# "Robust Digital Watermarking using Contourlet Transform and DWT and SVD Techniques"

By

**Preeti Sharma**

Under the supervision of

**Mr. Tapan Jain**

May-2014

*Dissertation submitted in partial fulfilment*

*Of the requirement for the degree of*

**MASTER OF TECHNOLOGY**

**IN**

**ELECTRONICS & COMMUNICATION ENGINEERING**

**JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY**

(Established under the Act 14 of Legislative Assembly of Himachal Pradesh)

Waknaghat, P.O. DomeharBani. Teh. Kandaghat, Distt. Solan- 173234(H.P)

Phone: 01792-245367, 245368,245369

Fax-01792-245362

# CERTIFICATE

This is to certify that the work titled **"Robust Digital Watermarking using Contourlet Transform and DWT and SVD Tecnhiques"** submitted by **"Ms. Preeti Sharma"** in the partial fulfillment for the award of degree of Master of Technology (ECE) of Jaypee University of Information Technology, Waknaghat has been carried out under my supervision. This work has not been submitted partially or wholly to any other university or institution for the award of this or any other degree or diploma.

Mr. Tapan Jain

(Assistant Professor)

Department of Electronics and Communication Engineering

Jaypee University of Information Technology (JUIT)

Waknaghat, Solan – 173234, India

(Supervisor)

**JAYPEE UNIVERSITY OF INFORMATION TECHNOLOGY**

(Established under the Act 14 of Legislative Assembly of Himachal Pradesh)

Waknaghat, P.O. DomeharBani. Teh. Kandaghat, Distt. Solan- 173234(H.P)

Phone: 01792-245367, 245368,245369

Fax-01792-245362

# DECLARATION

I hereby declare that the work reported in the M. Tech thesis entitled **"Robust Digital Watermarking using Contourlet Transform and DWT and SVD Techniques"** submitted by **"Ms. Preeti Sharma"** at Jaypee University Of Information Technology, Waknaghat is an authentic record of our work carried out under the supervision of **Mr. Tapan Jain**. This work has not been submitted partially or wholly to any other university or institution for the award of this or any other degree or diploma.

Ms. Preeti Shamra

Department of Electronics and Communication Engineering

Jaypee University of Information Technology (JUIT)

Waknaghat, Solan – 173234, India

# ACKNOWLEDGEMENT

The completion of my thesis dedicates a solemn gratitude to all those who have helped me in not falling astray and focusing on the right parameters to accomplish my goal. I am sincerely obliged to my guide Mr. Tapan Jain who has been a total support all throughout my work and has enriched me with ideas. I extend my heartfelt thank you to my friend Pallvi Chawla who has boosted me to always step forward and strife hard in all what I do. I would also like to applaud and gratify the faculty members of JUIT who have in some or the other way helped in grooming my personality in the field of technology.

The last words of heart wrenching gratitude is extended to my parents, Mr. Anil Kumar Sharma and Mrs. Anjana Sharma, who have made my breath in this world possible and have always stood strong in all my falls and taught me that it is more about getting up again rather than falling down.

Preeti Sharma

M.Tech (ECE)

# ABSTRACT

Watermarking is a data hiding technique and is highly reliable for copyright protection in case of digital data transmission. Contourlet Transform is used for watermarking purposes as in order to extract the directionalities and geometric shapes. It is highly efficient in terms of non-linear approximations thereby quite reliable for image compression. DWT and SVD techniques though provide a high frequency resolution but lack in the fulfilment of directions and geometric shapes, which is thus provided by Contourlet Transform. In my thesis, i have proposed the differences in the DWT and SVD techniques than Contourlet Transform and then combined their benefits in order to yield a transform with a better PSNR and more directionalities and geometric shapes. The technique for watermarking proposed in the thesis provides better horizontal and vertical directions and also provides high robustness. Therefore, the proposed scheme of watermarking is tested and proven to be highly reliable and robust.

# Table of Contents

# LIST OF FIGURES and IMAGES

# LIST OF TABLES

# ACRONYMS

| | |
|---|---|
| SVD | Singular Value Decomposition |
| DWT | Discrete Wavelet Transform |
| PSNR | Peak Signal to Noise Ratio |
| DCT | Discrete Cosine Transform |
| CWT | Continuous Wavelet Transform |

# INDEX WORDS

Data Hiding, Watermarking, Frequency domain transforms, SVD, DWT, Contourlet Transform.

# PUBLICATION

Ms. Preeti Sharma and Mr. Tapan Jain "**Robust Digital Watermarking using SVD and DWT Technique"** 4th IEEE International Advance Computing Conference – IACC, 2014.

# CHAPTER 1

# INTRODUCTION

## 1.1 History

Although the art of paper making was invented in China over a thousand years earlier, paper watermarks did not appear until about 1282, in Italy. The marks were made by adding thin wire patterns to the paper molds. The paper would be slightly thinner where the wire was and hence more transparent. The meaning and purpose of the earliest watermarks are uncertain. They may have been used for practical functions such as identifying the molds on which sheets of papers were made, or as trademarks to identify the paper maker. On the other hand, they may have represented mystical signs, or might simply have served as decoration.[1]

By the eighteenth century, watermarks on paper made in Europe and America had become more clearly utilitarian. They were used as trademarks, to record the date the paper was manufactured and to indicate the sizes of original sheets. It was also about this time that watermarks began to be used as anti counterfeiting measures on money and other documents.

The term watermark seems to have been coined near the end of the eighteenth century and may have been derived from the German term *wassermarke* (though it could also be that the German word is derived from the English). The term is actually a misnomer, in that water is not especially important in the creation of the mark. It was probably given because the marks resemble the effects of water on paper.[2]

## 1.2 Watermarking

The sudden increase in watermarking interest is most likely due to the increase in concern over copyright protection of content. Watermarking has been considered for many copy prevention and copyright protection applications. In copy prevention, the watermark may be used to inform software or hardware devices that copying should be restricted. In copyright protection applications, the watermark may be used to identify the copyright holder and ensure proper payment of royalties.[3]

Digital watermarking is a key ingredient to copyright protection. It provides a solution to illegal copying of digital material and has many other useful applications such as broadcast monitoring and the recording of electronic transactions.[4]

