean-based feature code. The original image is divided into non-overlapping blocks and the mean of each block calculated and quantized according to a predefined parameter. The calculated results are then encoded to form the authentication code. In the verification process the quantized means of each block of the received image is calculated. The quantized code is compared with the original quantized code on a block-by-block basis. A binary error map is produced as an output with '1' denoting match and '0' denoting mismatch. The verifier can thus tell the possibly tampered blocks by inspecting the error map. There is also some capability to restore the untampered version, which may be attractive in some real time image application such as video.

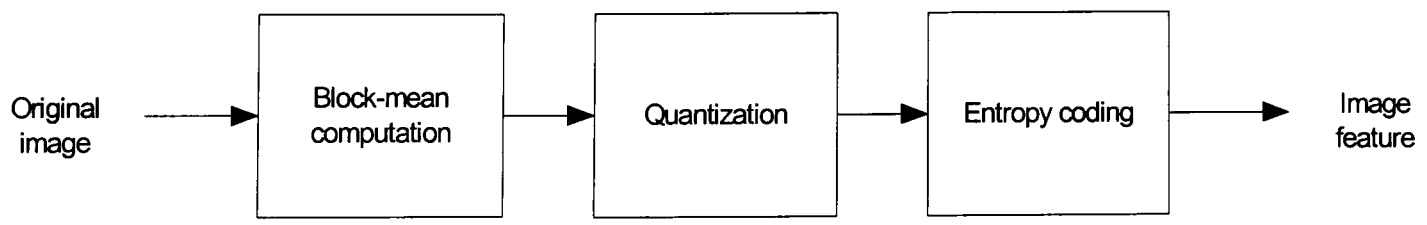


Figure 2.3 Mean based feature code

- **Relation-based method**

Unlike the methods introduced before, relation-based methods divide the original image into non-overlapping blocks, and use the relation between blocks as the feature code. The method proposed by Lin and Chang (Lin and Chang 2001) is called SARI. The feature code in SARI is generated to survive the JPEG compression. To serve this purpose, the process of the feature code generation starts with dividing the original image into 8x8 non-overlapping blocks. Each block is then DCT transformed. The transformed DCT blocks are further grouped into two non-overlapping sets. There are equal numbers of DCT blocks in each set. A secret key-dependant mapping function then maps one-to-one each DCT block in one set into another DCT block in another set, and generates N/2 DCT block pairs. For each block pair, a number of DCT coefficients are then selected and compared. Comparing the corresponding coefficients of the paired blocks then generates the feature code.

The feature code of the received image is extracted using the same secret key and is compared with the original feature code. If neither block in each block pair has been maliciously manipulated, the relation between the selected coefficients is maintained. Otherwise, the relation between the selected coefficients may be changed.

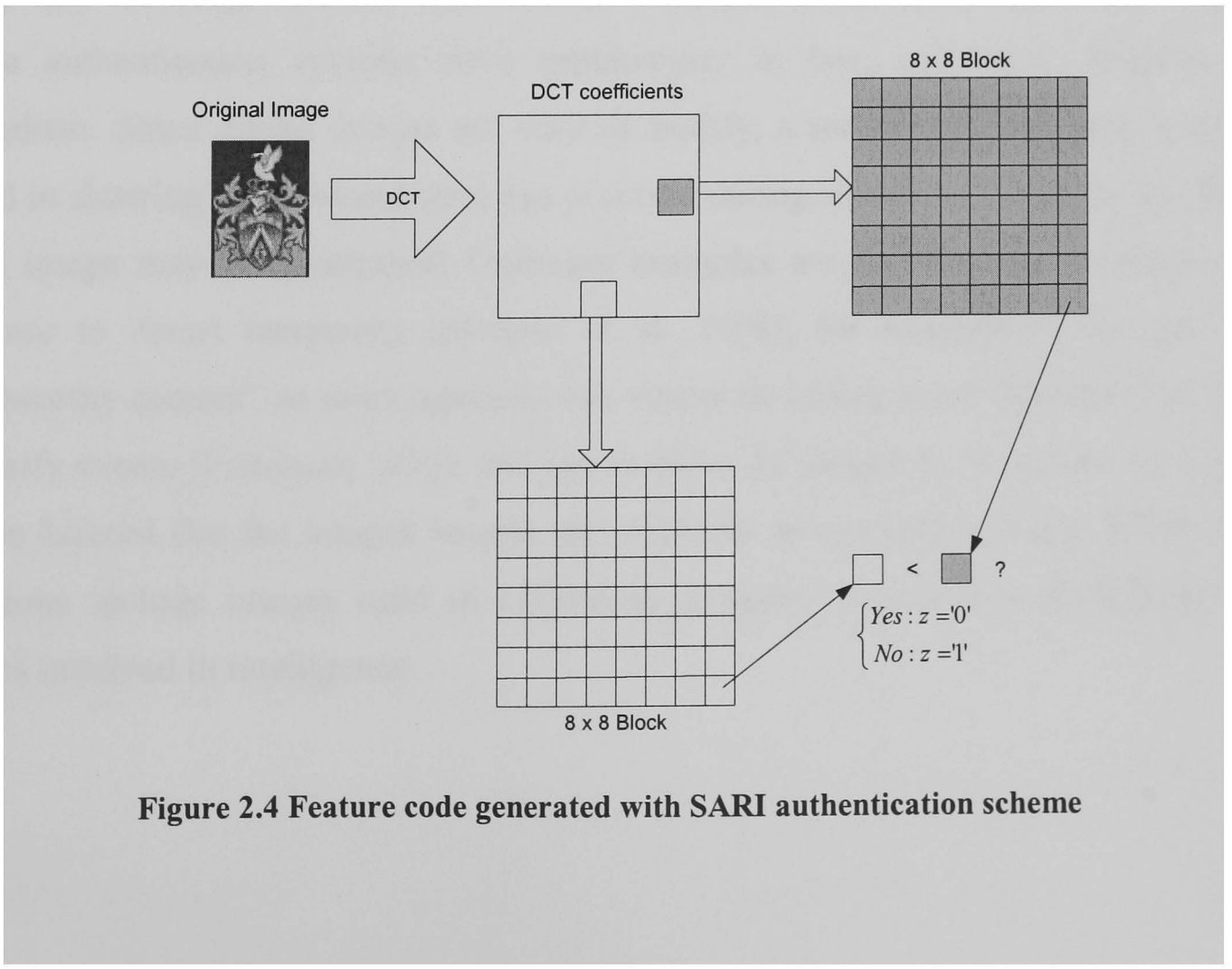


Figure 2.4 Feature code generated with SARI authentication scheme

2.5 Fragile and Semi-fragile Watermarking

A fragile watermarking is one that is likely to be destroyed and become undetectable after the image has been modified in any way. Watermarking researchers have considered fragility as undesirable and therefore seek to design robust watermarks that can survive many forms of distortion. However, fragility can be an advantage for authentication purposes. If a fragile watermark is detected correctly in an image, we can say that the image has not been altered or tampered with since the watermark has been embedded. A fragile watermark is a mark that is readily altered or destroyed when the host image is modified through linear or nonlinear transformation (Yeung and Mintzer 1998). In the case of authenticity, a fragile watermark has to prove that the image has been modified and is no longer authentic. However, for copy protection applications, the watermark has to be robust and be able to withstand different types of alterations such as lossy compression and filtering.

Fragile watermarks are not suited for enforcing copyright ownership of digital images. An attacker would attempt to destroy the embedded watermark and fragile watermarks are by definition easily destroyed. The sensitivity of fragile watermarks to modification leads to their use in image authentication. That is, it may be of interest for parties to verify that an image has not been edited, damaged, or altered since it was marked. Image authentication systems have applicability in law, commerce, defense, and journalism. Since digital images are easy to modify, a secure authentication system is useful in showing that no tampering has occurred during situations where the credibility of an image may be questioned. Common examples are the marking of images in a database to detect tampering (Mintzer et al. 1998), for example in the use of a "trustworthy camera" so news agencies can ensure an image is not fabricated or edited to falsify events (Friedman 1993), and the marking of images in commerce so a buyer can be assured that the images bought are authentic upon receipt (Wong 1998). Other situations include images used in courtroom evidence, journalistic photography, or images involved in intelligence.

As mentioned previously, one of the methods used to verify the authenticity of a digital work is the use of a signature system (Stallings 2003). In a signature system, a digest of the data to be authenticated is obtained by the use of cryptographic hash functions (Stallings 2003, Wolfgang and Delp 1999). The digest is then cryptographically signed to produce the signature that is bound to the original data. Later, a recipient verifies the signature by examining the digest of the (possibly modified) data and using a verification algorithm to determine if the data is authentic.

While the purpose of fragile watermarking and digital signature systems are similar, watermarking systems offer several advantages compared to signature systems (Memon et al. 1999) at the expense of requiring some modification (watermark insertion) of the image data. As a watermark is embedded directly in the image data, no additional information is necessary for authenticity verification. This is unlike digital signatures since the signature itself must be bound to the transmitted data. Therefore the critical information needed in the authenticity testing process is discreetly hidden and more difficult to remove than a digital signature. Also, digital signature systems view an image as an arbitrary bit stream and do not exploit its unique structure. Therefore a signature system may be able to detect that an image has been modified but cannot characterise the alterations. Many watermarking systems can determine which areas of a marked image have been altered and which areas have not, as well as estimate the nature of the alterations.

2.5.1. Examples of Fragile Marking Systems

Early fragile watermarking systems embedded the mark directly in the spatial domain of an image, such as techniques described in Walton (1995) and van Schyndel et al. (1994). These techniques embed the mark in the least significant bit plane for perceptual transparency. Their significant disadvantages include the ease of bypassing the security they provide (Yeung and Mintzer 1998, Fridrich 1998) and the inability to lossy compress the image without damaging the mark.

Any processing of the image, such as compression will result in changes to the LSB. If a watermark is to be embedded in the LSB plane of the image, we imply that the image has not undergone any such process. Fragile watermarking algorithms are concerned with complete integrity verification. The slightest modification of the host image will alter or destroy the fragile watermark. Yeung and Mintzer (1998) embeds a binary logo of the same size as the host image by means of a key dependent look-up table (LUT) that maps every possible pixel luminance value to either 0 or 1. The watermark is inserted by adjusting the least significant bit (LSB) value of each image pixel in the spatial domain to match its corresponding LUT value. At the receiving side, the LUT can be reconstructed due to the knowledge of the secret key. The integrity verification can be performed either by simple visual inspection of the extracted watermark, or by automated comparison with the original one. This watermarking scheme is very sensitive to any distortion in the image and is very vulnerable to a block analysis attack.

Fridrich and Baldoza (2000) improved the algorithm by using 64x64 block cipher instead of LUT, and the watermark is embedded in a 32x32 block. The improved scheme can be used against the block analysis attack.

A further fragile marking technique described by Wong (1999), obtains a digest using a hash function. The image, its dimensions and marking key are hashed during embedding and used to modify the least-significant bit plane of the original image. This is done in such a way that when the correct detection side information and unaltered marked image are provided to the detector, a bi-level image chosen by the owner (such as a company logo or insignia) is observed. This technique has localisation properties and can identify regions of modified pixels within a marked image. However, Holliman and Memon (2000) soon presented a vector quantization (VQ) counterfeiting attack that can construct a counterfeit image from a VQ codebook generated from a set of watermarked images. To solve the problem of VQ counterfeiting attack, several enhanced algorithms were proposed (Holliman and Memon 2000, Fridrich et al. 2000, Wong and Memon 2000). Nonetheless, they either fail to effectively address the problem or sacrifice the tamper localisation accuracy of the original methods (Celik et al. 2002). Celik et al. (2002) then presented an algorithm based on Wong's scheme and

demonstrated that their algorithm can thwart the VQ codebook attack while sustaining the localisation property.

The technique of Yeung and Mintzer (1998), whose security is examined in (Memon et al. 1999), is also one where the correct detection information results in a bi-level image. However, the embedding technique is more extensive than inserting a binary value into the least-significant bit plane. The marking key is used to generate several pseudo-random look-up tables (one for each channel or colour component) that control how subsequent modification of the pixel data will occur. Then, after the insertion process is completed, a modified error diffusion process can be used to spread the effects of altering the pixels, making the mark more difficult to see. As discussed in (Memon et al. 1999), the security of the technique depends on the difficulty of inferring the look-up tables. The search space for the table entries can be drastically reduced if knowledge of the bi-level watermark image is known. A modification (position-dependent lookup tables) is proposed in (Memon et al. 1999) to dramatically increase the search space.

Various transformations, such as the discrete cosine transform (DCT) and wavelet transforms, are widely used for lossy image compression and much is known about how the transform coefficients may be altered (quantized) to minimize perceptual distortion (Wolfgang et al. 1999). There is also a great deal of interest in transform embedding for robust image marking systems to make embedded marks more resilient to attacks.

There are advantages for fragile watermarking systems to use the transform domain. Many fragile watermarking systems are adapted from lossy compression systems (such as JPEG), which have the benefit that the watermark can be embedded within the compressed representation. The properties of a transform can be used to characterise how an image has been damaged or altered. Also, applications may require a watermark to possess robustness to certain types of modification (such as brightness changes) yet be able to detect other modifications (e.g. local pixel replacement). Wu and Liu (1998) describe a technique based on a modified JPEG encoder. The watermark is inserted by changing the quantized DCT coefficients before entropy coding. A special lookup table of binary values (whose design is constrained to ensure mark invisibility) is used to

partition the space of all possible DCT coefficient values into two sets. The two sets are then used to modify the image coefficients to encode a bi-level image (such as a logo.) To reduce the blocking effects of altering coefficients, it is suggested that the DC coefficient and any coefficients with low energy be unmarked.

2.5.2. Examples of Semi-fragile watermarking

A semi-fragile watermark describes a watermark that is unaffected by legitimate distortions, but destroyed by illegitimate distortions. It provides the mechanism for implementing selective authentication. Semi-fragile watermark combines the properties of fragile and robust watermarks. Like a robust watermark, a semi-fragile watermark is capable of tolerating some degree of change to the watermarked image, such as the addition of quantization noise from lossy compression. And like a fragile watermark, the semi-fragile watermark is capable of localising regions of the image that have been tampered with and distinguish them from regions that are still authentic. Thus, a semi-fragile watermark can differentiate between localised tampering and information preserving, lossy transformations. Many fragile watermarking systems perform watermark embedding in the LSB plane and are unable to tolerate a single bit error in this bit. However, the quantization noise introduced by compression is likely to cause many least significant bits to change. Furthermore, recent fragile watermarking systems employ cryptographic hash functions that are not suitable in a semi-fragile framework. A hash function h(x) will produce completely different outputs h(x1) and h(x2) if the binary inputs are distinct but very similar. Even if some characteristic of the image that is expected to remain invariant during lossy compression were hashed, the output of the hash function would have to be embedded in a way that is resilient to errors.

Wolfgang and Delp (1996) extended van Schyndel's work to improve robustness and localisation in their VW2D technique. Adding a bipolar M-sequence in the spatial domain embeds the watermark. Detection is via a modified correlation detector. For localisation, a blocking structure is used during embedding and detection. This mark has been compared to other approaches using hash functions (Wolfgang, Delp 1999).

Fridrich (1998) proposes a similar technique. To prevent unauthorised removal or intentional watermark distortion, the author recommends making the mark dependent on the image in which it is embedded. The binary mark used corresponds to a pseudo-random signal generated from a secret key, the block number and the content of the block represented with an M-tuplet of bits. Each block is then watermarked using O'Ruanaidh (1997) spread spectrum technique. The author claims that the watermark is fairly robust with respect to brightness and contrast adjustment, noise adding, histogram manipulation, cropping and moderate JPEG compression up to 55% quality.

Kundur and Hatzinakos (1998) and Xie and Arce (1998) describe techniques based on the wavelet transform. Kundur embeds a mark by modifying the quantization process of Haar wavelet transform coefficients while Xie selectively inserts watermark bits by processing the image after it is in a compressed form using the SPIHT algorithm (Said 1996). A wavelet decomposition of an image contains both frequency and spatial information about the image. Hence, watermarks embedded in the wavelet domain have the advantage of being able to locate and characterise tampering of a marked image.

2.5.3. Summary of Different Methods

We summarise the different methods presented in this chapter in Table 2.1. The class to which each method belongs is indicated as fragile, semi-fragile, and digital signature, as well as the type of authentication data used and whether the method offers a possible localisation and reconstruction of the areas tampered with. From the table, we notice that, generally, the fragile watermarking methods allow only a strict integrity service, while the semi-fragile watermarking methods and methods based on external signature guarantee a content authentication. It is also interesting to note that few methods are currently able to restore, even partially, the tampered regions of the image.

Method	Class	Mark	Dependent	Integrity	Localization	Recovery
Yeung and Mintzer (1997)	fragile	Predefined logo	no	strict	yes	No
Walton (1995)	fragile	checksums	yes	strict	yes	No
Fridrich and Goljan (1999)	fragile	image comp.	yes	strict	yes	Yes
Wong (1999)	fragile	Hash function	yes	strict	yes	No
Lin and Chang (2000)	semifragile	DCT coeff.	yes	content	yes	Yes
Wolfgang and Delp (1996)	semifragile	m-sequences	no	content	yes	No
Fridrich (1998)	semifragile	Block-based	yes	content	yes	No
Kundur and Hatzinakos (1998)	semifragile	Random noise	no	strict	yes	No
Queluz (2002)	signature	edges	yes	content	yes	No
Bhattacharjee and Kutter (1998)	signature	Interest points	yes	content	yes	No
Lin and Chang (1998)	signature	DCT coeff.	yes	content	yes	No
Wolfgang and Delp (1996)	signature	Hash function	yes	strict	yes	No

Table 2.1 Summary of methods ensuring an authentication service

2.6 Requirements of Watermarking-based Authentication System

A watermarking-based authentication system can be considered as effective if it satisfies the following requirements as outlined by (Tong and Zheng-ding 2002) and (Lin and Chang 2000):

- Invisibility: The embedded watermark is invisible. It is the basic requirement of maintaining the quality of marked images. The marked image must be perceptually identical to the original under normal observation. It is a question of making sure that the visual impact of watermarking is as weak as possible so that the watermarked image remains identical to the original.
- Detect tampering: An authentication watermarking system should detect any tampering in a marked image. This is the most fundamental property to reliably test authenticity of the image. The system must be sensitive to malicious manipulations such as altering the image in specific areas.
- Security: The embedded watermark cannot be forged or manipulated. In such systems, the marking key is private and should be difficult to deduce. Insertion of a mark by unauthorised parties should be difficult.
- Identification of a manipulated area or localisation: The authentication watermark should be able to detect the location of altered areas and verify other areas as authentic. The detector should also be able to estimate what kind of modification has occurred.
- Reconstruction of altered regions: The system should have the ability to restore, even partially, altered or destroyed regions in order to allow the user to know the original content of the manipulated areas.
- Protocols: Protocols are an important aspect of any image authentication. It is obvious that any algorithm alone cannot guarantee the security of the system. It is necessary to define a set of scenarios and specifications describing the operation and rules of the system, such as management of the keys, physical security and the communication protocols between parties and so forth.

We further classify the requirements into mandatory requirements and desirable requirements as seen in Table 2.2.

Classification	Requirements
Mandatory	• Invisibility • Tamper detection • Security
Desirable	• Localize tamper • Reconstruction
Other	• Protocols

Table 2.2 Authentication watermarking requirements

2.7 Main Components of a Watermarking System

A watermarking system can be divided into three main components:

1. The generating function, f_g, of the watermark signal, W, to be added to the host signal. Typically, the watermark signal depends on a key, k, and watermark information, i. Examples of watermark information are company logo and user information.

$$W = f_g(i,k) \tag{2.4}$$

Possibly, it may also depend on the host data, Y, into which it is embedded

$$W = f_g(i,k,Y) \tag{2.5}$$

2. The embedding function, f_m, which incorporates the watermark signal, W, into the host data, Y, yielding the watermarked data Y_w. Typically, the watermark signal depends on a key, K

$$Y_w = f_m(Y,W,K) \tag{2.6}$$

3. The extracting function, f_y, which recovers the watermark information, W, from the received watermarked data, $\hat{Y}_w$, using the key corresponding to embedding and the help of the original host data, Y

$$\hat{W} = f_y(Y, \hat{Y}_w, K) \tag{2.7}$$

Or without the original host data, Y

$$\hat{W} = f_y(\hat{Y}_w, K) \tag{2.8}$$

The first two components, watermarking generating and watermarking embedding, are often regarded as one, especially for methods in which the embedded watermark is independent of the host signal. We separate them out for a better analysis of the watermarking algorithms, since some of the watermark is host signal content dependent, with the watermark generating from the host signal and being embedded back to the host signal.

Figure 2.4 shows the generic watermarking scheme. The inputs to the embedding process are the watermark, the host data, and an optional key. The watermark can take many forms, such as number, text, binary sequence, or image. The key is used to enforce security and to protect the watermark. The output of the watermarking scheme is the watermarked data. The channel for the watermarked image could be lossy and susceptible to malicious attack. The inputs for extraction are the received watermarked data, the key corresponding to the embedding key, and, depending on the method, the original data and/or watermarking information. The output of the watermark recovery process is the recovered watermark. The watermark is inspected to determine if the original image altered and recover information such as copyright status.

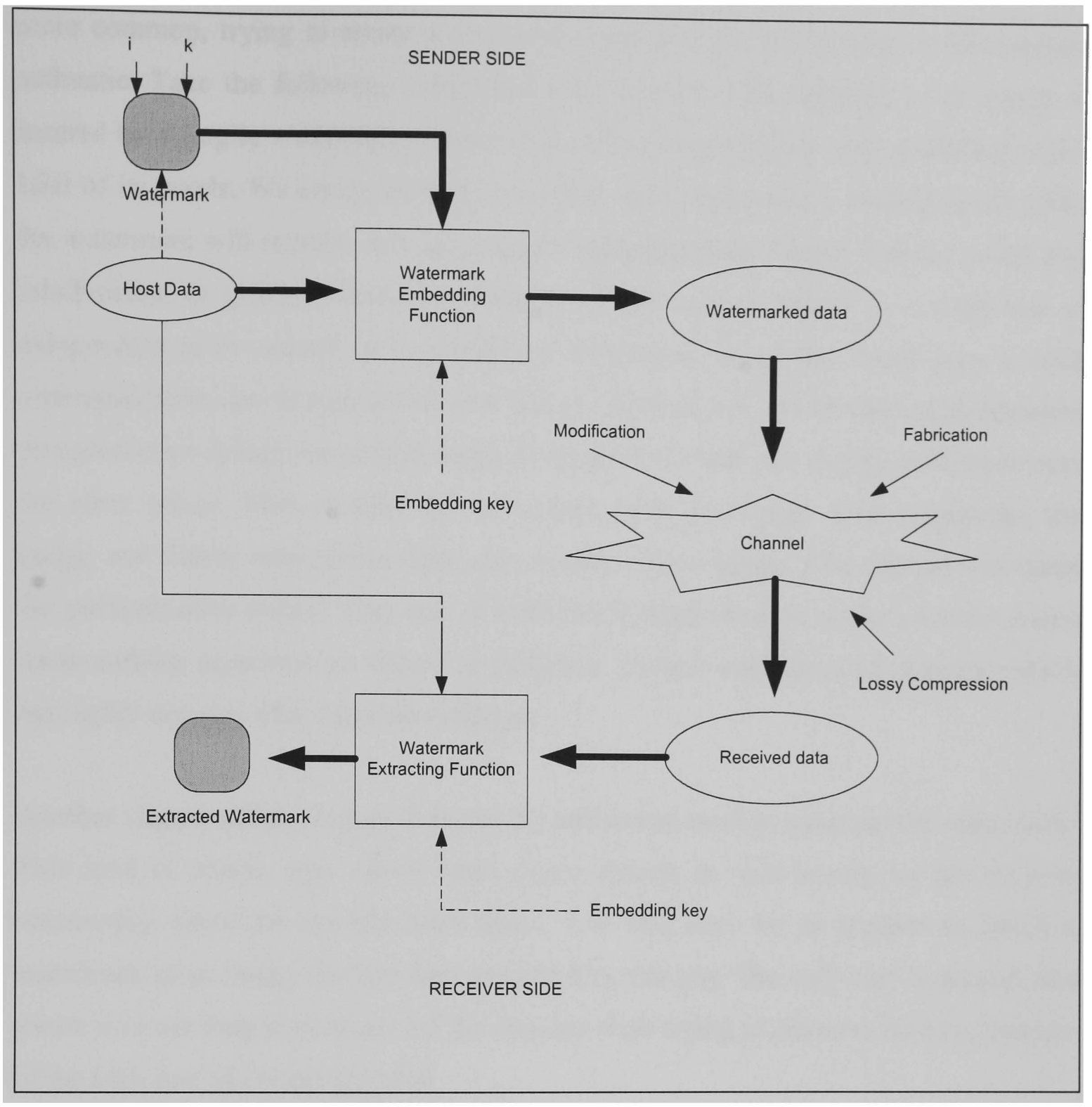


Figure 2.4. Generic Watermarking Scheme

2.8 Malicious Attacks

This section will show some of the most frequent attacks that an image authentication system has to overcome. The common objective of these attacks is to trick the authentication system, in other words, to show that an image remains authentic even though its content has been modified (or sometimes, the opposite).

One of the most common attacks against fragile watermarking systems consists of trying to modify the protected image without altering the embedded watermark, or even

more common, trying to create a new watermark that the authenticator will consider authentic. Take the following simplified example where the integrity of an image is insured by a fragile watermark, independent of the image content and embedded in the LSB of its pixels. We easily see that if we alter the image without modifying the LSB, the watermark will remain as it was, and the authentication process will not detect any falsification. In general, when the integrity of an image is based on a mark that is independent of its content, it is possible to develop an attack that could copy a valid watermark from one image into another image. By doing so, the second image becomes protected even though the second image is false. This attack can also be performed over the same image. First, extract the watermark from the image, then manipulate the image, and finally reinsert the watermark on the altered image. This process will cheat the authentication system. One way to resist this kind of attack is to use a content-based watermarking algorithm or choose a transform domain authentication scheme, which has higher security than a spatial technique.

Another classic attack tries to discover the secret key used to generate the watermark. This kind of attack, also called Brute Force Attack, is well known by the security community. Once the key has been found, it is very easy for an attacker to falsify a watermark of an image that has been protected by this key. The only way to counter this attack is to use long keys to put off the attacker from trying to discover the key, because of the high cost of computing time.

2.9 Embedding Techniques

2.9.1 Least Significant Bit Modification

The most straightforward method of watermark embedding would be to embed the watermark into the least significant bits of the cover object (Johnson and Katzenbeisser 2000). Given the extraordinarily high channel capacity of using the entire cover for transmission in this method, a smaller object may be embedded multiple times. Even if most of these were lost due to attacks, a single surviving watermark would be considered a success.

LSB substitution however, despite its simplicity has many drawbacks. Although it may survive transformations such as cropping, any addition of noise or lossy compression is likely to defeat the watermark. An even better attack would be to simply set the LSB bits of each pixel to one fully defeating the watermark with negligible impact on the cover object. Furthermore, once the algorithm is discovered, an intermediate party could easily modify the embedded watermark.

LSB modification proves to be a simple and fairly powerful tool, however lacks the basic robustness that watermarking applications require.

2.9.2 Correlation-Based Techniques

Another technique for watermark embedding is to exploit the correlation properties of additive pseudo-random noise patterns as applied to an image (Langelaar et al. 2000). A pseudo-random noise (PN) pattern $W(x, y)$ is added to the cover image $I(x, y)$, according to the equation shown below in equation 2.9.

$$I_w(x,y) = I(x,y) + k * W(x,y) \tag{2.9}$$

In equation 2.9, k denotes a gain factor, and I_W the resulting watermarked image. Increasing k increases the robustness of the watermark at the expense of the quality of the watermarked image. Rather than determining the values of the watermark from "blocks" in the spatial domain, we can employ CDMA spread-spectrum techniques to scatter each of the bits randomly throughout the cover image, increasing capacity and improving resistance to cropping (Langelaar et al. 2000). To detect the watermark, each seed is used to generate its PN sequence, which is then correlated with the entire image. If the correlation is high, then that bit in the watermark is set to "1", otherwise a "0". The process is then repeated for all the values of the watermark. CDMA improves on the robustness of the watermark significantly, but requires several orders more of calculation.

2.9.3 Frequency Domain Techniques

The classic and still most popular domain for image processing is that of the Discrete-Cosine-Transform, or DCT. The DCT allows an image to be broken up into different frequency bands, making it much easier to embed watermarking information into the middle frequency bands of an image. The middle frequency bands are chosen so that they minimise effects on the most visually important parts of the image (low frequencies) without being removed through compression and noise attacks (high frequencies).

One such technique utilizes the comparison of middle-band DCT coefficients to encode a single bit into a DCT block. To begin, we define the middle-band frequencies (F_M) of an 8x8 DCT block as shown below in figure 2.5.

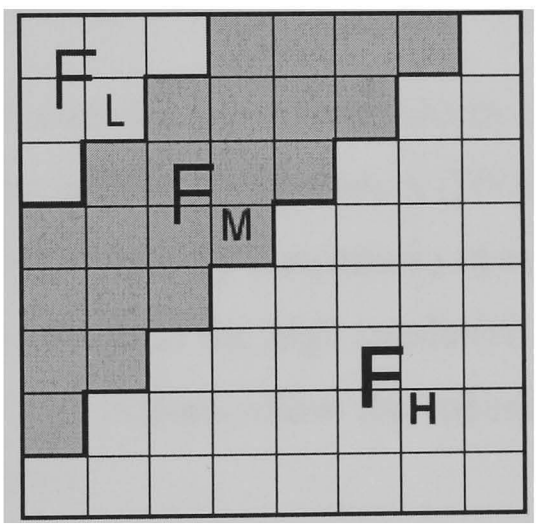


Figure 2.5. Definition of DCT Regions

F_L is used to denote the lowest frequency components of the block, while F_H is used to denote the higher frequency components. F_M is chosen as the embedding region as to provide additional resistance to lossy compression techniques, while avoiding significant modification of the cover image (Hernandez et al. 2000).

2.9.4 Wavelet watermarking

The wavelet transform provides another possible domain for watermark embedding. The DWT (Discrete Wavelet Transform) separates an image into a lower resolution approximation image (LL) as well as horizontal (HL), vertical (LH) and diagonal (HH) detail components. The process can then be repeated to compute multiple "scale" wavelet decomposition, as in the 2-scale wavelet transform shown below in figure 2.6.

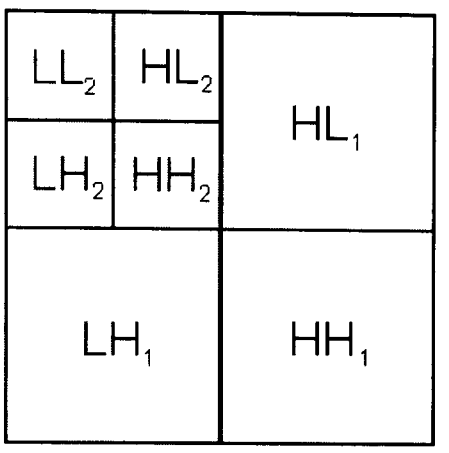


Figure 2.6. 2 Scale 2-Dimensional Discrete Wavelet Transform

One of the many advantages over the wavelet transform is that it is believed to model more accurately aspects of the human visual system (HVS) as compared to the FFT or DCT. This allows us to use higher energy watermarks in regions that the HVS is known to be less sensitive to the eye, such as the high resolution detail bands {LH, HL, HH). Embedding watermarks in these regions allow the robustness of the watermark to be increased (Langelaar et al. 2000).

One of the most straightforward techniques is to use a similar embedding technique to that used in the DCT, the embedding of a CDMA sequence in the detail bands according to the equation shown below in equation 2.10,

$$I_{W_{u,v}} = \begin{cases} W_i + \alpha |W_i| x_i, & u,v \in HL, LH \\ W_i & u,v \in LL, HH \end{cases} \tag{2.10}$$

where W_i denotes the coefficient of the transformed image, x_i the bit of the watermark to be embedded, and a scaling factor. To detect the watermark we generate the same

pseudo-random sequence used in CDMA generation and determine its correlation with the two transformed detail bands. If the correlation exceeds some threshold T, the watermark is detected.

This can be easily extended to multiple bit messages by embedding multiple watermarks into the image. As in the spatial version, a separate seed is used for each PN sequence, which is then added to the detail coefficients as in equation 2.5. During detection, if the correlation exceeds T for a particular sequence a "1" is recovered; otherwise a zero. The recovery process then iterates through the entire PN sequence until all the bits of the watermark have been recovered.