## 1.3 Techniques of Watermarking:

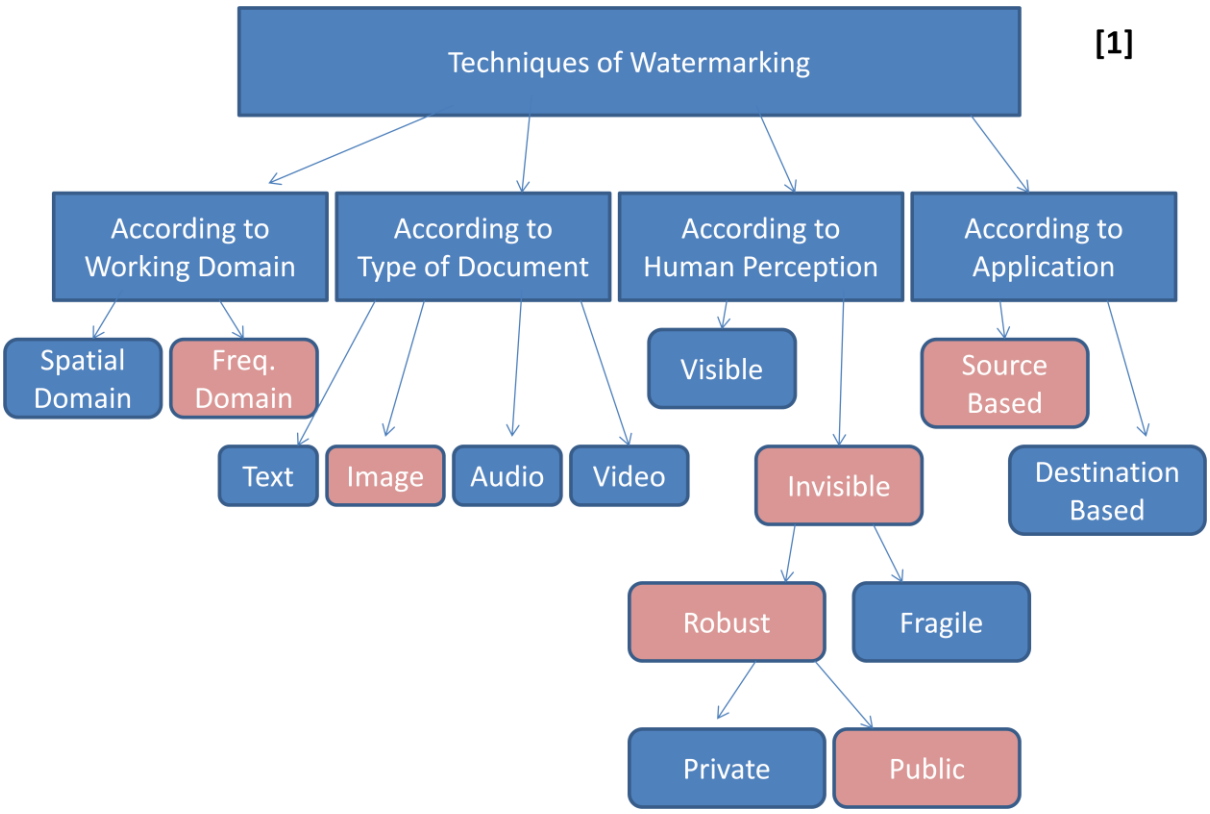



Figure 1.1: Techniques of Watermarking

Human perception is also used as a criterion to classify the watermarking techniques. Visible and invisible watermarks are of this type. Logos are the examples of the visible watermarks that indicate the owner of the content. An usual way of visible image watermarking is to print "©date,owner" mark onto the image. One disadvantage of visible watermarks is that it can be easily removed from the digital cover image. Invisible watermarks alter the media in a way that they are perceptually unnoticeable. They can only be detected by using an appropriate detection method. They identify the owner of the digital media. Unlike visible watermarks, the invisible watermarks could not be removed from the media because they became an integral component of the content after being embedded. However, they can be made undetectable by some manipulations and distortions called "attacks". The watermark, ideally, must be resilient to all possible attacks. Proof of ownership is another application area for invisible watermarks; however, it needs a higher level security than owner identification. Craver et al. proposed a watermarking scheme that can be applied on a watermarked image, to allow multiple claims of rightful ownership. The two types of invisible watermarks are robust and fragile watermarks. Purpose of the robust algorithms is the endurance of watermark after possible distortions such as possible compressions, filtering and noise additions. However, the fragile watermarks are used to detect if there is any manipulation or modification on the digital content. These modifications would change or destroy the watermark. Fragile watermarks can be used for content authentication such as trustworthy camera. A watermark is embedded into the frame when it is captured by the camera. The watermark will be lost if any altering made so verifying if the frame is the original captured one or not. The invisible robust watermarks are divided into two categories as private and

public watermarks, as described in previous section. The private algorithms need the original content to detect the watermark where the public watermarks do not need.
According to the applications, the watermark could be classified as source based and destination based watermarks. In the source based algorithms, all the copies are watermarked with a unique watermark and used for ownership identification or authentication. The watermark identifies the owner of the content. However, the destination based watermarks (fingerprints) are embedded individually to each copy and used to mark out the buyer in the case of an unlawful operation. Fingerprints can be used for broadcast monitoring. A unique watermark is embedded into each video or audio-clip before broadcasting. Automated computers monitor the broadcast and detect when and where each clip is appeared. Another application area of the watermarks is copy control. The digital media can be copied without sacrificing quality. To check this, a watermark can be inserted in a media such that a recorder would not copy it if it detects a watermark that indicates copying is illegal. However, this could be successful if all the manufactured recorders can implement watermark detection algorithms.

## 1.4 Applications of Watermarking

Watermarking can be used in a wide variety of applications. In general, if it is useful to associate some additional information with a work, this metadata can be embedded as a watermark. There are seven proposed or actual watermarking applications: broadcast monitoring, owner identification, proof of ownership, transaction tracking, authentication, copy control, and device control.[5]

### 1.4.1 Broadcast Monitoring:

There are a number of potential problems with implementing passive monitoring systems. First, comparing the received signals against a database is not trivial. In principle, dividing the signals into recognizable units, such as individual frames of video, and search for the min the database. However, each frame of video consists of several million bits of information, and it would be impractical to use such a large bit sequence as an index for a database search. Thus, the system must first process the received signals into smaller signatures that are rich enough to differentiate between all possible Works yet small enough to be used as indices in a database search. Defining these signatures are difficult. Furthermore, broadcasting generally degrades the signals, and this degradation might vary overtime, with the result that multiple receptions of the same Work at different times might lead to different signatures. This means that the monitoring system cannot search for an exact match in its database. Instead, it must perform a nearest neighbour search, which is known to be substantially more complex. Because of the difficulty of deriving meaningful signatures.

Watermarking is an obvious alternative method of coding identification information for active monitoring. It has the advantage of existing within the content itself, rather than exploiting a particular segment of the broadcast signal, and is therefore completely compatible with the installed base of broadcast equipment, including both digital and analog transmission. The primary

disadvantage is that the embedding process is more complicated than placing data in the VBI or in file headers. There is also a concern, especially on the part of content creators, that the watermark may degrade the visual or audio quality of the Work. Nevertheless, there are a number of companies that provide watermark based broadcast monitoring services.[6]

### 1.4.2 Owner Identification*:*

Through 1988, if copyright holders wanted to distribute their Works without losing any rights, they had to include a copyright notice in every distributed copy. After 1988, this was changed so that the copyright notice is now no longer required. However, if a Work that is protected by copyright is misused, and the courts choose to award the copyright holder damages, that award can be significantly limited if a copyright notice of acceptable form and placement was not found on the distributed material. The exact form of the copyright notice is important. For visual Works, it must say either "Copyright date owner," "c・ date owner," or "Copr. Date owner." Furthermore later as the time passed by when Digimarc's detect or recognizes a watermark, it contacts a central database over the Internet, and uses the watermark message as a key to find contact information for the image's owner. At present, given that the exact form of a copyright notice holds such legal significance, a copyright notice in a watermark probably would not suffice as an alternative to including the standard "c・" notice. However, the system does make it easier for honest people to find out who they should contact about using an image.[6]

### 1.4.3 Proof of Ownership*:*

It is enticing to try to use watermarks not just to identify copyright ownership but to actually prove ownership. This is something a textual notice cannot do, because it can be so easily forged. To achieve the level of security required for proof of ownership, it is probably necessary to restrict the availability of the detector. When an adversary does not have a detector, removal of a watermark can be made extremely difficult. For example if we consider two people with two different image named Alice and Bob therefore, when Alice and Bob go before the judge, Alice would produce her original copy of the image. Her original, and the disputed copy, would be entered into the watermark detector, and the detector would detect Alice's watermark. However, even if Alice's watermark cannot be removed, Bob might be able to undermine her. As pointed out by Craver et al. Bob, using his own watermarking system, might be able to make it appear as though his watermark were present in Alice's original copy of the image. Thus, a third party would be unable to judge whether Alice or Bob had the true original. This problem can be solved if we make a slight change in the problem statement. Instead of trying to directly prove ownership by embedding an "Alice owns this image" watermark message in it, we will instead try to prove that one image is derived from another. Such a system provides indirect evidence that it is more

likely the disputed image is owned by Alice than by Bob, in that Alice is the one who has the version from which the other two were created. The evidence is similar to that provided if Alice were to produce the negative from which the image was created, except that it is stronger, in that Bob can fabricate a negative but cannot fabricate a fake original that passes our test.[6]

### 1.4.4 Transaction Tracking*:*

In this application of watermarking, the watermark records one or more transactions that have taken place in the history of the copy of a Work in which it is embedded. For example, the watermark might record the recipient in each legal sale or distribution of the Work. The owner or producer of the Work would place a different watermark in each copy. If the Work were subsequently misused (leaked to the press or redistributed illegally), the owner could find out who was responsible.[6]

### 1.4.5 Content Authentication:

It is becoming easier and easier to tamper with digital Works in ways that are difficult to detect. For example, consider a image authentication system that stores the metadata in a JPEG header field. If the image is converted to another file format that has no space for a signature in its header, the signature will be lost. When a signature is lost, the Work can no longer be authenticated. A preferable solution might be to embed the signature directly into the Work using watermarking. Epson offers such a system as an option on many of its digital cameras. We refer to such an embedded signature as anauthentication mark. Authentication marks designed to become invalid after even the slightest modification of a Work are called fragile watermarks. The use of authentication marks eliminates the problem of making sure the signature stays with the Work. A different type of information that might be learned from examining a modified authentication mark is whether or not lossy compression has been applied to a Work. The quantization applied by most lossy compression algorithms can leave tell tale statistical changes in a watermark that might be recognizable. Conversely, we might not care about whether a Work was compressed, and might only be concerned with whether more substantial changes were made. This leads to the field of semi-fragile watermarks and signatures, which survive minor transformations, such as lossy compression, but are invalidated by major changes.[6]

### 1.4.6 Copy Control:

Most of the applications of watermarking discussed so far have an effect only after someone has done something wrong. In the copy control application, we aim to prevent people from making illegal copies of copyrighted content. The first and strongest line of defense against illegal copying is encryption. By encrypting a Work according to a unique key, we can make it unusable to anyone who does not

have that key. The key would then be provided to legitimate users in a manner that is difficult for them to copy or redistribute. There are three basic ways an adversary might try to overcome an encryption system. The first, and most difficult, is to decrypt the data without obtaining a key. This usually involves some form of search, in which the adversary exhaustively tries decrypting the signal with millions of keys. If the cryptographic system is well designed, the adversary will have to try every possible key. This is impractical if the keys are longer than 50 bits or so. An easier approach for the adversary is to try to obtaina valid key. This might be done by reverse engineering hardware or software that contains the key. Because watermarks are embedded in the content itself, they are present in every representation of the content and therefore might provide a better method.[6]