Chapter 3

Medical Image Watermarking

3.1 Introduction

This chapter discusses the properties of medical image watermarking and discusses techniques available for tamper localisation and image reconstruction. This chapter is structured as follows:

- Section 3.2 highlights the properties of medical image watermarking and outlines the objectives for watermarking in medical domain.
- Section 3.3 introduces reversible watermarking and describes such a scheme.
- Section 3.4 introduces the concept of the region of interest (ROI) in medical images.
- Section 3.5 discusses authentication watermarking with localisation capabilities and the security risk of such techniques.
- Section 3.6 presents a few techniques available for using watermark as an aid in the reconstruction of image that have been corrupted.
- Section 3.7 gives a review of previous work done on medical images.

- Section 3.8 introduces Digital Imaging and Communications in Medicine (DICOM) standard and Picture Archiving and Communication System (PACS).
- Section 3.9 discusses methods for evaluating perceptual impacts of watermarks.

3.2 Properties of Medical Image Watermarking

Security of medical information, derived from strict ethics and legislative rules, gives rights to the patient and duties to the health professionals. This imposes three mandatory characteristics: confidentiality, reliability and availability:

- Confidentiality means that only the entitled persons have access to the information and that information is not made available or disclosed to unauthorised individuals, entities or processes
- Reliability which has two aspects; Integrity: the information has not been modified or destroyed by non-authorized person, and authentication: proof that the information belongs indeed to the correct patient and is issued from the correct source
- Availability is the ability of an information system to be used by the entitled persons in the normal conditions of access and exercise.

Security risks of medical images can vary from random errors occurring during transmission to lost or overwritten segments in the network during exchanges in the intra- and inter-hospital networks. One must also guarantee that the header of the image file always matches that of the image data. In addition to these unintentional modifications one can envision various malicious manipulations to replace or modify parts of the image, called tampering. The usual constraints of watermarking are invisibility of the mark, capacity, secrecy to unauthorised persons, and robustness to attempts to suppress the mark. These demands also exist in the medical domain but additional constraints are added. Three main objectives are foreseen in the medical domain (Coatrieux et al. 2000, Mintzer et al. 1997):

1. Imperceptible / Reversible Watermarking

Medical tradition is very strict with the quality of biomedical images. Thus the watermarking method must be reversible, in that the original pixel values must be exactly recovered (Macq and Dewey 1999). This limits significantly the capacity and the number of possible methods.

An alternative way is to define regions of interest, to be left intact, and leave us with regions of insertion where a watermark could be inserted and does not interfere or disturb the radiologist.

2. Integrity Control

The "secure camera" concept applies also to biomedical images, especially in the context of legal aspects and insurance claims. There is thus a need to prove that the images on which the diagnoses and any insurance claims are based have preserved their integrity.

3. Authentication

A critical requirement in patient records is to authenticate the different parts of the electronic patient record, in particular the images. More often an attached file or a header, which carries all the needed information, identifies an image. However, keeping the meta-data of the image in a separate header file is prone to forgeries or clumsy practices. An alternative would be to embed all such information into the image data itself.

3.3 Reversible Watermarking

Reversible watermarking means that the original data will be available after a watermark is embedded. In summary any reversible watermarking system comprises the following steps:

i. Embedding a digital watermark, w in an original image x resulting in $y = f(x, w)$

ii. Transmitting the watermarked image y from the encoder to the decoder through an error-free transmission channel

iii. Extracting the watermark image w and restoring the original image $x = f^{-1}(y, w)$

The concept for reversible data embedding first appeared in an authentication method for images in a patent from the Eastman Kodak Company (Honsinger et al. 2001). There are several techniques for reversible data embedding and the scheme proposed by Goljan et al (2001) will be described.

Let us assume that the original image is a greyscale image with M x N pixels and with pixel value from the set P. For example, for an 8-bit greyscale image, P= {0, …, 255}. They start with dividing the image into disjoint groups of n adjacent pixels (x_1, …, x_n). For example we can choose groups of n = 4 consecutive pixels in a row. They also define so called discrimination function f that assigns a real number $f(x_1,...,x_n) = f(G) \in \Re$ to each pixel group G = (x_1, …, x_n). The purpose of the discrimination function is to capture the smoothness or regularity of the group of pixels G. Discrimination function used was:

$$f(x_1, x_2, ..., x_n) = \sum_{i=1}^{n-1} |x_{i+1} - x_i|$$

Then an invertible operation F on P called 'flipping' is defined. Flipping is a permutation of grey levels that consists of 2-cycles. Thus, F will have the property that F^2 = Identity or $F(F(x)) = x$ for all $x \in P$. For example, the permutation F_{LSB} defined as 0 ↔ 1, 2 ↔ 3, …, 254↔255 correspond to flipping the LSB of each grey level. The permutation 0↔2, 1↔3, 4↔6, 5↔7, … corresponds to an invertible noise with larger amplitude. Discrimination function f and the flipping operation F were used to define three types of pixel groups: R, S and U.

Regular groups: $G \in \mathrm{R}$ if $f(F(G)) > f(G)$

Singular groups: $G \in \mathrm{S}$ if $f(F(G)) < f(G)$

Unusable groups: $G \in \mathrm{U}$ if $f(F(G)) = f(G)$

3.4 Region of Interest (ROI)

Typically, a medical image is diagnosed before being archived in long-term storage, so the significant part of the image is already determined. The significant part is called ROI (Region Of Interest), which must be preserved without any lack of information. In Chapter 4, we propose a strict authentication watermarking considering ROI. In general, the ROI is stored as it is or compressed by a lossless algorithm and the other part is compressed by a lossy algorithm, which can achieve a higher compression rate than lossless compression algorithm (Wakatani 2002).

Distant learning is one of applications using a database of medical images, which may refer to the image of a newly discovered medical case, and there may be images with the ROI part for long-term storage. Therefore, it is desirable that the copyright and integrity of the medical image with ROI part are protected. However it is impossible to embed signature information into the ROI part since the ROI must be kept without any distortion.

3.5 Localisation and Security Risk

Many authentication methods based on watermarking have the ability to identify regions of the image that have been tampered with, while verifying that the remainder of the image has not been changed. This capability is referred to as localisation. Localisation is useful because knowledge of where an image has been tampered with can be used to infer: 1) the motive for tampering; 2) a possible attacker; and 3) whether the alteration is legitimate. For example, consider an ultrasound image of a kidney. If our authenticator simply states that the image has been modified, the tampered image is useless. However, if the authenticator also indicated that the modification only occurred within the region of non -interest, the image is still very useful for learning purposes.

Most localised authentication methods rely on some form of block-wise authentication, in which the image is divided into a number of spatial regions, each of which is authenticated separately. If part of the image is modified, only the affected regions fail to authenticate.

There are a number of security risks associated with localised authentication systems. Although the risks discussed here can be countered with simple modifications, it is important to be aware of them. We are concerned with forgery attacks in which an attacker wishes to embed a valid watermark into either a modified or false image. Two basic attacks will be examined. In search attacks, the attacker is assumed to have a detector that can determine whether the image is authentic or not. In collage attacks, the attacker is assumed to have two or more images embedded with the same watermark.

3.5.1 Search Attacks

Let us consider the situation in which everyone, including potential attackers, has access to a watermark detector. This situation might arise, for example, if images are being distributed to the public over the Internet, and an authentication system is to be used to guarantee that each image is delivered without corruption or tampering. In theory, the attacker can use the detector to defeat any authentication system, regardless of whether it is localised or not. To do so requires a brute-force search. To embed a forged watermark into an image, the attacker can enter slightly modified versions of the image into the detector until one is found that the detector reports as authentic.

In practice, this search would usually be prohibitive. However, with a block-wise authentication system, the search space can be considerably smaller. The attacker can perform a separate, independent search on each block. If the block size is small enough, the search becomes feasible. Such attacks can be countered by choosing a sufficiently large block size.

3.5.2 Collage Attacks

The second category of attacks relies on having access to one or more authentic watermarked images. By examining these images, an attacker can come up with sets of blocks that are authentic and construct a forged image from them like a 'collage'. Holliman and Memon (2000) describe an attack applicable to block-wise watermarks, in which a cryptographic signature is embedded in each block and the signature depends only on the content of the block itself. Consider what happens in such a system when two blocks of a watermarked image are interchanged, thereby changing the image as a whole. Because each block contains a self-authenticating watermark, and because each block remains unaltered, the image is deemed authentic. Thus, even if all blocks are scrambled into a random order, the system will regard the entire image as authentic.

By exploiting this weakness of block-wise independent system, it is possible to create a completely new image that is assembled from the set of independent, authentic blocks. Suppose an attacker has a number of images available, all watermarked using the same key, this can be viewed as a large database of authentic blocks from which a new image can be built. To forge a watermark in an unwatermarked image, the attacker divides the image into blocks and replaces each block with the most similar block from the database. With a large enough database of watermarked images, the results may be quite effective.

The solution to counter these types of attacks is to use a different key for watermarking every image. However, such an approach is not always feasible, in that a given image can only be authenticated if the correct key is available. The keys would need to be either known to the users or stored as associated data.

A more practical approach suggested in Holliman and Memon (2000) is to make the blocks overlap so that the signature of each block depends on surrounding data, as well as the data within the block itself. This introduces ambiguity to the localisation, because a change in one block will change the signature that must be embedded in its neighbours. This complicates the attacker's attempt to build an image out of

watermarked blocks, as each block must match up properly with the neighbouring blocks.

3.6 Restoration

From the previous section we have seen that it is possible to verify if an image has been altered and determine where it has been altered. This section considers the manner in which an altered or tampered image might be restored.

There are two restoration strategies: exact restoration and approximate restoration. In exact restoration the image is restored to its original state, where the goal is to create a perfect copy. This is a well-studied problem in communication and will be discussed in the next section. Approximate restoration is a more recent concept that seeks to restore an image to approximately the original state while accepting that there will be differences between the restored and original image. However, the restored image may still be valuable if these differences are not significant.

3.6.1 Embedded Redundancy

It is well known that error detection and error correction codes allow changes in data to be detected, and in the latter to be corrected. Error correction codes (ECC) are widely used in communication and data storage to maintain the integrity of digital data. ECC codes are like digital signatures, are appended to the data. However, there are important differences between ECC codes and signatures:

1. Digital signatures are used to verify that the data has not been altered.
2. Digital signatures need fewer bits than ECC to detect a change has occurred than are needed to perform correction (Shannon 1948).
3. ECC codes usually assume a maximum number of bit changes. If this number is exceeded, it is possible for errors to go undetected

The size of an ECC code is usually very much larger than a digital signature. In fact, an ECC code can represent a significant fraction of transmitted bits. The size of the ECC

code determines both the maximum number of bit changes that can be detected and the maximum number of bits that can be corrected.

An image can be considered as a collection of bits, and a variety of different error correction codes can be applied (e.g., Hamming codes, turbo codes, and trellis codes). This metadata can be represented as watermark. For example, a Reed Solomon ECC code can be used to generate parity bytes for each row and column of an image (Lee and Won 1999, Lee and Chen 2002). These parity bytes can be embedded as a watermark in the two significant bit planes of the image. It is reported that for a 229 X 229 image, up to 13 bytes in a single row or column can be corrected. Even if the errors cannot be corrected, they can be localised, because parity bytes are calculated for each row and column.

This method is modified when it is expected that errors will come as bursts (Lee and Won 2000). This is the case when a localised region of an image has been modified or cropped. To increase the resistance to burst errors, the locations of the pixels in the image are randomised prior to calculation of the ECC codes. This randomisation is a function of a watermark key.

If we want to restore an image to its original state, a very significant cost must be incurred to store the ECC codes. If this cost is too high, or the resources are simply unavailable, then approximate restoration techniques may be a good compromise.

3.6.2 Self-embedding

A further approach is self-embedding (Lin and Chang 2000, Fridrich and Goljan 1999), which is a highly compressed version of the image in the image itself. Thus, if portions of the watermarked image are removed or destroyed, these modified regions can be replaced with their corresponding low-resolution versions.

In the algorithm of Fridrich and Goljan (1999), a highly compressed JPEG (50% quality factor) version of the image is produced. This low-resolution image requires only one bit per pixel and can thus be inserted in the LSB plane of the image. However, each

compressed DCT block is not simply inserted into the LSB of its corresponding spatial block; rather the binary sequence is first encrypted and then inserted in the LSB plane of a block that is some distance away and in a randomly chosen direction. The authors suggest a minimum distance of 3/10 the image size. The random mapping is generated using a key that must be known to both the embedder and the detector. Storing the low-resolution version of a block some distance away from the corresponding block allows this block to be restored even if it has been completely deleted. A higher quality reconstruction is possible if more bits are allocated to the storage of the low-resolution image. The method is severely affected by any modifications to the encoding bit plane.

3.6.3 Blind Restoration

An alternative approach to approximate correction of errors is based on blind restoration. Blind restoration attempts to first determine what distortions an image has undergone, and then to invert these distortions to restore the image to its original state (Kundur and Hatzinakos 1996). Such a process is only appropriate if the distortion is invertible. Thus, blind restoration is not useful against, for example, clipping.

The method assumes that the image and the watermark undergo the same distortion. If the watermark is made capable of determining the distortion that has occurred, then an inverse process (assuming such a process exists), can be applied to restore the watermark and the image. A combination of blind restoration and self-embedding may also be appropriate. In principle, blind restoration might allow a lower resolution image to be embedded. This is because at least some of the distortion may be invertible. In addition, where clipping or other non-invertible distortions have been applied, the self-embedded information allows for a low-resolution restoration.

3.7 Previous Work on Medical Image Watermarking

Digital watermarking can imperceptibly embed messages without changing image size or format. When applied to medical images, the watermarked image can still conform to the DICOM format (Guo and Zhuang 2003). Some researchers already apply watermarking technique to medical data. Zhou et al (2001) present a watermarking

method for verifying the authenticity and integrity of a digital mammography image. They used a digital envelope as a watermark and the least significant bits (LSB) of one random pixel of the mammogram are replaced by one bit of the digital envelope (DE) bit stream. Instead of the whole image data, only partial image data (i.e., the most significant bits (MSB) of each pixel is used for verifying integrity). Cao et al (2003) extend their work on digital envelopes and embed their DE by making a random walk sequence and replacing the LSB of each selected pixel.

Other researchers adapt digital watermarking for interleaving patient information with medical images to reduce storage and transmission overheads (Acharya et al 2001). Again, the LSB of image pixels are replaced for embedding. Chao et al (2002) propose a discrete cosine transform (DCT) based data-hiding technique that is capable of hiding those EPR related data into a marked image. The information is embedded in the quantized DCT coefficients. The drawback of the above watermarking approaches is that the original medical image is distorted in a non-invertible manner. Therefore it is impossible for a watermark decoder to recover the original image.

A reversible watermarking scheme involves inserting a watermark into the original image in an invertible manner, so that when the watermark was later extracted, the original image can be recovered completely. Research has also been done in the area of reversible watermarking in medical images. Trichili et al (2002) proposes an image virtual border as the watermarking area. Patient data is then embedded in the LSBs of the border. Guo and Zhuang (2003) present a scheme where the digital signature of the whole image and patient information is embedded. They define three types of pixel groups as suggested by Goljan et al (2001), R, S, and U. The problem with this technique is that the capacity for embedding is highly dependent on the number of R and S group of pixels. The maximum number of bits available for embedding in Guo and Zhuang's (2003) scheme for ultrasound images of 640x480x8 bits is 1668 bits. This will give an embedding rate of 0.0054 bits/pixel.

Cho et al (2001) studied watermark methods appropriate for medical images and conclude that the spatial watermark method such as LSB had the advantage that it did

not damage the important information if the watermark was embedded outside the region of interest.

3.8 DICOM and PACS

The initial goal in developing a standard for the transmission of digital images was to enable users to retrieve images and associated information from digital imaging equipment in a standard format that would be the same across multiple manufacturers. The first result was the American College of Radiology (ACR)-National Electrical Manufacturers' Association (NEMA) standard, which specified a point-to-point connection. The rapid evolution of computer networking and of picture archiving and communication systems meant that this point-to-point standard would be of limited use. Consequently, a major effort was undertaken to redesign the ACR-NEMA standard by taking into account existing standards for networks and current concepts in the handling of information on such networks. The Digital Imaging and Communications in Medicine (DICOM) standard was the result of this effort. Its popularity has made discussion, if not implementation, of the standard common whenever digital imaging systems are specified or purchased.

The use of DICOM has now extended beyond only an image and has been adapted to manage data from many medical specialties (e.g., pathology, ECG). It is also a global standard being adopted by the European standards organization, the Comitee Europeân de Normalisation (CEN), as MEDICOM standard. In Japan, the Japanese Industry Association of Radiation Apparatus and the Medical Information Systems Development Centre have adopted portions of DICOM that pertain to the exchange of images on removable media and are considering DICOM for future versions of the Medical Image Processing Standard. The DICOM standard is now being maintained and extended by an international, multi-specialty committee (Horii 1997).

The DICOM standard has become the predominant standard for the communication of medical images. The DICOM standard consists of multiple documents (National Electrical Manufacturers Association 2003), which at the time of writing consist of 16

published parts. Each DICOM document is identified by a title and standard number, which takes the form "PS 3.X-YYYY," where "X" is commonly called the part number and "YYYY" is the year of publication. For example, DICOM Part 2 has a title of "Conformance" and document number PS 3.2-2003. In informal usage, the year is often dropped. Watermarking is not currently considered in any part of this standard.

Picture archiving and communication system (PACS) is a work flow-integrated system for managing medical image and related data. It is designed to streamline operations throughout the whole patient care delivery process (Huang 2003). PACS was originally developed for radiology services over 20 years ago to capture digital medical images rather than in film-based media. Figure 3.3 describes the enterprise level web-based image/data EPR server with archive.

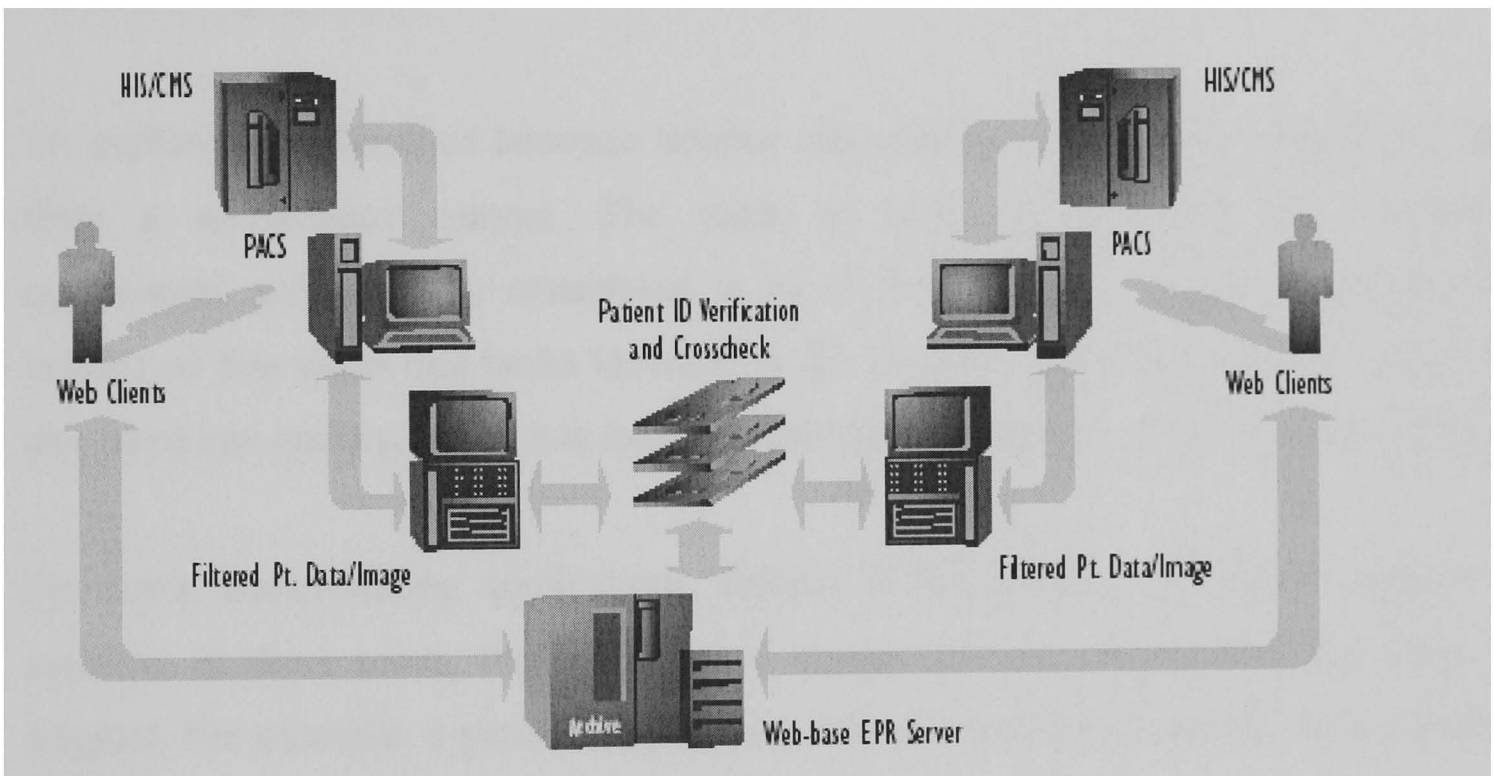


Figure 3.3 Enterprise Level Web-based Image/Data EPR server with archive

3.9 Evaluating Perceptual Impact of Watermarks

There is few, if any, watermarking systems producing watermarks that are perfectly imperceptible. However, the perceptibility of a given system's watermark may be high or low compared against other watermarks or other types of processing, such as compression. In this section we address the question of how to measure that

perceptibility, so that such comparison can be made. This section begins with a discussion of two types of perceptibility that exist as causes for concern.

3.9.1 Fidelity and Quality

In the evaluation of a watermarking system, there are two different types of perceptibility that can be judged: fidelity and quality. Fidelity is a measure of the similarity between images before or after watermarking (Cox et al. 2002). A high fidelity reproduction is a reproduction that is very similar to the original. A low fidelity reproduction is dissimilar or distinguishable from the original. Quality on the other hand is an absolute measure of appeal. A high quality image simply looks good. It has no obvious processing artefacts. Both types of perceptibility are significant in evaluating watermarking systems.

To explore the difference between fidelity and quality, consider an example of video from a surveillance camera. The video is typically greyscale, low resolution, compressed and generally considered to be of low quality. Consider a watermarked version of this video that looks identical to the original. This watermarked video must also have low quality, but as it is indistinguishable from the original, it has high fidelity.

For some watermarking applications, fidelity is the primary perceptual measure of concern. In these cases, the watermarked image must be indistinguishable from the original. For example, a patient may require this of a watermark applied to his medical images.

3.9.2 Human Evaluation Measurement Techniques

Although the claim of imperceptibility is often made in the watermarking literature, rigorous perceptual quality and fidelity studies involving human observers are rare. Some claims of imperceptibility are based on automated evaluations, discussed in section 3.9.3. However, many claims are based on a single observer's judgements on a

small number of trials. These empirical data points are not sufficient for proper perceptual evaluation or comparison of watermarking algorithms.

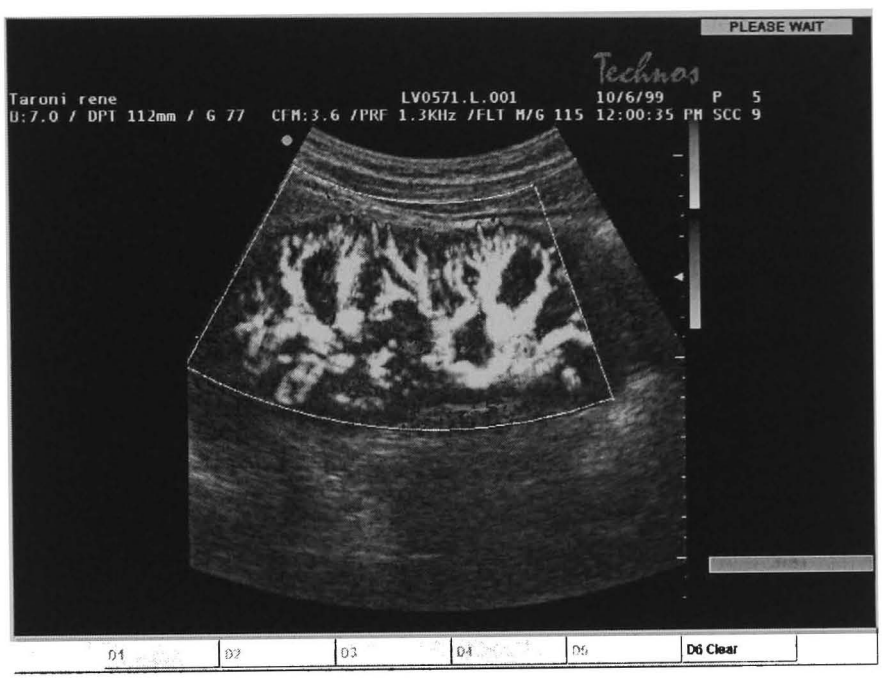
An experimental paradigm for measuring perceptual phenomena is the two alternatives forced choice (2AFC) (Green and Swets 1974). In this procedure, observers are asked to give one of two alternative responses to each of several trial stimuli. For example, to test the quality impact of a watermarking algorithm, each trial of the experiment might present the observer with two versions of one image. One version of the image would be the original, the other would be watermarked. The observer, unaware of the differences between the images, must decide which one is higher in quality. In the case where no difference in quality can be perceived, the responses are expected to be random. Random choices suggest that observers are unable to identify one selection as being consistently better quality than the other. Thus, 50% correct answers correspond to zero JND, while 75% correct corresponds to one JND (Cox et al. 2002).

The 2AFC technique can also be used to measure fidelity. Consider an experiment in which the observer is presented with three images (Figure 3.4). One is labelled as the original. Of the other two, one is an exact copy of the original and the other is the watermarked version. The subject must choose which of the two latter images is identical to the original. The results are tabulated and examined statistically. Any bias in the data represents the fact that the observers could distinguish between the original and watermarked images, and serves as a measure of the fidelity of the watermarking process.

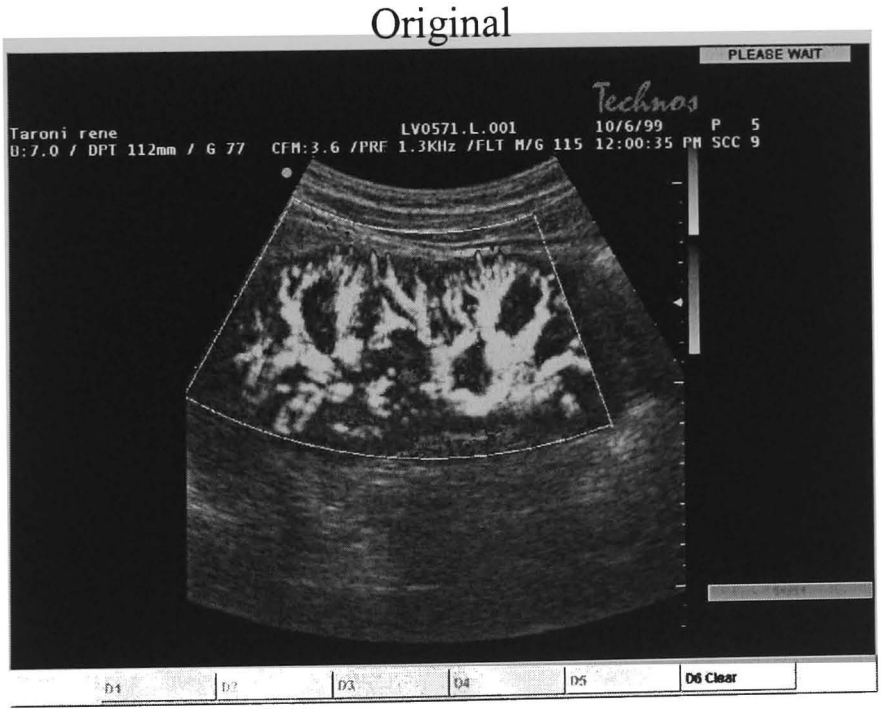
Five-Grade Scale	
Quality	Impairment
5 Excellent	5 Imperceptible
4 Good	4 Perceptible, but not annoying
3 Fair	3 Slight annoying
2 Poor	2 Annoying
1 Bad	1 Very annoying

Table 3.1 Quality and impairment scale as defined in ITU-R Rec. 500

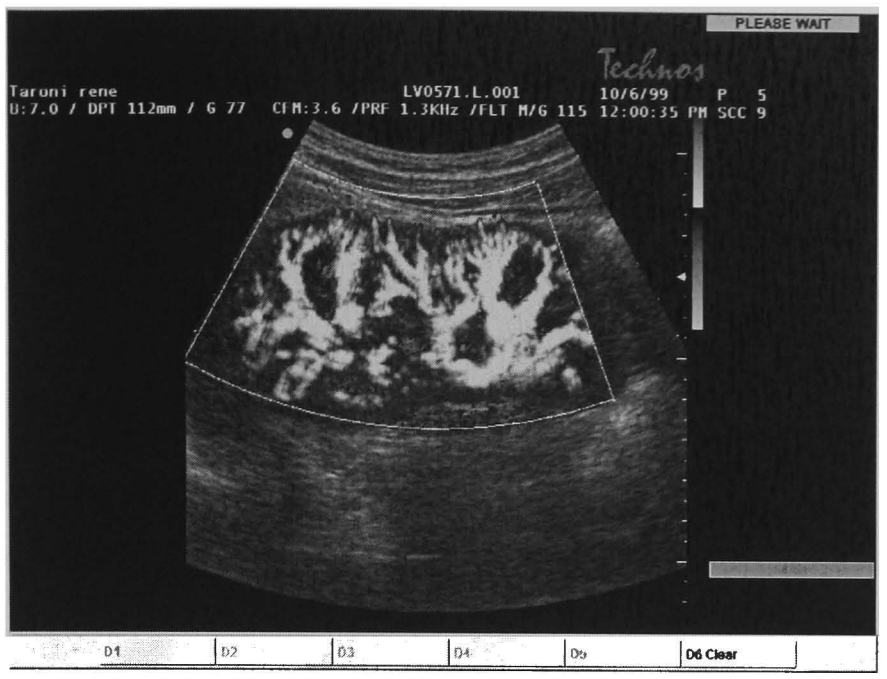
A second, more general experimental paradigm for measuring quality allows the observers more latitude in their choice of responses. Rather than selecting one of two images as 'better', observers are asked to rate the quality of an image, sometimes with a reference to a second image. For example, the ITU-R Rec. 500 quality rating scale specifies a quality scale and an impairment scale that can be used for judging the quality of television pictures (ITU 2000). These scales, summarised in Table 3.1, have been suggested for use in the evaluation of image watermarking quality (Kutter and Hartung 2000).



Original



A



B

Figure 3.4 A two alternative, forced choice experiment studying image fidelity.

3.9.3 Automated Evaluation

The experimental techniques outlined previously can provide very accurate information about the fidelity of watermarked content. However, they can be very expensive and are not easily repeated. An alternative approach is the use of an algorithmic quality measure based on a perceptual model. The goal of a perceptual model is to predict the response of an observer. The immediate advantages of such a system are that it is cheaper and faster to implement and the evaluation can be repeated so that different methods can be compared directly.

Ideally, a perceptual model (function) intended for automated fidelity tests should predict the results of tests performed with human observers. However, for the purposes of comparing the fidelity of different watermarking algorithms, or watermarking strength, it is sufficient for the model to provide a value that is related to the results of human tests, that is to produce a measure of the perceptual distances between watermarked and unwatermarked images (Cox et al. 2002). One of the simplest distance functions is the mean squared error (MSE). This is defined as:

- The mean square error (MSE),

$$MSE = \frac{1}{n}\sum_{i}^{n}(I_i' - I_i)^2,$$

which is the averaged term by term difference between the original image, I, and the watermarked image, I'. Although MSE is often used as a rough test of a watermarking system's fidelity impact, it is known to provide a poor estimate of the true fidelity (Girod 1993).