## 1.5 Properties of Watermarking*:*

Watermarking systems can be characterized by a number of defining properties. The relative importance of each property is dependent on the requirements of the application and the role the watermark will play. In fact, even the interpretation of a watermark property can vary with the application. Properties typically associated with a watermark embedding process: effectiveness, fidelity, and payload. Properties typically associated with detection: blind and informed detection, false positive behavior, and robustness. Then exttwo properties, security and the use of secret keys, are closely related in that the use of keys is usually an integral part of any security feature inherent in a watermarking scheme. The section concludes with a discussion of the various costs associated with both watermark embedding and watermark detection.[7]

### 1.5.1 Embedding Effectiveness:

We define a watermarked Work as a Work that when input to a detect or results in a positive detection. With this definition of watermarked Works, the effectiveness of a watermarking system is the probability that the output of the embedder will be watermarked. In other words, the effectiveness is the probability of detection immediately after embedding. This definition implies that a watermarking system might have an effectiveness of less than 100%. Although 100% effectiveness is always desirable, this goal often comes at a very high cost with respect to other properties.[7]

### 1.5.2 Fidelity:

We may define the fidelity of a watermarking system as the perceptual similarity between the unwatermarked and watermarked Works at the point at which they are presented to a consumer.[7]

## 1.5.3 Data Playload:

Data payload refers to the number of bits a watermark encodes within a unit of time or within a Work.[7]

## 1.5.4 Blind or Informed Detection:

In other applications, detection must be performed without access to the original Work. Consider a copy control application. Here, the detector must be distributed in every consumer recording device. Having to distribute the unwatermarked content to every detector would not only be impractical, it would defeat the very purpose of the watermarking system.[7]

## 1.5.5 False Positive Behaviour:

Equivalently, we can discuss the probability that a false positive will occur in any given detector run. There are two subtly different ways to define this probability, which are often confused in the literature. They differ in whether the watermark or the Work is considered the random variable. In the first definition, the false positive probability is the probability that given a fixed Work and randomly selected watermarks the detector will report that a watermark is in that Work. The watermarks are drawn from a distribution defined by the design of a watermark generation system. Typically, watermarks are generated by either a bit encoding algorithm or by a Gaussian, independent random number generator. In many cases, the false positive probability, according to this first definition, is actually independent of the Work, and depends only on the method of watermark generation. In these conditions, the false positive probability is the probability that given a fixed watermark and randomly selected Works the detector will detect that watermark in a Work. The distribution from which the Work is chosen is highly application dependent. Natural images, medical images, graphics, music videos, and surveillance video all have very different statistics. The same is true of rock music, classical music, and talk radio. Moreover, while these distributions are different from one another, they are also likely to be very different from the statistics of the watermark generation system. Thus, false positive probabilities based on this second definition can be quite different from those based on the first definition.[7]

## 1.5.6 Robustness:

Robustness refers to the ability to detect the watermark after common signal processing operations. Not all watermarking applications require robustness to all possible signal processing operations. Rather, a watermark need only survive the common signal processing operations likely to occur between the time of embedding and the time of detection. In some cases, robustnesss may become completely irrelevant, or even undesirable. In fact, an important branch of watermarking research focuses on fragile watermarks. A fragile watermark is one designed so that it is not robust.[7]

## 1.5.7 Security:

This is usually the case when a watermark is used to provide enhanced functionality to consumers. However, for applications that do require some level of security it is important to understand the distinctions between these types of attack. Unauthorized removal refers to attacks that prevent a Work's watermark from being detected. It is common to distinguish between two form so for authorized removal: elimination attacks and masking attack. Intuitively, elimination of a watermark means that an attacked Work cannot be considered to contain a watermark at all. Masking of a watermark means that the attacked Work can still be considered to contain the watermark, but the mark is undetectable by existing detectors. Unauthorized detection, or passive attacks, can be broken down into three levels of severity. The most severe level of unauthorized detection occurs when an adversary detects and deciphers an embedded message. This is the most straight forward and comprehensive form of unauthorized reading. A less severe form of attack occurs when an adversary can detect watermarks, and distinguish one mark from another, but cannot decipher what the marks mean. Because watermarks refer to the Works in which they are embedded, the adversary might be able to divine the meanings of the marks by comparing them to their cover Works. The least severe form of attack occurs when an adversary is able to determine that a watermark is present, but is neither able to decipher a message nor distinguish between embedded messages. In general, passive attacks are of more concern in steganography than in watermarking, but there are watermarking applications in which they might be important.[7]

## 1.5.8 Secret Key:

The purpose of security key should have a well designed chipper and a well-designed cipher should meet the following standards:

a. Knowledge of the encryption and decryption algorithms should not compromise the security of the system.
b. Security should be based on the use of keys.
c. Keys should be chosen from a large key space so that searching over the space of all possible keys is impractical.

It is desirable to apply the same standards to watermarking algorithms. However, the security requirements for watermarks are often different from those for ciphers. Therefore, we cannot simply adapt cryptographic methods to watermarking. Cryptography is only concerned with the prevention of unauthorized reading and writing of messages. It can therefore be used in watermarking to prevent certain forms of passive attack and forgery, but it does not address the issue of watermark removal. Ideally, it should not be possible to detect the presence of a watermark in a Work without knowledge of the key, even if the watermarking algorithm is known. Further, by restricting knowledge of the key to only a trusted group (i.e., by preventing an adversary from learning the key), it should become extremely difficult, if not

impossible, for an adversary to remove a watermark without causing significant degradation in the fidelity of the cover Work. Because the keys used during embedding and detection provide different types of security from those used in cryptography, it is often desirable to use both forms of key in a watermarking system. That is, messages are first encrypted using one key, and then embedded using a different key. To keep the two types of keys distinct, we use the terms cipher key and watermark key, respectively.[7]

### 1.5.9 Modification and Multiple Watermarks:

When a message is embedded in a cover Work, the sender may be concerned about message modification. Whereas the ability to modify watermarks is highly undesirable in some applications, there are others in which it is necessary. For example, American copyright law grants television viewers the right to make a single copy of broadcasted programs for time-shifting purposes (i.e., you are permitted to make a copy of a broadcast for the non commercial purpose of watching that broadcast at a later time). However, you are not permitted to make a copy of this copy. Thus, in copy control applications, the broadcasted content may be labelled copy-once and, after recording, should be labelled copy-no-more. Finally, each website might embed a unique watermark in each Work it sells for the purpose of uniquely identifying each purchaser.[7]

### 1.5.10 Cost:

The economics of deploying watermark embedders and detectors can be extremely complicated and depends on the business models involved. From a technological point of view, the two principal issues of concern are the speed with which embedding and detection must be performed and the number of embedders and detectors that must be deployed. Other issues include whether the detectors and embedders are to be implemented as special-purpose hardware devices or as software applications or plug-ins. In broadcast monitoring, both embedders and detectors must work in (atleast) real time. This is because the embedders must not slow down the production schedule, and the detectors must keep up with rea time broadcasts. On the other hand, a detector for proof of ownership will be valuable even if it takes days to find a watermark. Such a detector will only be used during ownership disputes, which are rare, and its conclusion about whether the watermark is present is important enough that the user will be willing to wait. Furthermore, different applications required for different numbers of embedders and detectors. Many watermarking systems are designed to carry very small data payloads (on the order of 8 bits or fewer), with very high robustness, very low false positive probability, and or very low cost. Such systems typically employ codes that cannot be expanded to 80 bits. Furthermore, the tests performed by Stirmark are not critical to many applications that do not have stringent security requirements. They also do not represent a

comprehensive test of the security required in applications in which an adversary is likely to have a watermark detector.[7]

## 1.6 Data Hiding Techniques:

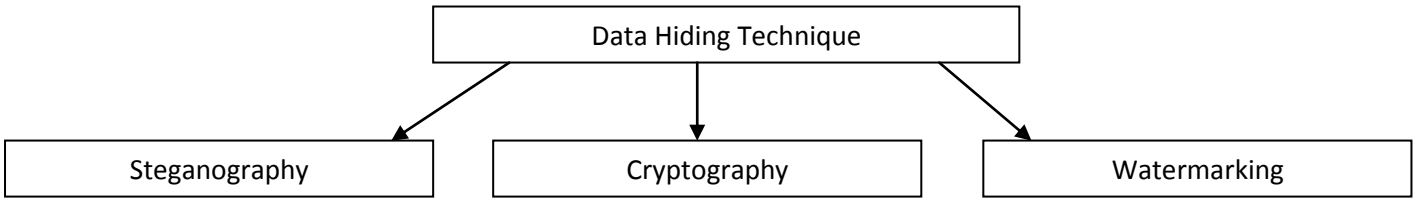



Figure 1.2: Data Hiding Techniques

1.6 Difference between Steganography, Cryptography and Watermarking

### 1.6.1 Comparison of Steganography and Cryptography:[8-10]

- Steganography and Cryptography are closely related.
- Cryptography scrambles messages so it can't be understood. Steganography on the other hand, hide the message so there is no knowledge of the existence of the message.
- With Cryptography, comparison is made between portions of the plaintext and portions of the cipher text. In Steganography, comparison may be made between the cover-media, the stego-media, and possible portions of the message.
- The end result in Cryptography is the cipher text, while the end result in Steganography is the stego-media.