Some perceptual distance functions are asymmetric. In these functions, the two arguments have slightly different interpretations. By convention, the first argument is interpreted as an original image, and the second as a watermarked version of it. For example, one commonly used asymmetric distance is based on the reciprocal of the signal-to-noise ratio (SNR). This is defined as:

- The signal-to-noise ratio (SNR),

$$SNR(dB) = 10\log_{10}\frac{\sum_{i}^{n} I_i^2}{\sum_{i}^{n}(I_i' - I_i)^2},$$

- The peak signal to noise ratio (PSNR),

$$PSNR(dB) = 10\log_{10}\frac{\max I^2}{MSE},$$

where max I is the peak value of the original image (usually 255 for 8 bit grey-scale image). The PSNR of an image is a typical measure used for assessing image fidelity by considering that the just noticeable distortions are uniform in all coefficients in a specific domain, such as spatial domain, frequency domain, or some other transform domain. It is well known that these distance functions are not well correlated with the human visual system (Kutter and Hartung 2000). In this thesis, PSNR is used as a measure of image fidelity.

There are a few assumptions that were made:

- All images will be stored in their original sizes.
- All tampering is done locally, using image editing software and includes:
    - Cut and paste (including cutting from another watermarked image)

- Cloning
- Healing brush

Chapter 4

Strict Authentication Watermarking (SAW)

4.1 Introduction

This chapter proposes two types of strict authentication watermarking for medical images. This chapter is structured as follows:

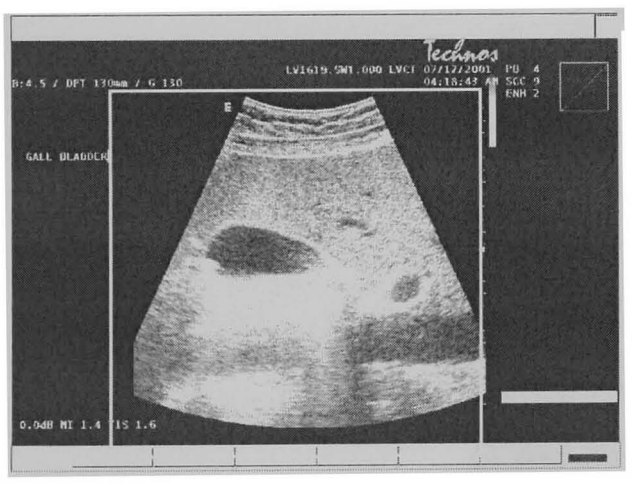
- Section 4.2 proposes strict authentication watermarking for ultrasound images. In this scheme, we define region of interest (ROI) by taking the smallest rectangle around an image. The watermark is generated from hashing the area of interest. The embedding region is considered to be outside the region of interest as to preserve the area from distortion as a result from watermarking.

- Section 4.3 proposes another strict authentication watermarking that is robust to some degree of JPEG compression (SAW-JPEG). JPEG compression will be

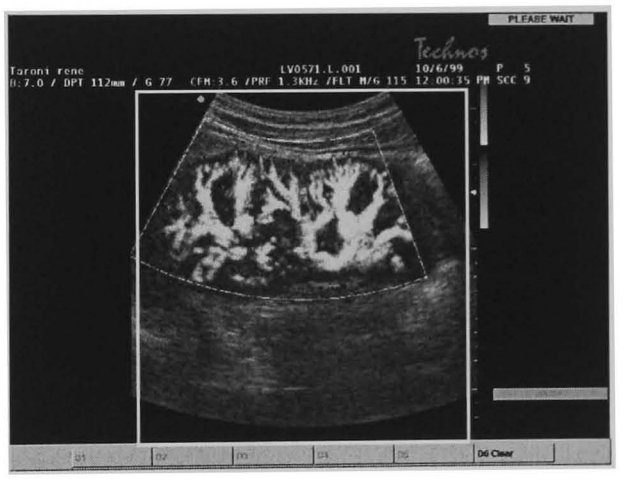
reviewed. To embed a watermark in the spatial domain, we have to make sure that the embedded watermark will survive JPEG quantization process.

4.2 Strict Authentication Watermarking (SAW)

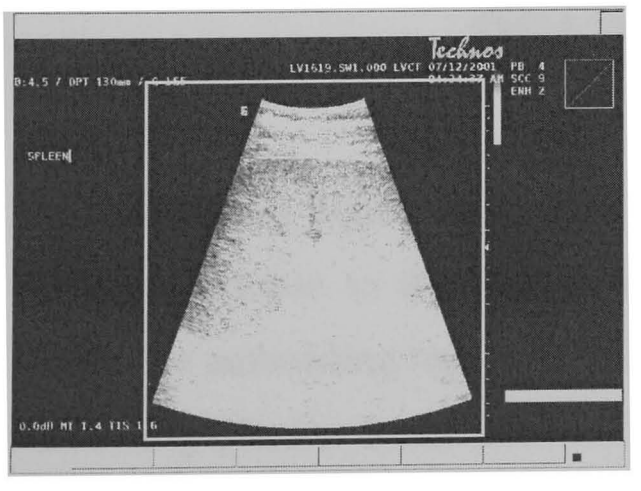
We propose a strict authentication watermarking for ultrasound images, where the watermark embedded will remain lossless in storage and transmission. In order to reduce the effects on the image, it is to be embedded into a constrained area that is defined to be outside the ROI. In this scheme, we define region of interest (ROI) by taking the smallest rectangle around an image (figure 4.1). This border will be used for our watermark embedding later. The watermark is generated from hashing the area of interest. The embedding region is considered to be outside the region of interest as to preserve the area from distortion as a result from watermarking. Our scheme is an enhancement of the scheme proposed by Cao et al (2003) with reversible capability.



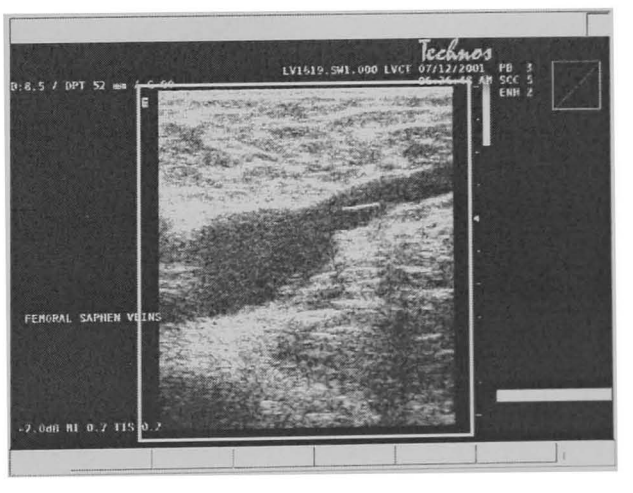
(a) gallbladder



(b) kidney



(c) spleen



(d) vein

Figure 4.1. Ultrasound images with a border drawn around them

4.2.1 Watermark

The watermark is generated by creating a hash value from the region of interest (inside the rectangle), X of size m x n. .The pixels will be arranged in a string, S.

$$S = B(X_{(1,1)}X_{(1,2)}...X_{(1,m)}X_{(2,1)}...X_{(m,n)}), \tag{4.1}$$

where X_{mn} is the 8 bit binary value of each pixel.

The hash value is obtained by applying a hash function to the string

$$Hash = \mathrm{H}(S) \tag{4.2}$$

where H is any hash function such as MD5 and SHA256.

4.2.2 Embedding Region and Domain

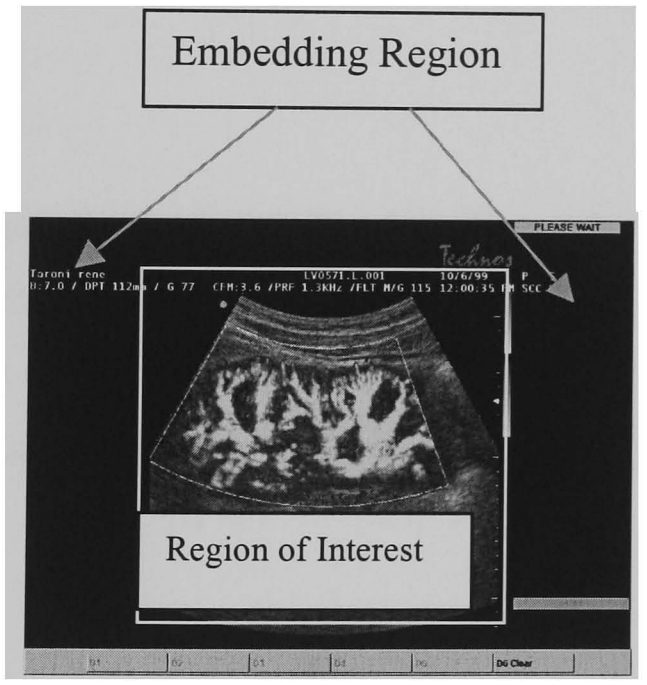


Figure 4.2. Embedding region

The embedding region is considered to be outside the region of interest in order to prevent distortion to the area as a result of adding the watermark. In an ultrasound image, the embedding region is normally a dark region with pixel values 0. This feature will be exploited to create a reversible or invertible watermarking.

In strict authentication watermarking, it is vital that the system will detect any change to the image. Fragile watermarking is the most appropriate as any change in the image will also affect the watermark. Least Significant Bit (LSB) watermarking has an advantage as the method of choice, as it is well known that LSB is vulnerable and easy to manipulate.

4.2.3 Security

A watermark is secure if it is able to resist intentional tampering by an attacker. This would include remaining secure even when the attacker knows the algorithm for embedding and extracting the watermark.

The strength of the security of the watermark will depend on the key chosen. A typical attack would involve removing the watermark, changing the image, then recalculating and embedding the new hash value into the embedding area. If the key for calculating the hash value remains secret, then the system may be considered secure. The secret key can be used to create the hash value and to create a random embedding. These will be examined in turn.

- Key for hashing

A key can be used to create the hash for the selected region. In this method, the sender and recipient will use the same key to carry out the hash function. The hash value obtained will be used as the watermark. At the recipient end, the key will be used to carry out the hash function on the received image and the hash value will be compared with the hash extracted.

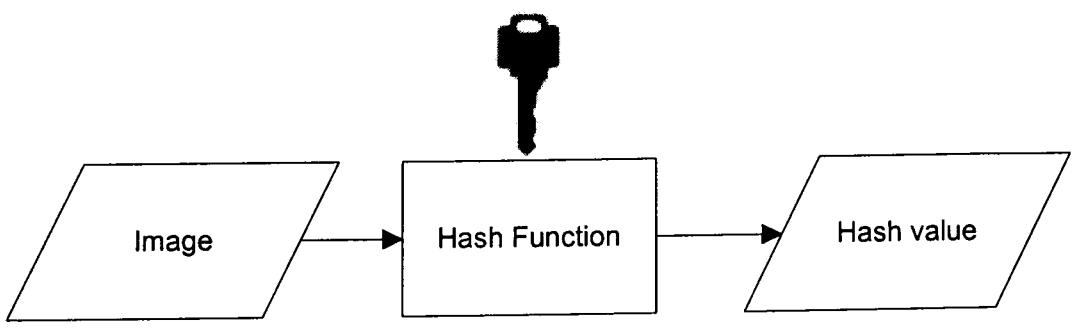


Figure 4.3. Key for hash

- Key for embedding

A key used for embedding will determine the random mapping of watermark values into the embedding region as in figure 4.4.

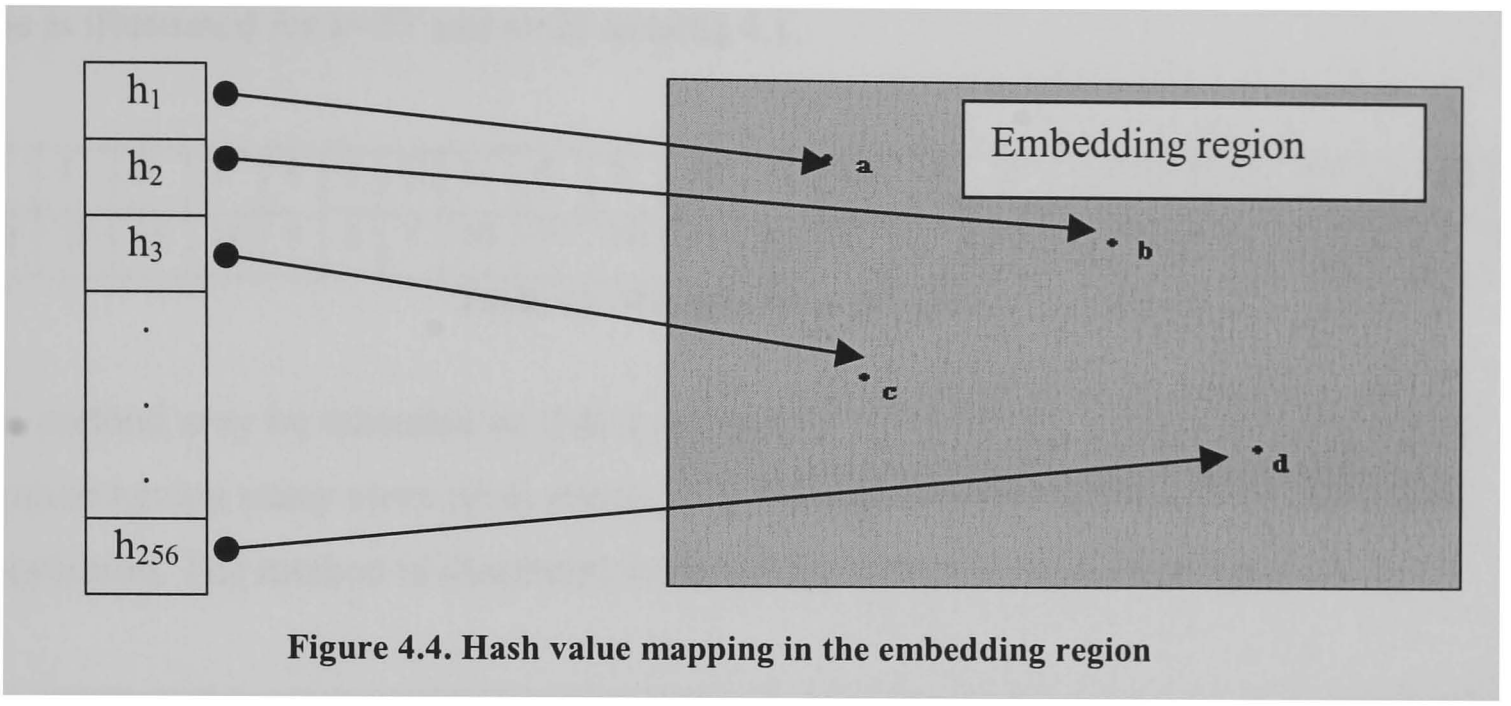


Figure 4.4. Hash value mapping in the embedding region

This supposes that the number of points or pixels in the embedding region is greater than or equal to the number of bits in the hash value holds. As an example, suppose the pixels are arranged as a simple raster scan as in figure 4.5.

1	2	3	4	5
6	7	8	9	10
11	12	13	14	15
16	17	18	19	20

Figure 4. 5. Embedding region of 5 x 4 pixels

which may be described by the mapping function of equation 4.3:

$$f(x) = x \bmod n \tag{4.3}$$

where x is the bit position and $x \in \{1, h\}$ and n is the total number of pixels available for embedding. In this example, we use h=20 to make full use of the embedding region. Applying equation 4.3, bit position one will be located in pixel number one, bit position

two will be located in pixel number 2 and so on. By using a key, k, the position will be randomised. If a simple function, e.g. equation 4.4 is applied,

$$f(x) = kx \bmod n \tag{4.4}$$

where k is a prime key, then the mapping will be a randomised one-to-one mapping. This is illustrated for k=37 and n=20 in table 4.1.

X	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
F(x)	18	15	12	9	6	3	20	17	14	11	8	5	2	19	16	13	10	7	4	1

Table 4.1. Mapping for k=37, n=20

The method may be extended so that a number, h, of hash values are distributed within a region having many more pixel points, n so that the results appears as a sparse random distribution. The method is illustrated for k=37, h= 20 and n=100 in table 4.2.

x	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
f(x)	38	75	12	49	86	23	60	97	34	71	8	45	82	19	56	93	30	67	4	41

Table 4.2. Mapping for k=37, h=20,n=100

If the embedding region is 10 x 10 pixels, then the distribution of embedding will be pictured as in figure 4.6.

			19				11		
	3							14	
		6							17
			9				1		
20				12				4	
					15				7
						18			
10				2					
	13				5				
		16				8			

Figure 4.6. Distribution of embedding for k=37, h=20, n=100

This simple method relies on the use of symmetric keys, which has an associated problem of key management. This is beyond the scope of this research. In practice asymmetric key systems are favoured; these are discussed in the next section.

4.2.4 Hashing – SHA256

The Secure hash Algorithm (SHA) was developed by the National Institute of Standards and Technology (NIST) and published as a federal information processing standard (FIPS PUB 180) in 1990. The algorithm is an iterative, one-way hash function that can process a message to produce a condensed representation called a *message digest*. The algorithm enables the integrity of a message to be determined and any change to the message will, with a very high probability, result in a different message digest. This property is useful in the generation and verification of digital signatures and message authentication codes. It is based on a public/private key, and thus overcomes the problem of key management.

4.2.5 Method

SHA-256 may be incorporated into a watermarking algorithm as shown in Figure 4.7. The general methodology and principles as listed below:

At sender site

1) **Define Area**: The Region of Interest (ROI) is determined as the smallest rectangle that bounds the known image area. Figure 4.1 shows an example of a rectangle defining the ROI in an ultrasound image.
2) **SHA-256**: The hash value for the whole image using SHA-256 is calculated. This produces a 256-bit one-way hash value that can be the basis of the watermark.
3) **Embed**: The hash value is embedded into the Region of Non-Interest (RONI) in the LSB. The specific location is not important, as it is known that it will not affect the image under any circumstances.

At receiver site:

1) **Extract watermark**: The watermark is extracted by recovering the LSB from the watermarking area.
2) **Flipping**: In flipping, the LSB in the watermarking area are reset to their original values. This acts as the reversible function and is possible for any image that has an area of known constant. In the case of ultrasound images, this may be easily achieved by resetting all the bits to zero.
3) **SHA-256**: the SHA-256 algorithm is applied to the received image and the hash value computed.
4) **Authentication**: the hash value calculated in step 3 is compared to that extraction in step 2. If found to be the same, the image is authenticated.

4.2.6 Experimental Results

An 800 x 600 pixels ultrasound image was watermarked using the method described in section 4.2.5. The watermarked image was modified using the cloning tools of Adobe Photoshop CS2. The cloning area was around 50x50 pixels and the change may be seen as the image in figure 4.9. Figure 4.8 shows the results of hashing using SHA-256.

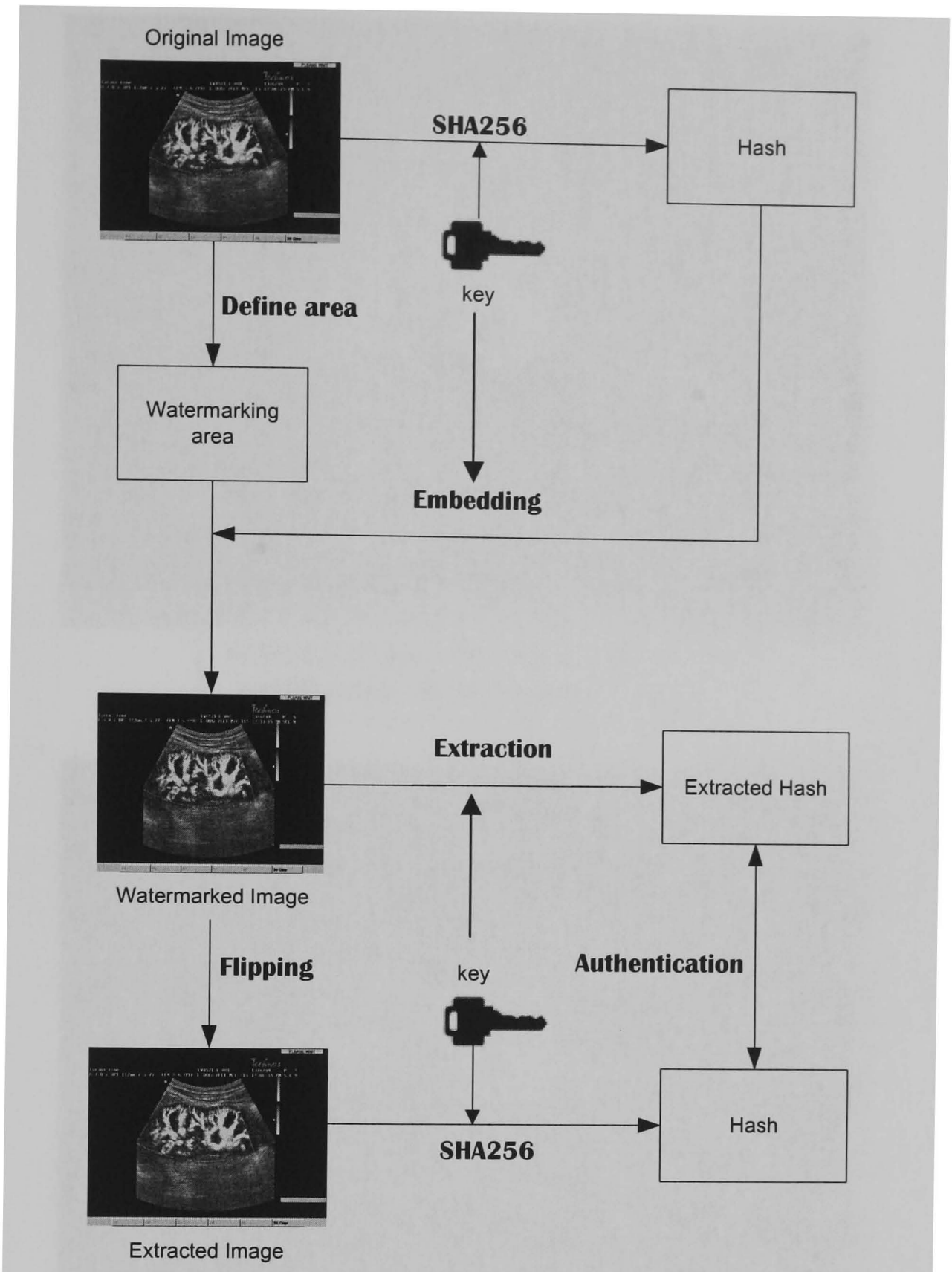
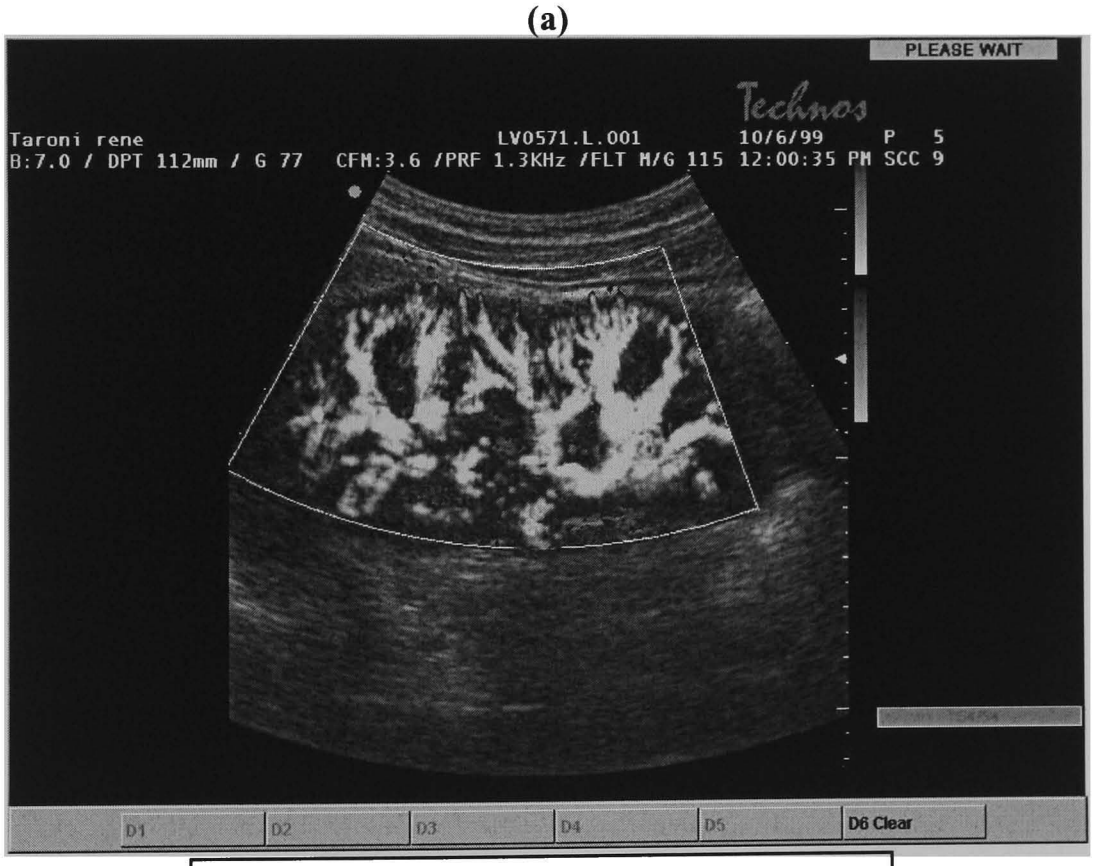


Figure 4.7 Strict Authentication Watermarking (SAW) System

68fb5d32d1fcabd3b22d53da68a30667
0c8ff9a348623072080f8feb28764bfa

(a)



838145455036aa5521d96d5d4657b8
b1d43be9107661ce492015bc999008

(b)

Figure 4.8 (a) Original image and its hash (b) Tampered image and its hash

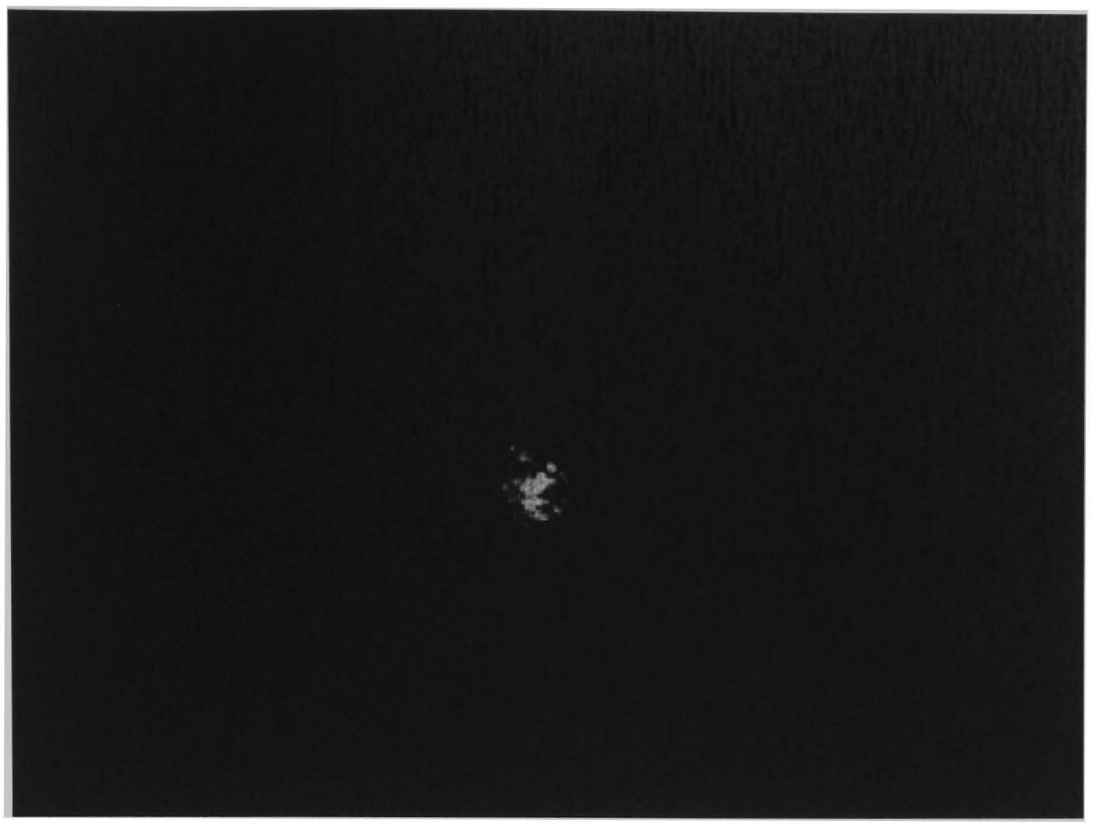


Figure 4.9. Image difference

Two blocks were then watermarked, using one LSB and two LSBs, increasing in the number of bits embedded to determine the capacity of LSB embedding before the recommended PSNR of 32dB was reached. Table 4.3 shows the result of embedding 270kb up to 550kb in the region of non-interest.