### 1.6.2 Comparison of Steganography and Watermarking:[8-10]

- Steganography performs message hiding such that an attacker cannot detect the presence of the message in the image/audio/video.
- Watermarking hides the message such that an attacker cannot tamper with the message contained within the image/audio/video.
- Steganography methods are in general not robust i.e. the hidden information cannot be recovered after data manipulation.
- Watermarking as opposed to Steganography has the additional notion of robustness against attacks.

### 1.6.3 Comparison of Cryptography and Watermarking:[8-10]

- Cryptography only provides security through encryption and decryption. However, encryption cannot help the seller monitor how a legitimate customer handles the content after decryption. So there is no protection after decryption.
- Watermarking can protect the content even after decoding.
- Cryptography is only about protecting the content of the message.
- Watermarks are inseparable from the cover in which they are embedded so in addition to protecting content watermarking provide many other applications like, copyright protection, ID card security, etc.
- Cryptographic system is broken when the attacker can read the secret message.
- Watermarking system has 2 stages to break through:
    1. The attacker can detect that watermarking has been made.
    2. The attacker is able to read, modify or remove the hidden message.

### 1.6.4 Combining Steganography and Cryptography:[8-10]

The message can also be encrypted before it is hidden inside a cover message. This provides a double layer of protection. To begin with, encryption may make the existence of the message even more difficult to detect, due to the fact that some encryption techniques cause the patterns of the characters in the encrypted version to be more random than in the original version. In addition, even if the existence of the encrypted message is detected, it is unlikely that an eavesdropper will be able to read the message.

# CHAPTER 2.

# TYPES OF ENCODING

## 2.1 Spatial Watermarking (spatial domain)[11-14]

- Low level Encoding.
- Easily attacked.
- Use of Image Analysis Operation, e.g. Edge Detection/Color Separation.

## 2.2 Spectral Watermarking (frequency domain)[11-14]

- Many types due to variety of transforms.
- Adjustments made in frequency domain.
- More robust.
- Decomposition of Image.
- Addition of Watermark.
- Re-composition of Image.

## 2.3 Frequency domain Implementation:

### 2.3.1 Discrete Cosine Transform:

Similar to discrete fourier transform (DFT), discrete cosine transform (DCT) is a function that maps the input signal or image from spatial domain to frequency domain. DCT transforms the input into a linear combination of weighted basis functions. These basis functions are the frequency component of the input data [15]. The two-dimensional DCT is just a one-dimensional DCT applied twice, once in the x direction, and again in the y direction. When you apply the DCT to an input image, it yields a matrix of weighted values corresponding to how much of each basis function is present in the original image [16]. For most images, much of the signal energy lies at low frequencies; these appear in the upper-left corner of the DCT [17]. The lower-right values represent higher frequencies, and are often small – small enough to be neglected with little visible distortion [18]. With an input image, f(x,y), the coefficients for the output image, F(u,v) are (2D-DCT) [19]:

$$F(u,v) = \frac{2}{\sqrt{MN}} C(u)C(v) \sum_{x=0}^{M-1} f(x,y) \cos\left[\frac{(2x+1)u\pi}{2M}\right] \cos\left[\frac{(2x+1)v\pi}{2N}\right]$$

Inverse transform (2D-IDCT) are:

$$f(x,y) = \frac{2}{\sqrt{MN}} C(u)C(v) \sum_{x=0}^{M-1} F(x,y) \cos\left[\frac{(2x+1)u\pi}{2M}\right] \cos\left[\frac{(2x+1)v\pi}{2N}\right]$$

where, x = 0,1,.....,M-1
u = 0,1,.....,M-1
y = 0,1,.....,N-1
v = 0,1,.....,N-1
C(u), C(v) = $\{1/\sqrt{2}$ ,u,v = 0 ; 1, u,v ≠ 0

## 2.3.2 Disadvantages of DCT:

- Only spatial correlation of the pixels inside the single 2-D block is considered and the correlation from the pixels of the neighboring blocks is neglected [20].
- Impossible to completely decorrelate the blocks at their boundaries using DCT.
- Undesirable blocking artifacts affect the reconstructed images or video frames. (high compression ratios or very low bit rates) [21].
- Scaling as add-on→ additional effort
- DCT function is fixed→ cannot be adapted to source data.
- Does not perform efficiently for binary images (fax or pictures of fingerprints) characterized by large periods of constant amplitude (low spatial frequencies), followed by brief periods of sharp transitions [22].

## 2.3.3 Advantages of DWT over DCT:

- No need to divide the input coding into non-overlapping 2-D blocks, it has higher compression ratios avoid blocking artifacts [23].
- Allows good localization both in time and spatial frequency domain.
- Transformation of the whole image→ introduces inherent scaling.
- Better identification of which details relevant to human perception→ higher compression ratio.(64:1 vs. 500:1).
- Higher flexibility: Wavelet function can be freely chosen [24-25].

## 2.3.4 Discrete Wavelet Transform:

### 2.3.4.1 Wavelets, Wavelet Transform

A wavelet is a kind of mathematical function used to divide a given function or continuous-time signal into different components and study each component with a resolution matched to its scale. The term wavelet means a small wave. The smallness refers to the condition that this (window) function is of finite length (compactly supported). The wave refers to the condition that this function is oscillatory. The term mother implies that the functions with different region of support that are used in the transformation process are derived from one main function, or the mother wavelet. In other words, the mother wavelet is a prototype for generating the other window functions [26]. The following is the formula for the mother wavelet.

$\varphi\tau, s(\mathrm{t}) = 1/\sqrt{s}\,\varphi(\frac{t-\tau}{s})$

Here $\tau$ & s are the translation and scaling factors. Some common wavelets used are: Haar, Meyer and Morlet [27].

| | Wave | Wavelet |
|---|---|---|
| Definition | A never ending repetitive signal | A small confined signal confined within a finite region |
| Energy | Infinite because signal never ends | Finite and concentrated around a point |
| Statistical properties | Time invariant i.e. Stationary signal | Time variant i.e. non-stationary signal |
| Associated analytical properties | Fourier transform | Wavelet transform |
| Examples | Cosine wave | Haar, debauchies, Mexican hat, etc. |

Table 1: Difference between a Wave and Wavelet

In our project, we have used Haar Wavelets. The Haar wavelet is also the simplest possible wavelet. The technical disadvantage of the Haar wavelet is that it is not continuous and therefore not differentiable [28].

### 2.3.4.2 Wavelet Transform

The wavelet transform is the representation of a function by wavelets. The wavelets are scaled and translated copies of a finite-length (known as "daughter wavelets") or fast-decaying oscillating waveform (known as "mother wavelet") [29].

*Wavelet* transforms are classified into discrete wavelet transforms (DWTs) and continuous transforms (CWTs). Note that both DWT and CWT are of continuous- time (analog) transforms. They can be used to represent continuous- time (analog) signals. CWTs operate over every possible scale and translation whereas DWTs use a specific subset of scale and translation values or representation grid [30].

### 2.3.4.3 Continuous Wavelet Transform:

The continuous wavelet transform is defined as follows:

$CWT_x^{\Psi}(\tau,s) = \Psi_x^{\Psi}(\tau,s) = 1\sqrt{s}\int x(t)\varphi^*\left(\frac{t-\tau}{s}\right)dt$

As seen above, the transformed signal is a function of two variables, tau and s, the translation and scale parameters, respectively. Psi (t) is the transforming function, and it is called the mother wavelet.

The term translation is related to the location of the window, as the window is shifted through the signal. This term corresponds to time information in the transform domain.

The parameter scale in the wavelet analysis is similar to the scale used in maps. As in the case of maps, high scale correspond to a non-detailed global view (of the signal), and low scales correspond to a detailed view. Similarly, in terms of frequency, low frequencies (high scales) correspond to a global information of a signal (that usually spans the entire signal), whereas high frequencies (low scales) correspond to a detailed information of a hidden pattern in the signal (that usually lasts a relatively short time).