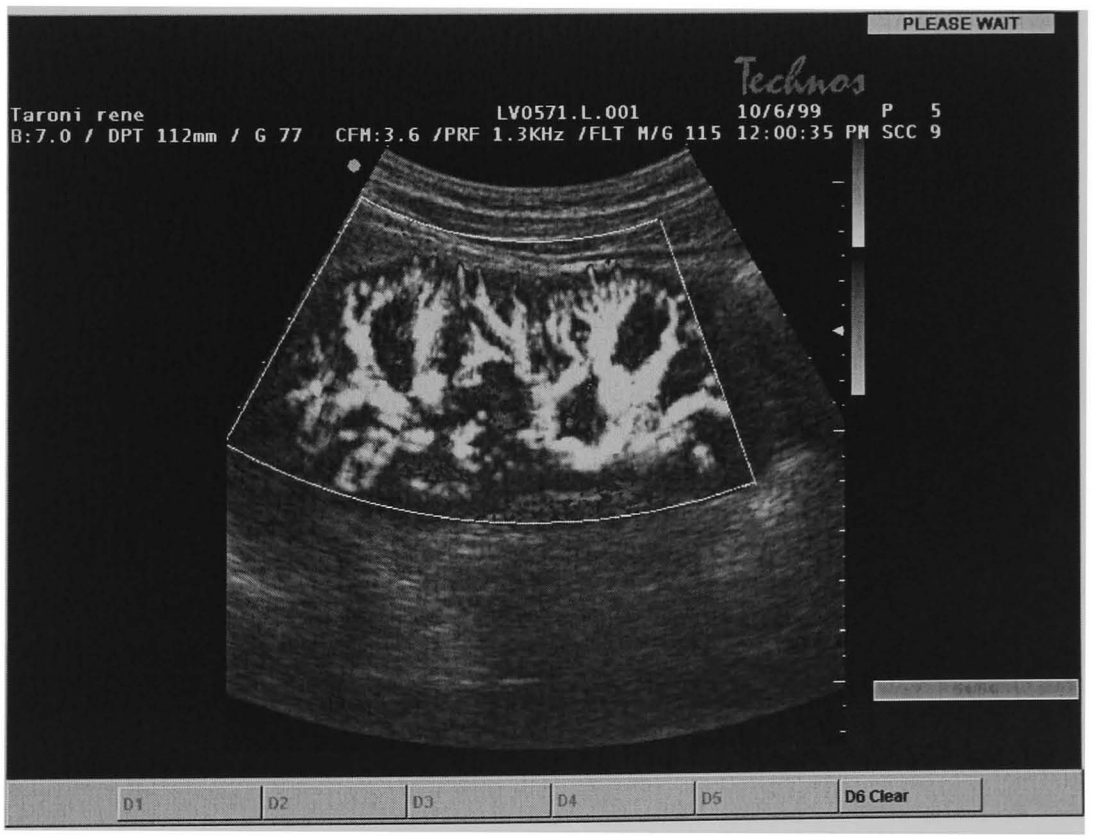


Figure 4.10. Watermarked image with 550kb payload

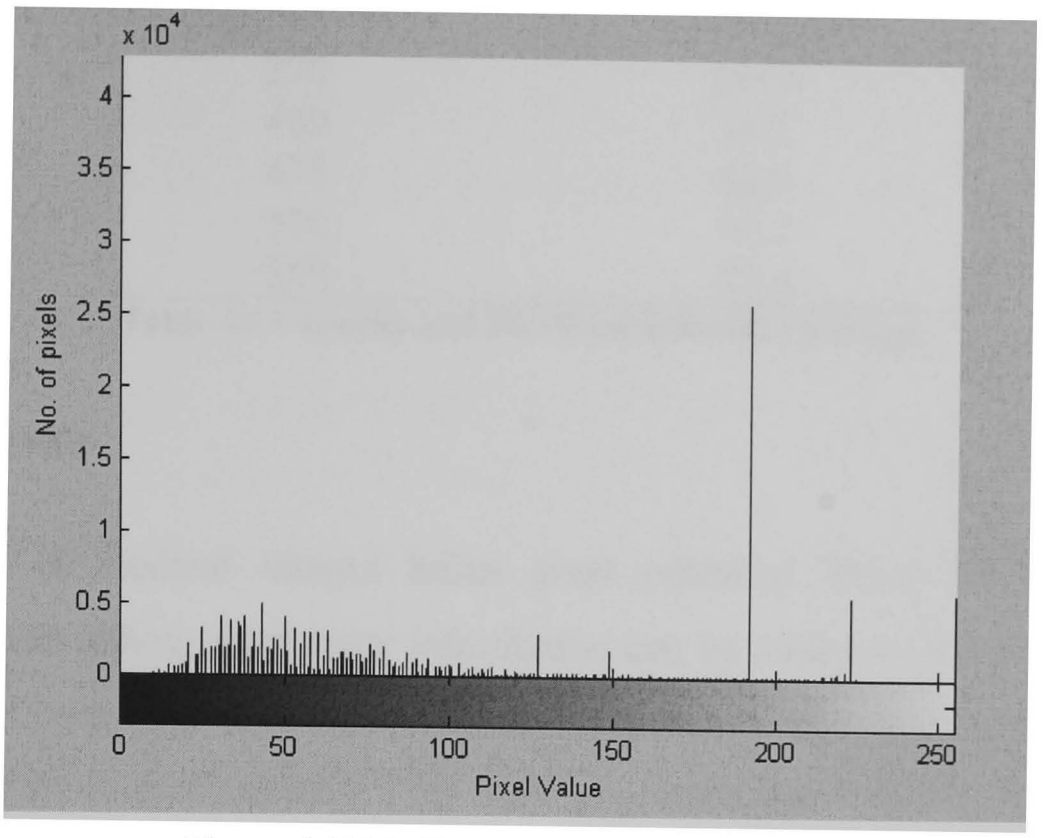


Figure 4.11(a) Histogram of Original Image

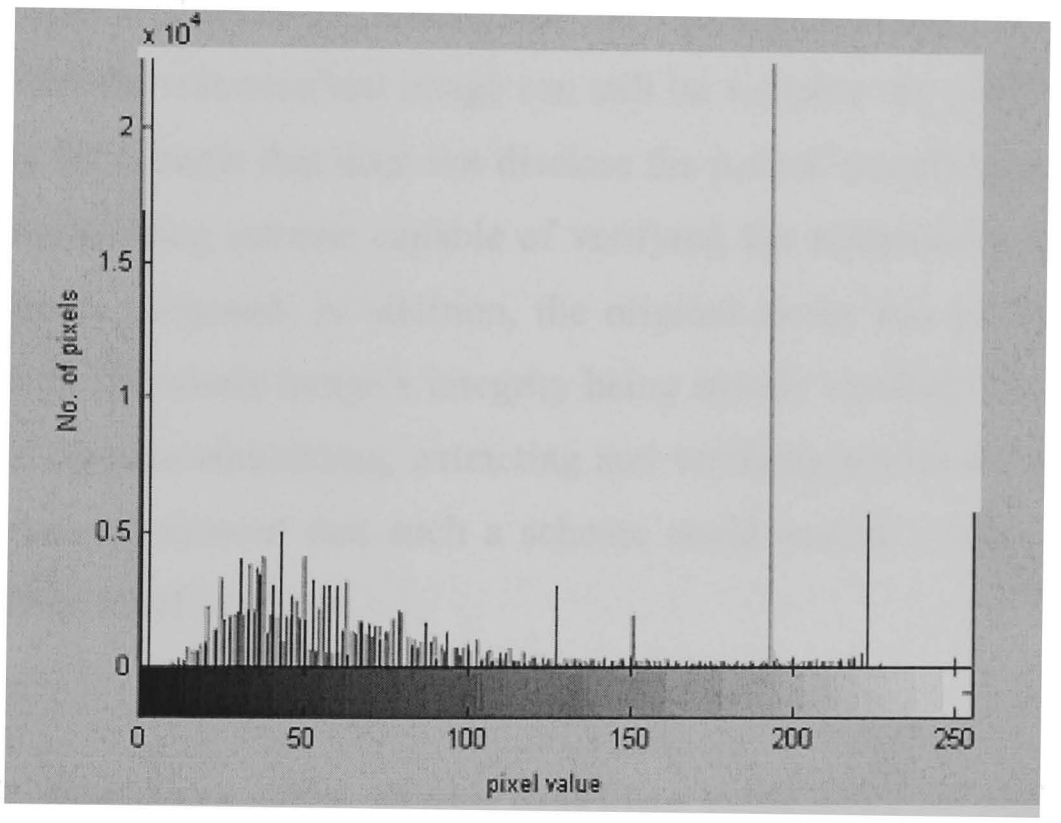


Figure 4.11(b) Histogram of watermarked image (550kb)

Figure 4.11 (a-b) shows the image histogram of the original image and a watermarked image with 550 kb payload. The histogram clearly shows the dramatic increase in the pixels with values 1 and 3, but keeping the remaining pixels exactly the same.

Capacity (kb)	PSNR (dB)
270	249.6
430	51.5
475	42.9
510	31.7
550	27.4

Table 4.3 Capacity and PSNR for 800x600 US Image

4.2.7 Conclusion

Watermarking in medical images holds great potential. From the large capacity available for embedding, a lot more information can be added to the image to make it more secure. Combining cryptography and compression will add security and more information to the limited capacity. The most important aspect regarding watermarking for medical image communications is that the image still conforms to the DICOM image format after watermarking takes place. In keeping distortion level very low, we could ensure that the watermarked image can still be valuable for other purposes, such as a case study for schools that does not disclose the patient's confidential information. A lossless watermarking scheme capable of verifying the authenticity and integrity of DICOM images is proposed. In addition, the original image can be recovered at the receiver site with the whole image's integrity being strictly verified. The watermarking scheme, including data embedding, extracting and verifying procedure were presented. Experimental results showed that such a scheme could embed a large payload while keeping distortion level very low.

4.3 Strict Authentication Watermarking with JPEG Compression (SAW-JPEG)

4.3.1 Image Compression

Image compression seeks to reduce the number of bits required to represent the image information. Two fundamental properties used in image compression are removal of redundancy and reduction of irrelevant content. Irrelevant content may include information not perceived by the viewer, namely the human visual system (HVS). Three types of redundancy may be exploited:

- Spatial redundancy or correlation between neighbouring pixels
- Spectral redundancy or correlation between different frequency bands
- Temporal redundancy or correlation between adjacent frames in a sequence of images (in video applications).

Compression algorithms can be divided into two main groups, lossless and lossy methods. In lossless compression schemes, only the redundancy is exploited, and the image is recorded in a more efficient manner. All the information is retained and so the reconstructed image is numerically identical to the original image. In lossy compression, information deemed irrelevant to the visual perception of the human viewer is discarded and so the compressed image cannot be perfectly reconstructed and distortion is introduced into the reconstructed image.

While lossless compression does not harm a watermarking system in any way (the original data can be perfectly reconstructed), lossy compression methods introduce distortion that has to be taken into account in watermarking applications. Lossy compression techniques are nowadays being commonly used as a means to effect a reduction on the requirement for bandwidth and storage space. It is therefore necessary to study the effects of lossy image compression on watermarking systems.

It should be observed that the design goal of lossy compression systems is opposed to that of watermark embedding systems. The HVS model of the compression system attempts to identify and discard perceptually insignificant information of the image, whereas the goal of the watermarking system is to embed the watermark information without altering the visual perception of the image. An optimal compression or de-noising system would immediately discard any such watermark information. Fortunately, all current compression methods are not optimal and allow watermarking schemes to be devised that will embed watermark information that is robust.

It remains unresolved how lossy compression should best be employed for the storage and transmission of medical images. There is little guidance from the scientific

literature, professional practice standards, regulatory authorities, or the common law. Although lossy compression schemes are included in medical standards such as DICOM, their clinical use is not defined; it is only that the technology is available for use at the discretion of the user or implementer.

There is no good metric by which to judge lossy compression schemes or determine appropriate threshold levels for diagnostic use. Quantitative metrics based on an analysis of the image pixels such as Mean Squared Error (MSE) and Peak Signal-to-Noise Ratio (PSNR) do not correlate well with observers' opinions of image quality, or the measurement of observers' performance when undertaking diagnosis. Metrics based on models of human visual perception are still in their infancy. They have not been thoroughly compared to observer performance for medical applications (Clunie 2000).

Hybrid lossless/lossy compression schemes have been developed for medical applications. These identify regions of images that are determined by some criterion to be of little or no clinical interest. These regions are then either discarded or compressed with greater loss. The remaining regions, which contain the regions of clinical interest, are compressed using a lossless compression scheme. This approach can result in a high compression overall and retain the effective quality of a lossless compression scheme. The difficulty is to determine the areas of clinical interest. There has been work to find automate algorithms, but the only reliable method has been to determine regions defined by physical characteristics Some early CT compression schemes did not encode information outside the circular reconstructed area at all (perimeter coding) and were very effective. However, if such areas are filled with a constant pixel value then most general-purpose lossless image compression schemes perform equally well.

4.3.2 JPEG Compression

JPEG (Wallace 1991) is currently the most frequently used compression algorithm for medical imaging. For example it is included within the DICOM standard. Improved compression algorithms such as JPEG2000, will replace JPEG in time. For the purposes of this work, the watermarking method will focus specifically on JPEG, although the

method should be extensible to other compression schemes based on a block compression scheme.

In this section, we briefly review the JPEG lossy compression standard (Wallace 1991). At the input to the JPEG encoder, the source image, X, is grouped into ρ nonoverlapping 8x8 blocks, X_p. Each block is sent sequentially to the Discrete Cosine Transform (DCT). Instead of representing each 8x8 matrix, we can rewrite it as a 64x1 vector following the "zigzag" order (Wallace 1991). Therefore the DCT coefficients, F_p, of the vector, X_p, can be considered as a linear transformation of X_p with a 64x64 transformation matrix D, such that,

$$F_p = DX_p \tag{4.5}$$

The two-dimensional DCT of an M x N image X is defined as follows:

$$B_{pq} = \alpha_p \alpha_q \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} X_{mn} \cos\frac{\pi(2m+1)p}{2M} \cos\frac{\pi(2n+1)q}{2N}, \quad \begin{matrix} 0 \le p \le M-1 \\ 0 \le q \le N-1 \end{matrix}$$

$$\alpha_p = \begin{cases} 1/\sqrt{M}, & p = 0 \\ \sqrt{2/M}, & 1 \le p \le M \end{cases} \qquad \alpha_q = \begin{cases} 1/\sqrt{N}, & q = 0 \\ \sqrt{2/N}, & 1 \le q \le N \end{cases} \tag{4.6}$$

The values B_{pq} are called the DCT coefficients of X. The DCT is an invertible transform. Each of the 64 DCT coefficients is uniformly quantized with a 64-element quantization table, Q.

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Figure 4.12. JPEG quantization table

In JPEG, the same table is used on all blocks of an image. Quantization is defined as the division of each DCT coefficient by its corresponding quantizer step size, and rounding to the nearest integer:

$$\widetilde{f}_p(v) \equiv IntegerRound(\frac{F_p(v)}{Q(v)}), \tag{4.7}$$

where v = 1…64. In equation (4.7), $\widetilde{f}_p$ is the output of the quantizer. We define $\widetilde{F}_p$, a quantized approximation of F_p, as

$$\widetilde{F}_p \equiv \widetilde{f}_p(v) \cdot Q(v) \tag{4.8}$$

In addition to quantization, JPEG also includes scan order conversion DC differential encoding, and entropy coding.

Inverse DCT (IDCT) is used to convert $\widetilde{F}_p$ to the spatial domain image block $\widetilde{X}_p$

$$\widetilde{X}_p = D^{-1}\widetilde{F}_p \tag{4.9}$$

All blocks are then tiled to form a decoded image frame. Theoretically, the results of IDCT are real numbers. However the brightness of an image is usually represented by

an 8-bit integer from 0 to 255 and thus a rounding process mapping those real numbers to integers is necessary.

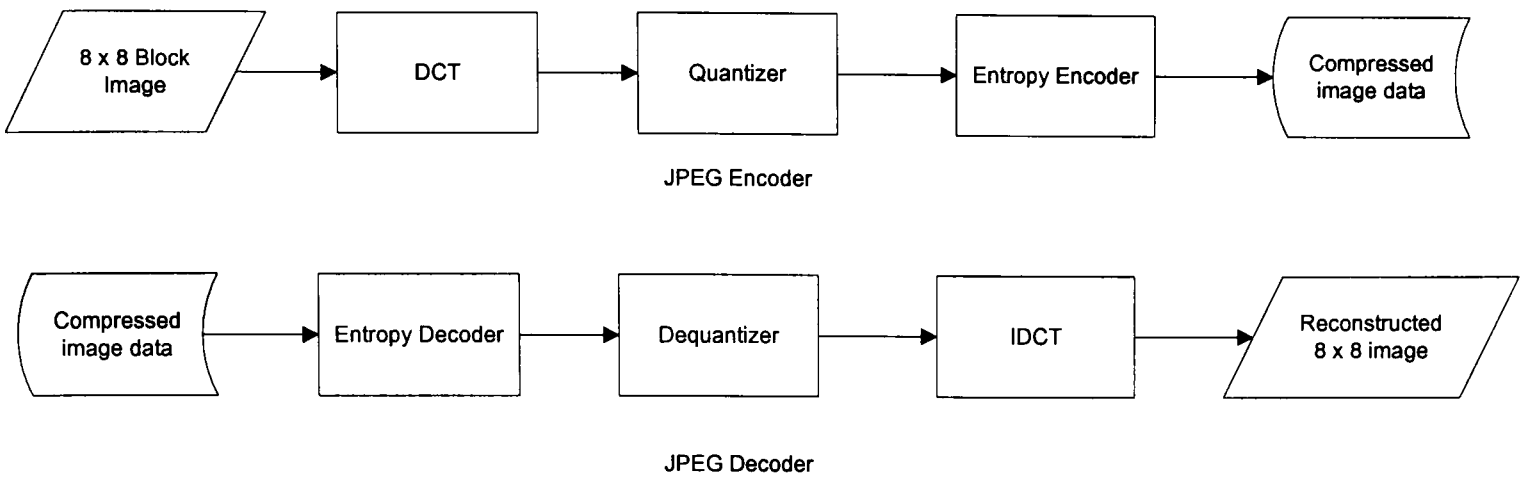


Figure 4.13. JPEG Encoder and decoder

To embed a watermark in the spatial domain, it is necessary to ensure that the embedded watermark will survive JPEG quantization process. JPEG processes images in 8 x 8 blocks, and so the method in which the watermark is embedded should be based on this same block structure. The process may be illustrated by encoding an 8 x 8 sub-image using JPEG. Consider if a '1' is embedded into the whole of the LSB plane of the 8 x 8 block as depicted by figure 4.14.

1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1
1	1	1	1	1	1	1	1

Figure 4.14. '1' bit embedded in 8x8 block

After the DCT transform of the block, figure 4.15 is the result.

8	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

Figure 4.15. DCT Transform of figure 4.14

To survive the quantization process, the value must be preserved through transformation and inverse transformation, that is

$$F_p = \widetilde{F}_p \tag{4.10}$$

To achieve this, $\frac{F_p(v)}{Q(v)}$ must be the integer and have no effect on the rounding process. In particular the DC quantization coefficient should be equal to the dc component in order to preserve an integer result, and all other quantization coefficients should be scaled accordingly. For higher compression rate, to preserve an integer value, the embedded level must be increased, which will naturally have an effect on the quality of the image.

By designing the watermark embedding algorithm around the properties of the compression scheme, it is possible to preserve the watermark values. In this case, a priori knowledge of the quantization algorithm allows the DC coefficient to be unchanged through the compression/decompression process.