Fortunately in practical applications, low scales (high frequencies) do not last for the entire duration of the signal, unlike those shown in the figure, but they usually appear from time to time as short bursts or spikes. High scales (low frequencies) usually last for the entire duration of the signal.

Scaling, as a mathematical operation, either dilates or compresses a signal. Larger scales correspond to dilated (or stretched out) signals and small scale corresponds to a compressed signals. In terms of mathematical functions, if f(t) is a given function f(st) corresponds to a contracted (compressed) version of f(t) if s >1 and to an expanded (dilated) version of f(t) if s<1. However, in the definition of the wavelet transform, the scaling term is used in the denominator, and therefore, the opposite of the above statements holds, i.e., scales s>1 dilates the signals whereas scales s<1, compresses the signal [31].

### 2.3.4.4 CWT Computation

Let x(t) is the signal to be analyzed. The mother wavelet is chosen to serve as a prototype for all windows in the process. All the windows that are used are the dilated (or compressed) and shifted versions of the mother wavelet. These are a number of functions that are used for this purpose [32].

The procedure will be started from scale s=1 and will continue for the increasing values of s, i.e., the analysis will start from high frequencies and proceed towards low frequencies. This first value of s will correspond to the most compressed wavelet. As the value of s is increased, the wavelet will dilate.

The wavelet is placed at the beginning of the signal at the point which corresponds to time=0. The wavelet function at scale "1" is multiplied by the signal and then integrated over all times. The result of the integration is the multiplied by the constant number 1/sqrt{s} .This multiplication is for energy normalization purposes so that the transformed signal will have the same energy at every scale. The final result is the value of the transformation, i.e., the value of the continuous wavelet transforms at time zero and scale s=1 . In other words, it is the value that corresponds to the point tau=0, s=1 in the time-scale plane.

The wavelet at scale s=1 is then shifted towards the right by tau amount to the location t= tau, and the above equation is computed to get the transform value at t=tau , s=1 in the time-frequency plane.

This procedure is repeated until the wavelet reaches the end of the signal. One row of points on the time-scale plane for the scale s=1 is now completed. Then, s is increased by a small value. Note that, this is a continuous transform, and therefore, both tau and s must be incremented continuously. However, if this transform needs to be computed by a computer, then both parameters are increased by a sufficiently small step size. This corresponds to a sampling the time-scale plane.

The above procedure is repeated for every value of s. Every computation for a given value of s fills the corresponding single row of the time-scale plane. When the process is completed for all desired values of s, the CWT of the signal has been calculated.
If the signal has a spectral component that corresponds to the current value of s (which is 1 in this case), the product of the wavelet with the signal at the location where this spectral component exists gives a relatively large value. If the spectral component that corresponds to the current value of s is not present in the signal, the product value will be relatively small, or zero [33].

As the window width increases, the transform starts picking up the lower frequency components. As a result, for every scale and for every time (interval), one point of the

time-scale plane is computed. The computations at one scale construct the rows of the time-scale plane, and the computations at different scales construct the columns of the time-scale plane.

Note that the axes are translation and scale, not time and frequency. However, translation is strictly related to time, since it indicates where the mother wavelet is located. The translation of the mother wavelet can be thought of as the time elapsed since t=0.

Lower scales (higher frequencies) have better scale resolution (narrower in scale, which means that it is less ambiguous what the exact value of the scale) which correspond to poorer frequency resolution. Similarly, higher scales have scale frequency resolution (wider support in scale, which means it is more ambitious what the exact value of the scale is), which correspond to better frequency resolution of lower frequencies.

#### 2.3.4.5 Discrete Wavelet Transform

In numerical analysis and functional analysis, a discrete wavelet transform (DWT) is any wavelet transform for which the wavelets are discretely sampled. The Wavelet series is just a sampled version of CWT and its computation may consume significant amount of time and resources, depending on the resolution required. The DWT, which is based on sub-band coding, is found to yield a fast computation of Wavelet Transform. It is easy to implement and reduces the computation time and resources required. In DWT, a time- scale representation of the digital signal is obtained using digital filtering techniques. The signal to be analyzed is passed through filters with different cut-off frequencies at different scales [34].

Although the discretized continuous wavelet transform enables the computation of the continuous wavelet transform by computers, it is not a true discrete transform. As a matter of fact, the wavelet series is simply a sampled version of the CWT, and the information it provides is highly redundant as far as the reconstruction of the signal is concerned. This redundancy, on the other hand, requires a significant amount of computation time and resources. The discrete wavelet transform (DWT), on the other hand, provides sufficient information both for analysis and synthesis of the original signal, with a significant reduction in the computation time.

The main idea is the same as it is in CWT. A time-scale representation of a digital signal is obtained using digital filtering techniques. Recall that CWT is a correlation between a wavelet at different scales and the signal with the scale (or the frequency) being used as a measure of similarity. The continuous wavelet transform was computed by changing the scale of the analysis window, shifting the window in time, multiplying by the signal, and integrating over all times. In the discrete case, filters of

different cutoff frequencies are used to analyze the signal at different scales. The signal is passed through a series of high pass filters to analyze the high frequencies, and it is passed through a series of low pass filters to analyze the low frequencies.

The resolution of the signal, which is a measure of the amount of detail information in the signal, is changed by the filtering operations, and the scale is changed by upsampling and downsampling (subsampling) operations. Subsampling a signal corresponds to reducing the sampling rate, or removing some of the samples of the signal. For example, subsampling by two refers to dropping every other sample of the signal. Subsampling by a factor n reduces the number of samples in the signal n times.

Upsampling a signal corresponds to increasing the sampling rate of a signal by adding new samples to the signal. For example, upsampling by two refers to adding a new sample, usually a zero or an interpolated value, between every two samples of the signal. Upsampling a signal by a factor of n increases the number of samples in the signal by a factor of n [35].

The procedure starts with passing this signal (sequence) through a half band digital lowpass filter with impulse response h[n]. Filtering a signal corresponds to the mathematical operation of convolution of the signal with the impulse response off the filter. The convolution operation in discrete time is defined as follows:

$x[n] * h[n] = \sum_{K=-\infty} x[k].h[n-k]$

A half band lowpass filter removes all frequencies that are above half of the highest frequency in the signal. For example, if a signal has a maximum of 1000Hz component, the half band lowpass filtering removes all the frequencies above 500Hz.

The unit of frequency is of particular importance at this time. In discrete signals, frequency is expressed in terms of radians. Accordingly, the sampling frequency of the signal is equal to $2\pi$ radians in terms of radial frequency. Therefore, the highest frequency component that exists in a signal will be $\pi$ radians, if the signal is sampled at Nyquist's rate (which is twice the maximum frequency that exists in the signal); that is, the Nyquist's rate corresponds to $\pi$ rad/s in the discrete frequency domain. Therefore using Hz is not appropriate for discrete signals. However, Hz is used whenever it is needed to clarify a discussion, since it is very common to think of frequency in terms of Hz. It should always be remembered that the unit of frequency for discrete time signals is radians [36].

After passing the signal through a half band lowpass filter, half of the samples can be eliminated according to the Nyquist's rule, since the signal now has a highest frequency of $\pi/2$ radians instead of $\pi$ radians. Simply discarding every other sample will subsample the signal by two, and the signal will then have half the number of points. The scale of the signal is now doubled. Note that the lowpass filtering removes

the high frequency information, but leaves the scale unchanged. Only the subsampling process changes the scale. Resolution, on the other hand, is related to the amount of information in the signal, and therefore, it is affected by the filtering operations. Half band lowpass filtering removes half of the frequencies, which can be interpreted as losing half of the information. Therefore, the resolution is halved after the filtering operation. Note, however, the subsampling operation after filtering does not affect the resolution, since removing half of the spectral components from the signal makes half the number of samples redundant anyway. Half the samples can be discarded without any loss of information. In summary, the lowpass filtering halves the resolution, but leaves the scale unchanged. The signal is then subsampled by 2 since half of the number of samples is redundant. This doubles the scale.

This procedure can mathematically be expressed as:

$y[n] = \sum_{k=-\infty}^{\infty} h[k].x[2n-k]$

Having said that, we now look how the DWT is actually computed: The DWT analyzes the signal at different frequency bands with different resolutions by decomposing the signal into a coarse approximation and detail information. DWT employs two sets of functions, called scaling functions and wavelet functions, which are associated with low pass and highpass filters, respectively. The decomposition of the signal into different frequency bands is simply obtained by successive highpass and lowpass filtering of the time domain signal. The original signal x[n] is first passed through a halfband highpass filter g[n] and a lowpass filter h[n]. After the filtering, half of the samples can be eliminated according to the Nyquist's rule, since the signal now has a highest frequency of $\pi/2$ radians instead of $\pi$. The signal can therefore be subsampled by 2, simply by discarding every other smaple. This constitutes one level of decomposition and can be mathematically be expressed as follows:

$y_{high}[k] = \sum_n x[n].g[2k-n]$

$y_{low}[k] = \sum_n x[n].h[2k-n]$

Where yhigh[k] and ylow[k] are the outputs of the highpass and lowpass filters, respectively, after subsampling by 2.