4.3.3 Method

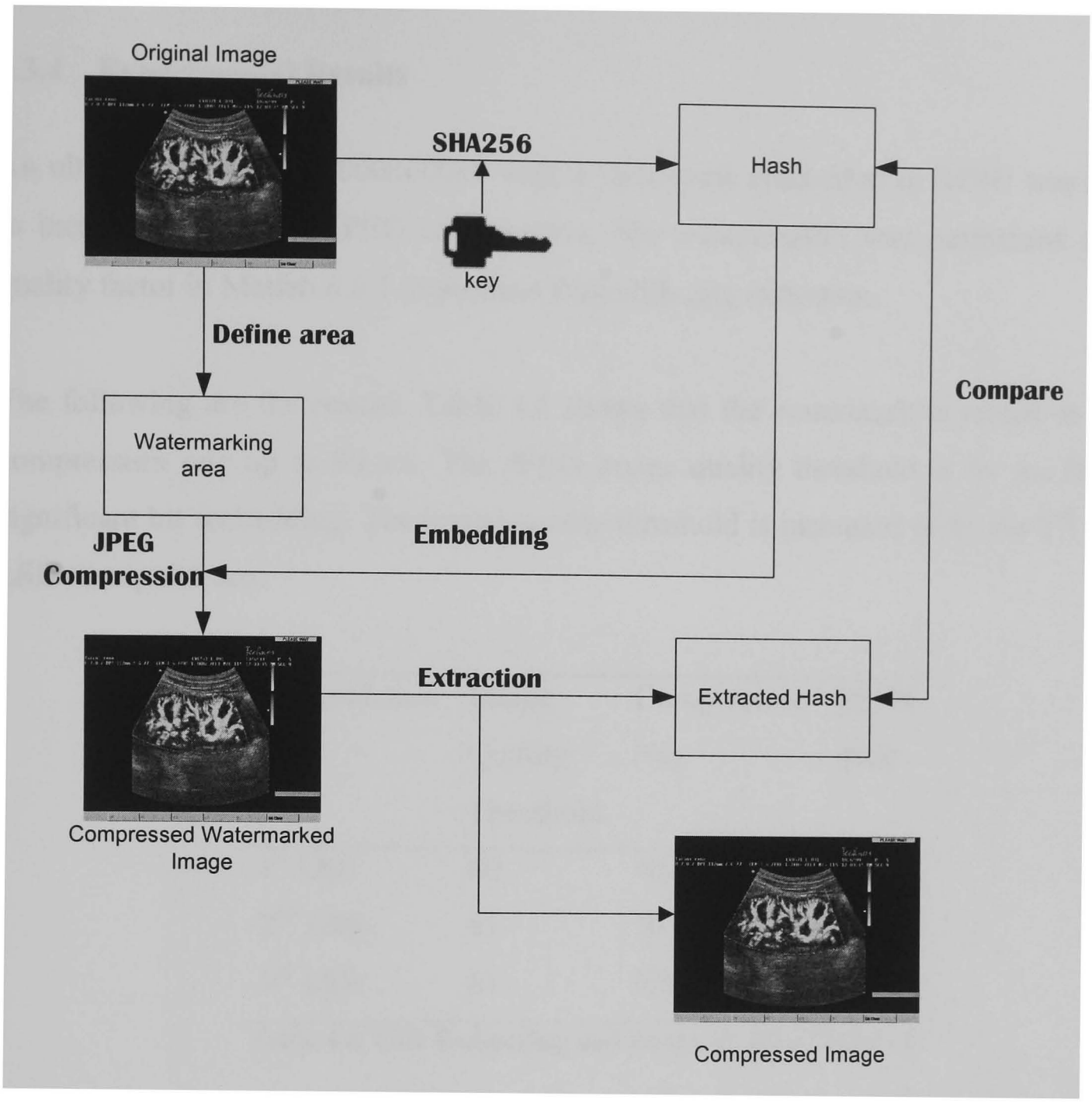


Figure 4.16. Watermarking scheme

The complete process is shown in figure 4.16 and comprises of the following steps:

1) **Define area**: This will define the Region of interest (ROI) where the smallest rectangle is obtained. Please refer to section 4.2.5.

2) **SHA256**: refer to section 4.2.5.

3) **Embedding**: Embed the hash value in the Region of Non-Interest (RONI) in the LSB. Since JPEG uses 8x8 blocks, we try to embed 1-bit in an 8x8 block. We only need 256 8x8 blocks to be able to embed the hash value.

4) **JPEG Compression**: Compression is performed on the watermarked image.

5) **Extraction**: The watermark is recovered from the watermarking area.

6) **Authentication**: The original hash and the extracted hash value are compared.

4.3.4 Experimental Results

An ultrasound image of 800x600x8 with a watermark embedded in RONI was subject to increasing levels of JPEG compression. The compression was performed using a quality factor in Matlab 6.5.1 to produce files with .jpg extension.

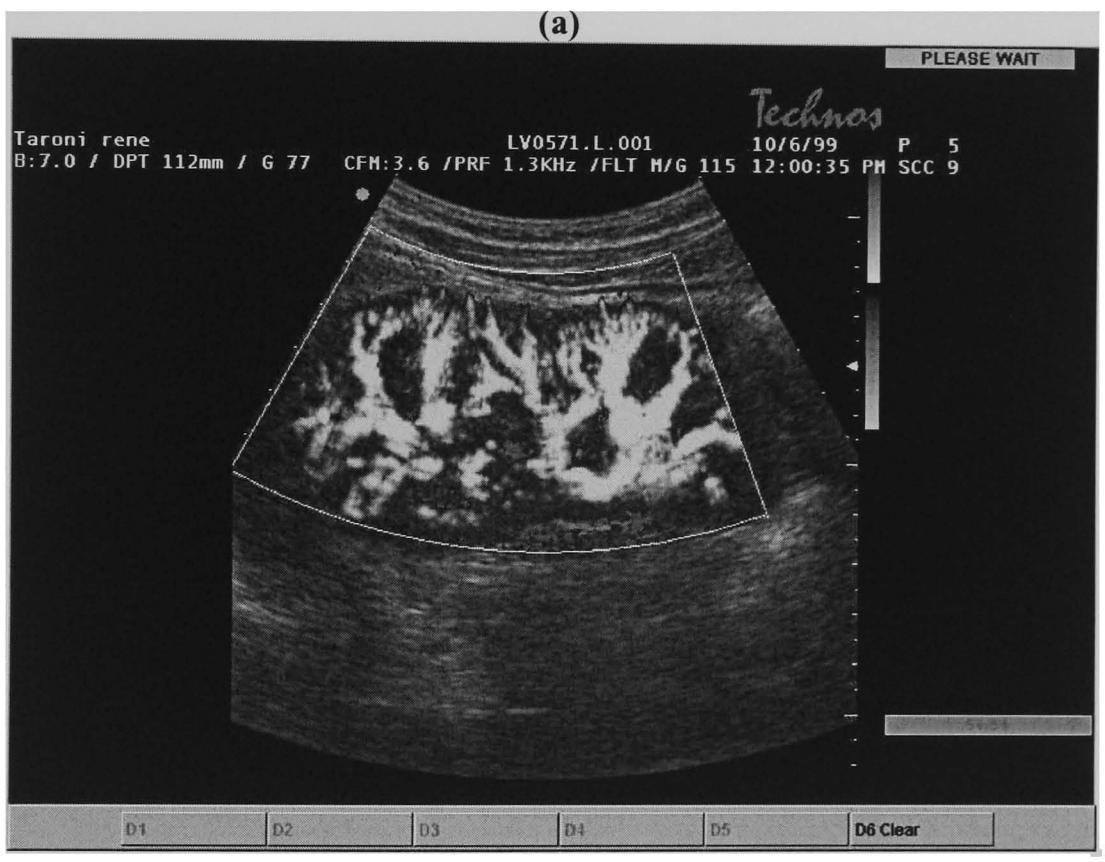
The following are the results. Table 4.4 shows that the watermark is robust to a high compression rate up to 90.6%. The JPEG image quality threshold is 60 for the least significant bit embedding. The image quality threshold is increased to 61 for 2nd and 3rd LSB manipulations.

Manipulation	Image Quality Threshold	Compression (%)	PSNR (dB)
1st LSB	60	90.6	40.75
2nd LSB	61	90.4	40.84
3rd LSB	61	90.4	40.84

Table 4.4. LSB Embedding and Image Quality Threshold

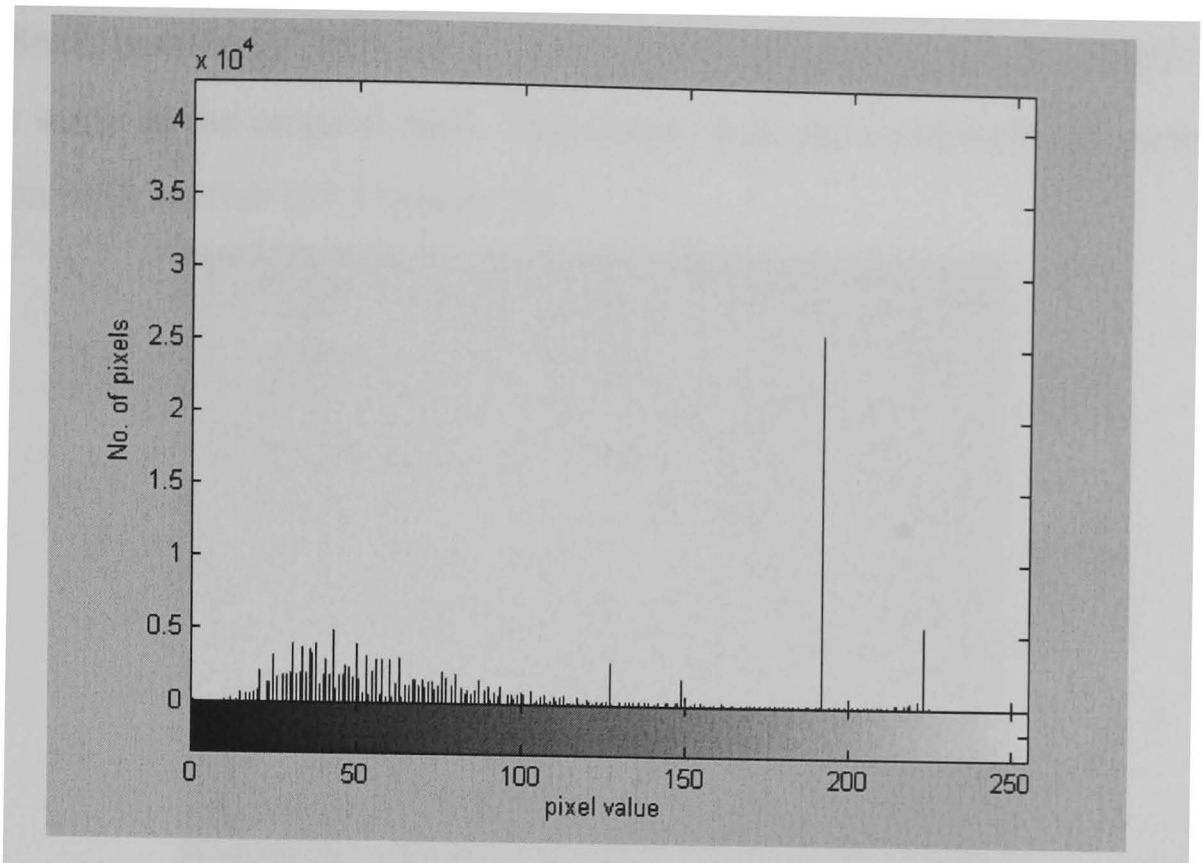
Figure 4.17 shows the original 800x600 US image and the compressed watermarked image with quality 60. This has the effect of changing some pixel values, with a marked effect on areas of abrupt change resulting in the increase of pixel values 2 – 10 (figure 4.18). The effect of adding the watermark is evident by the peaks of pixel value 0 and 1. JPEG loses definition, particularly at high frequencies, which has the effect of low pass or smoothing filter.

(a)

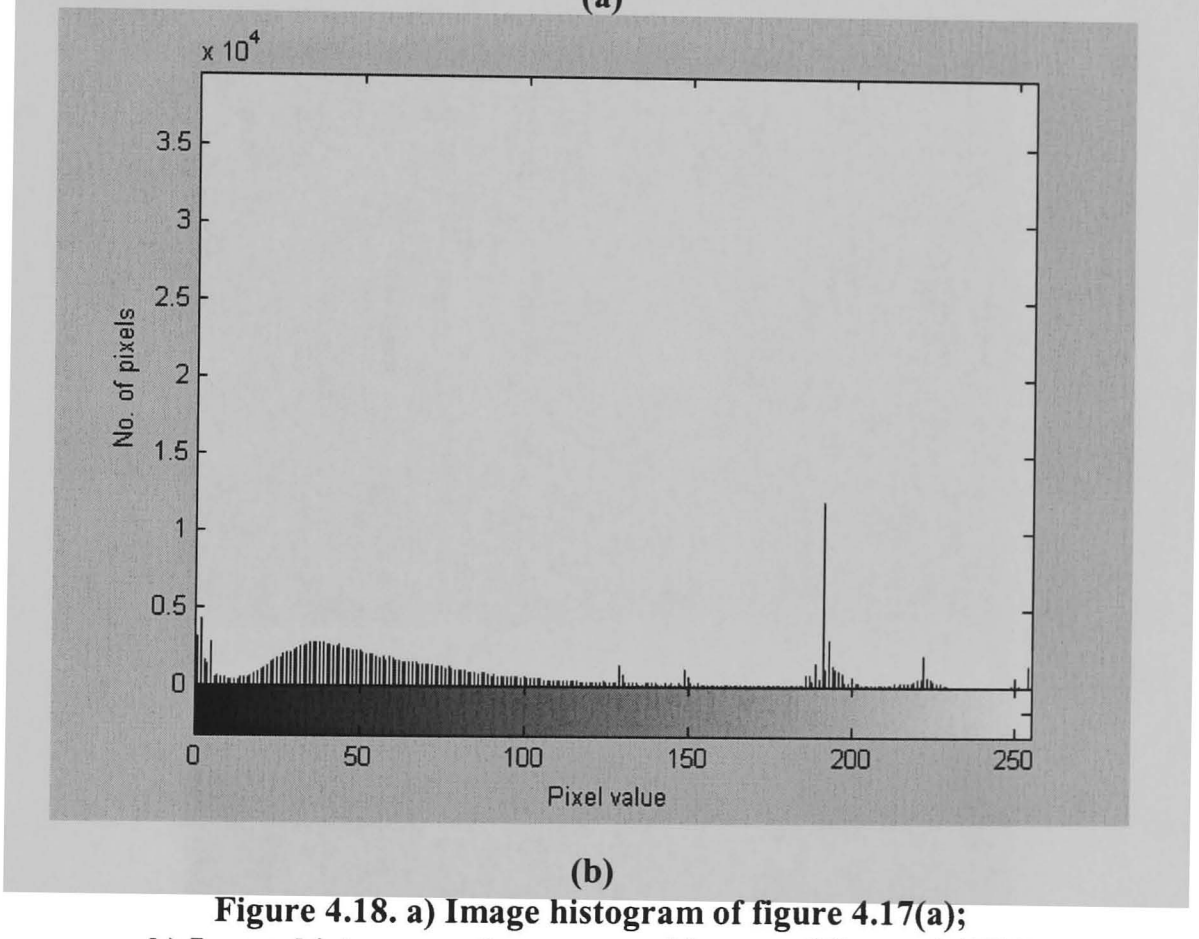


(b)

Fig. 4.17. a) Original 800x600 US image b) compressed watermarked image with quality 60



(a)



(b)

Figure 4.18. a) Image histogram of figure 4.17(a);
b) Image histogram of compressed image of figure 4.17(b)

An 800x600 ultrasound image was watermarked with its hash and then compressed with quality 60 in Matlab 6.5.1. The hash value of the original was recorded as "fcc29cbb8ea81be407cdd93e0326bf2bb68dca3d7872c9b6a033a981e184f989", and the hash value of the compressed image was extracted. Figure 4.19 shows (a) the original 800x600 ultrasound image, (b) the watermarked original image and (c) the watermarked image after compression with quality 60. The extracted hash value was

"fcc29cbb8ea81be407cdd93e0326bf2bb68dca3d7872c9b6a033a981e184f989", and is exactly the same as the original hash. This shows that the watermark can survive JPEG compression with Matlab 6.5.1 quality 60.

(a)

(b)

(c)

Figure 4.19. (a) original image (b) Watermarked image (c) the image after compression

4.3.5 Conclusion

A lossless watermarking scheme is proposed that is robust to lossy JPEG compression and at the same time is able to verify the authenticity and integrity of medical images. The watermarking scheme, including data embedding, extracting and verifying procedure were presented. Experimental results showed that such a scheme could embed and extract the watermark at a high compression rate. Combining cryptography and compression will add security to the medical images. In keeping the distortion level low, we could make sure that the watermarked image can still be valuable for other purposes, such as case studies in schools, but without disclosing a patient's confidential information.