This decomposition halves the time resolution since only half the number of samples now characterizes the entire signal. However, this operation doubles the frequency resolution, since the frequency band of the signal now spans only half the previous frequency band, effectively reducing the uncertainity in the frequency by half. The above procedure, which is also known as the subband coding, can be repeated for further decomposition. At every level, the filtering and subsampling will result in half the number of samples (and hence half the time resolution) and half the frequency band spanned (and hence doubles the frequency resolution). The following figure

illustrates this procedure, where x[n] is the original signal to be decomposed, and h[n] and g[n] are lowpass and highpass filters, respectively. The bandwidth of the signal at every level is marked on the figure as "f" [37].

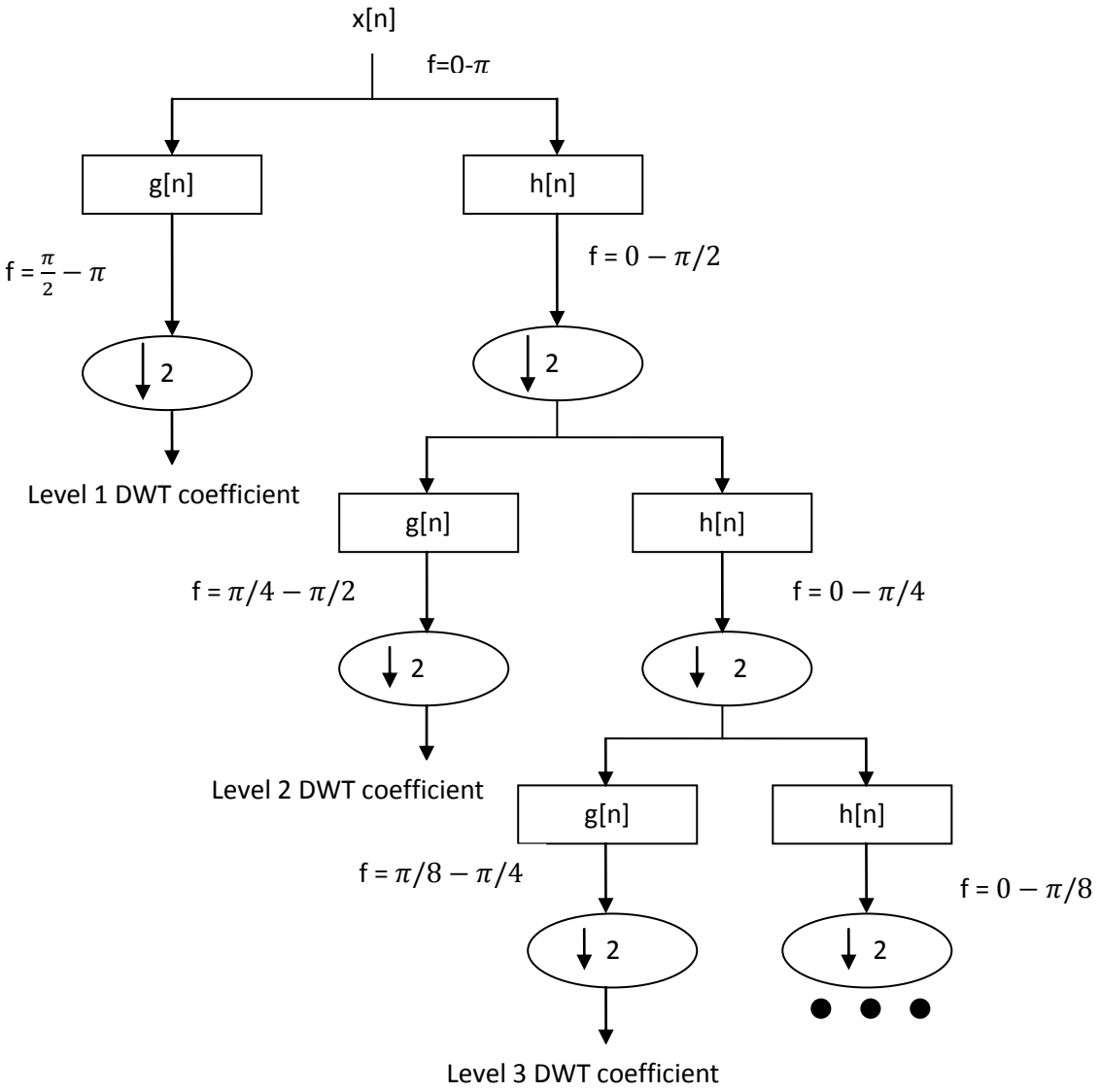



Figure 1.3: Subband Coding

The frequencies that are most prominent in the original signal will appear as high amplitudes in that region of the DWT signal that includes those particular frequencies. The difference of this transform from the Fourier transform is that the time localization of these frequencies will not be lost. However, the time localization will have a resolution that depends on which level they appear. If the main information of the signal lies in the high frequencies, as happens most often, the time localization of these frequencies will be more precise, since they are characterized by more number of samples. If the main information lies only at very low frequencies, the time localization will not be very precise, since few samples are used to express signal at these frequencies. This procedure in effect offers a good time resolution at high frequencies, and good frequency resolution at low frequencies. Most practical signals

encountered are of this type. The frequency bands that are not very prominent in the original signal will have very low amplitudes, and that part of the DWT signal can be discarded without any major loss of information, allowing data reduction.

One area that has been benefited the most from this particular property of the wavelet transforms is image processing. As you may well know, images, particularly, high-resolution images claim a lot of disk space [38].

For a given image, you can compute the DWT of, say each row, and discard all values in the DWT that are less than a certain threshold. We then save only those DWT coefficients that are above the threshold for each row, and when we need to reconstruct the original image, we simply pad each row with as many zeroes as the number of discarded coefficients, and use the inverse DWT to reconstruct each row of the original image. We can also analyze the image at different frequency bands, and reconstruct the original image by using only the coefficients that are of a particular band [36-39].

#### 2.3.4.6 Advantages of DWT

1. Using wavelets, the whole image is seen as one block – the edges are no longer given.
2. Wavelets property of multiresolution analysis reduces the computational time of the detection procedure.
3. Allows good localization both in time and spatial frequency domain.
4. Better identification of which data is relevant to human perception.
5. Higher flexibility: Wavelet function can be freely chosen.
6. Critical sampling, which is the ability of the basis element to have little redundancy.

#### 2.3.4.7 Disadvantages of DWT

1. The cost of computing DWT as compared to DCT may be higher.
2. The use of larger DWT basis function or wavelet filters producing blurring and ringing noise near edge regions in images or video frames.
3. Longer compression time.
4. Lower quality than JPEG at low compression rates.
5. Lacks in providing directionality, which is having basis elements defined in various directions.
6. Lacks in providing Anisotropy, which is having basis elements defined in various aspect ratios and shapes.

### 2.3.4.8 Singular Value Decomposition:

SVD is a matrix factorization technique commonly used for producing low-rank approximations. Given an m× n matrix A, with rank r, the singular value decomposition, SVD(A), is defined as [40]:

$SVD(A) = U \times S \times V^T$

where U, S and V are of dimensions m × m, m × n, and n × n, respectively. Matrix S is a diagonal matrix having only r nonzero entries, which makes the effective dimensions of these three matrices m × r, r × r, and r × n, respectively. U and V are two orthogonal matrices and S is a diagonal matrix, called the singular matrix. The diagonal entries (s1, s2, .. , sr) of S have the property that si > 0 and $s1 \geq s2 \geq \ldots \geq sr$. The first r columns of U and V represent the orthogonal eigenvectors associated with the r nonzero eigen values of $AA^T$ and $A^TA$, respectively [41].
In other words, the r columns of U corresponding to the nonzero singular values span the column space, and the r columns of V span the row space of the matrix A. U and V are called the left and the right singular vectors, respectively. SVD has an important property that makes it particularly interesting for our application. SVD provides the best low-rank linear approximation of the original matrix A. It is possible to retain only k << r singular values by discarding other entries. We term this reduced matrix $S_k$. Since the entries in S are sorted i.e., $s1 \geq s2 \geq \ldots \geq sr$, the reduction process is performed by retianing the first k singular values. The matrices U and V are also reduced to produce matrices $U_k$ and $V_k$, respectively. The matrix $U_k$ is produced by removing (r − k) columns from the matrix U and matrix $V_k$ is produced by removing (r − k) rows from the matrix V . When we multiply these three reduced matrices, we obtain a matrix $A_k$. The reconstructed matrix $A_k = U_k.S_k.V\ T_k$ is a rank-k matrix that is the closest approximation to the original matrix A. More specifically, $A_k$ minimizes the Frobenius norm $||A - A_k||_F$ over all rank-k matrices. Researchers pointed out that the low rank approximation of the original space is better than the original space itself due to the filtering out of the small singular values that introduce “noise” in the customer-product relatioship. The dimensionality reduction approach in SVD can be very useful for the collaborative filtering process. SVD produces a set of uncorrelated eigenvectors. Each customer and product is represented by its corresponding eigenvector. The process of dimensionality reduction may help customers who rated similar products (but not exactly the same products) to be mapped into the space spanned by the same eigenvectors. We now present an outline of the prediction generation algorithm using SVD [42-50].

As an optimal matrix decomposition technique, SVD packs maximum signal energy into as few coefficients as possible and has the ability to adapt to local variations of a given image. Also singular value has properties of stability, proportion invariance, and rotation invariance. SVD can effectively reveal essential property of image matrices, so it has been used in a variety of image processing applications such as:

a. The singular values (SVs) of an image have very good stability, that is, when a small perturbation is added to an image, its SVs do not change significantly; and
b. SVs represent intrinsic algebraic image properties [51].

### 2.3.4.9 DCT-SVD Based Watermarking:

Robustness, capacity and imperceptibility are the three important requirements of an efficient watermarking scheme. SVD based watermarking scheme has high imperceptibility. Although the SVD based scheme withstand certain attacks, it is not resistant to attacks like rotation, sharpening, etc. Also SVD based technique has only limited capacity. These limitations have led to the development of a new scheme that clubs the properties of DCT and SVD. This particular algorithm proves to be better than ordinary DCT based watermarking and ordinary SVD based watermarking scheme [52].

### 2.3.4.10 DWT-SVD Based Watermarking:

The above mentioned SVD-DCT scheme has enormous capacity because data embedding is possible in all the sub-bands. Watermark was found to be resistant to all sorts of attacks except rotation and achieved good imperceptibility. Disadvantage is that the embedding and recovery are time consuming process because the zig-zag scanning to map the coefficients into four quadrants based on the frequency. Alternatively if we apply DWT we get the four frequency sub-bands directly namely; approximation, horizontal, vertical and diagonal bands. So the time consumption will be greatly reduced. Also, SVD is a very convenient tool for watermarking in the DWT domain [53].

### 2.3.4.11 DWT-DCT-SVD Based Watermarking:

This method utilizes the wavelets coefficients of the cover image to embed the watermark. Any of the three high frequency sub bands of wavelet coefficients can be used to watermark the image. The DCT coefficients of the wavelet coefficients are calculated and singular values decomposed. The singular values of the cover image and watermark are added to form the modified singular values of the watermarked image. Then the inverse DCT transformed is applied followed by the inverse DWT. This is the algorithm that clubs the properties of SVD, DCT and DWT. Watermark embedded using this algorithm is highly imperceptible. This scheme is robust against all sorts of attacks. It has very high data hiding capacity. The new method was found to satisfy all requisites of an ideal watermarking scheme such as imperceptibility or fidelity, robustness and good capacity. Also, the method is robust against different kinds of mentioned attacks. This method can be used for authentication and data hiding purposes [54].

## 2.3.4.12 Contourlet Transform:

Recently, Do and Vetterli proposed the contourlet transform as a directional multiresolution image representation that can efficiently capture and represent smooth object boundaries in natural images. The contourlet transform is constructed as a combination of the Laplacian pyramid and the directional filter banks (DFB). Conceptually, the flow of operation can be illustrated by Figure 1(a), where the Laplacian pyramid iteratively decomposes a 2-D image into lowpass and highpass subbands, and the DFB are applied to the highpass sub- bands to further decompose the frequency spectrum. Using ideal filters, the contourlet transform will decompose the 2-D frequency spectrum into trapezoid-shaped regions as shown in Figure 1(b) [62].

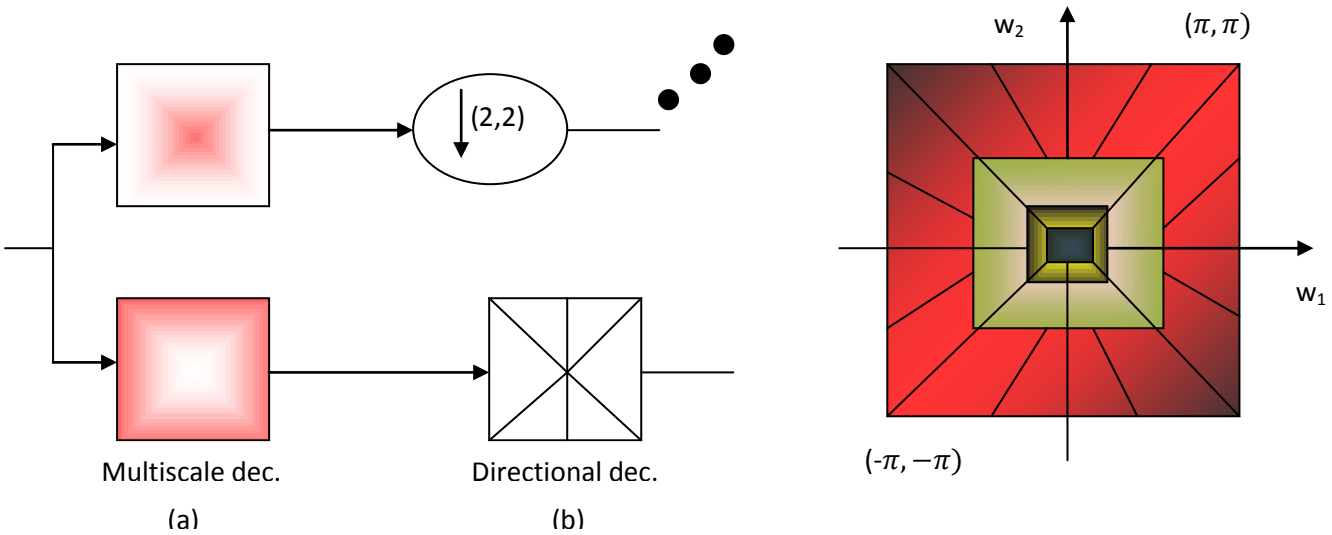



Figure 1.4: The original Contourlet Transform. (a) Block diagram.
(b) Resulting frequency division.

The Contourlet Transform is geometrical images based transform. In a Contourlet Transform the Laplacian Pyramid is the first step that has to be performed and includes point discontinuities. The second step is consumed by Directional Filter Banks which combats point discontinuities to linear structures [63]. The Laplacian Pyramid decomposition generates a low pass down sampled version of the host signal at each level, and the band pass version includes the difference between the original and the predicted signal. The bandpass image yielded in the LP decomposition is further processed by the DFB [64]. Designing of the DFB is efficient in providing high-frequency content, including smooth contours and directional edges. Implementation of DFB can be categorically termed as efficient due to the K-level binary tree decomposition that leads to $2^K$ subbands that has wedge-shaped frequency partitioning, which is included in it [65]. The final result accommodates the image expansion using contour segment as basic elements and thereby called Contourlet Transform, having its implementation by the Pyramidal Directional Filter Bank (PDFB). Including the integrated attribute of LP and DFB the CT also has some additional features [66]:

- PDFB offers the tense frame of which boundary frame is 1 only when the LP and DFB are computed by orthogonal filters.

- PDFB allows reconstructing the host image only when the LP and DFB are computed by complete restructuring filters.

- PDFB gives the computing complexity of N-pixel image as O(N) when finite impulse response FIR filter is used.

- The supporting set for PDFB-base image is $2^{j+lj-2}$ in length and, $2^{j}$ in width when the high pass subband obtained by decomposing the j level of LP is input for DFB, the $l_j$ binary tree.

- The upper limit of PDFB is 4/3 and its redundancy comes from LP.

The Contourlet Transform is vulnerable in providing a flexible image multi-resolution presentation. In comparison to Wavelet, Contourlet represent richer directions and shapes whereas Wavelets could only infer about the horizontal, vertical and diagonal directions. Later after Contourlet Transform the low frequency subimages suffer little impact caused by regular image processing as they gather most of the energy [67-68].

#### 2.3.4.13 Comparing Contourlet Transform with Wavelet Transform:

- Contourlet offers more directions and shapes so it is more efficient in capturing contours and geometric structures in images.

- Contourlet Transform provides higher non-linear approximations and is thus better in terms of image compression than wavelet transform.

- For image denoising, the random noise will generate significant wavelet coefficients, which is like the true edges, but it is less likely to generate significant contourlet coefficients. Therefore, the thresholding performed on the contourlet is more efficient in removing the noise than the wavelet and also provides preservation of high frequency texture.

# CHAPTER 3.

# IMPLEMENTATION

## 3.1 About MATLAB:

Short for "matrix laboratory", MATLAB was invented in the late 1970s by Cleve Moler, then chairman of the computer science department at the University of New Mexico. He designed it to give his student's access to LINPACK and EISPACK without having to learn FORTRAN. It soon spread to other universities and found a strong audience within the applied mathematics community. Jack little, an engineer, was exposed to it during a visit Moler made to Stanford University in 1983. Recognizing its commercial potential, he joined with Moler and Steve Bangert. They rewrote MatLab in C and founded The Math Works in 1984 to continue its development. These rewritten libraries were known as JACKPAC. MATLAB was first adopted by control design engineers, Little's speciality, but quickly spread to many domains. It is now also used in education, in particular the teaching of linear algebra and numerical analysis, and is popular amongst scientists involved with image processing MATLAB is built around the MATLAB language, sometimes called M-code or simply M. The simplest way to execute M-code is to type it in at the prompt, >>, in the Command Window, one of the elements of the MATLAB Desktop. In this way, MATLAB can be used as an interactive mathematical shell. Sequences of commands can be saved in a text file, typically using the MATLAB Editor, as a script or encapsulated into a function, extending the commands available.

## 3.2 Introduction to Digital Image:

A digital image is composed of pixels which can be thought of as small dots on the screen. A digital image is an instruction of how to color each pixel. We will see in detail later on how this is done in practice. A typical size of an image is 512-by-512 pixels. Later on in the course you will see that it is convenient to let the dimensions of the image to be a power of 2. For example, 2 POW 9 = 512. In general case we say that an image is of size m-by-n if it is composed of m pixels in the vertical direction and n pixels in the horizontal direction.

Let us say that we have an image on the format 512-by-1024 pixels. This means that the data for the image must contain information about 524288 pixels, which requires a lot of money. Hence compressing images is essential for efficient image processing. You will later on see how Fourier analysis and Wavelet analysis can help us to compress an image significantly. There are also a few "computer scientific" tricks (e.g. entropy coding) to reduce the amount of data required to store an image.

The following Formats are supported by MATLAB:

- BMP
- HDF
- JPEG
- PCX
- TIFF
- XWB

Most images you find on the internet are JPEG-images which is the name of one of the most widely used compression standards for images. If you have stored an image you can usually see from the suffix what format it is stored in.

## 3.3 Working formats in MATLAB:

If an image is stored as a JPEG-image we first read it into MATLAB. However, in order to start working with an image, for example perform a wavelet transform on the image, we must convert it into a different format. This section explains four common formats.

### 3.3.1 Intensity image (gray scale image):

This is the equivalent to a "gray scale image" and this is the image we will mostly work with in this course. It represents an image as a matrix where every element has a value corresponding to how bright/dark the pixel at the corresponding position should be colored. There are two ways to represent the number that represents the brightness of the pixel: The double class (or data type). This assigns a floating number ("a number with decimals") between 0 and 1 to each pixel. The value 0 corresponds to black and the value 1 corresponds to white. The other class is called uint8 which assigns an integer between 0 and 255 to represent the brightness of a pixel. The value 0 corresponds to black and 255 to white. The class uint8 anly requires roughly 1/8 of the storage compared to the class double. On the other hand, many mathematical functions can only be applied to the double class. We will see later how to convert between double and uint8.

### 3.3.2 Binary image:

This image format also stores an image as a matrix but can only color a pixel black or white (and nothing in between). It assigns a 0 for black and a 1 for white.

### 3.3.3 Indexed image:

This is a practical way of representing color images. (In this course we will mostly work with gray scale images but once you have learned how to work with a gray scale image you will also know the principle how to work with color images.) An indexed image stores an image as two matrices. The first matrix has the same size as the image and one number for each pixel. The second matrix is called map and its size may be different from the image. The numbers in the first matrix is an instruction of what number to use in the color map matrix.

### 3.3.4 RGB image:

This is another format for color images. It represent an image with three matrices of sizes matching the image format. Each matrix corresponds to one of the colors red, green or blue and gives an instruction of how much of each of these colors a crtain pixel should use.

### 3.3.5 Multi frame image:

In some applications we want to study a sequence of images. This is very common in biological and medical imaging where you might study a sequence of slices of a cell. For

these cases, the multiframe format is a convenient way of working with a sequence of images. In case you choose to work with biological imaging later on in this course, you may use this format.

## 3.4 Fundamental Operations:

How to convert different formats. The following table shows how to convert between the different formats given above. All these commands require the Image processing tool box. The command mat2gray is useful if you have a matrix representing an image but the values representing the gray scale range between, let's say, 0 and 1000.

| S.No. | Operations | MATLAB Commands |
|---|---|---|
| 1. | Convert between intensity/Indexed/RGB format to binary format | dither( ) |
| 2. | Convert between intensity format to Indexed format | gray2ind ( ) |
| 3. | Convert between indexed format to intensity format | ind2gray ( ) |
| 4. | Convert between indexed format to RGB format | ind2rgb( ) |
| 5. | Convert regular matrix to intensity format by scaling | mat2gray ( ) |
| 6. | Convert between RGB format to indexed format | rgb2ind ( ) |
| 7. | Convert between RGB format to intensity format | rgb2gray ( ) |

Table 3.4. Functions for image format conversion

The command mat2gray automatically rescales all entries so that they fall within 0 and 255 (if you use uint8 class) or 0 and 1 (if you use the double class).

### 3.4.1 How to convert between double and uint8:

When you store an image, you should store it as a uint8 image since this requires far less memory than double. When you are processing an image (that is performing mathematical operations on an image) you should convert it into a double. Converting back and forth between these classes is easy.

```
I = im2double (I);    %converts an image named I from uint8 to double%
I = im2uint8 (I);     %converts an image named I from double to uint8%
```

### 3.4.2 Loading and saving variables in MATLAB:

This section explains how to load and save variables in MATLAB. Once ffile is read, we probably convert intoan intensity image (a matrix) and work with this matrix.the matrix representing can be saved in order to continue to work with this matrix at another time. This is done easily using the commands save and load. Note that save and load are commonly used MATLAB commands, and works independently of what tool boxes that are installed.

| S.No. | Opertaion | MATLAB Commands |
|---|---|---|
| 1. | Save the Variable X | Save X |
| 2. | Load the Variable X | Load X |

Table 3.4.2. Loading and Saving Variables

## 3.5 Some Limitations:

MATLAB is proprietary product of The Math Works, so users are subject to vendor lock in. Some other source languages, however, are partially compatible and provide a migration path. The language shows a mixed heritage with a sometimes erratic syntax. For example, MATLAB uses parentheses, e.g. y = f(x), for both indexing into an array and calling a function. Although this ambiguous syntax can facilitate a switch between a procedure and a lookup table, both of which correspond to mathematical functions, a careful reading of the code may be required to establish the internet. Many functions have a different behaviour with matrix and vector arguments. Since vectors are matrices of one row or one column, this can give unexpected results. For instance, function sum(A) where A is a matrix gives a row vector containing the sum of each column of A, and sum(V) where V is a column or row vector gives the sum of its elements; hence the programmer must be careful if the matrix argument of sum can degenerate into a single row array. While sum and many similar functions accept an optical argument to specify a direction, others, like plot, do not, and require additional checks. There are other cases where MATLAB's interpretation of code may not be consistently what the user intended (e.g. how spaces are handled inside brackets as separators where it makes sense but not where it doesn't, or backlash escape sequences which are interpreted by some functions like fprint but not directly by the language parser because it wouldn't be convenient for Windows directories). What might be considered as a convenience for commands typed interactively where the user can check that MATLAB does what the user wants may be less supportive of the need to construct reusable code. Though other data types are available, the default is a matrix of doubles. This array type does not include a way to attach attributes such as engineering units or sampling rates. Although time and date markers were added in R14SP3 with the time series object, sample rate is still lacking. Such attributes can be managed by the user via structures or other methods. Array indexing is one-based which is the common convention for matrices in mathematics, but does not accommodate the indexing convention of sequences that have zero or negative indices. For instance, in MATLAB the DFT (or FFT) is defined with the DC component at index 1 instead of index 0, which is not consistent with the standard definition of the DFT. This one-based indexing convention is hard coded into MATLAB, making it impossible for a user to define their own zero-based or negative indexed arrays to concisely model an idea having non-positive indices. MATLAB doesn't support references, which makes it difficult to implement data structures that contain indirections, such as open hash tables, linked list, trees, and various other common computer science data structures. In addition since the language is consistently call-by-value, it means that functions that modify array or object value must return these through output parameters and the caller must assign those modified values for the change to be persistent.

# CHAPTER 4.

# Steps for Watermark Extraction and Embedding

## 4.1 Steps for Watermark Embedding using DWT and SVD Technique:

1. Extract the red component of the image with (:, :, 1).

2. One level Haar Discrete Wavelet Transform to decompose cover image into four subbands.
   [ca1, ch1, cv1, cd1]=dwt2 (image,'haar') (1)

3. Apply Singular Value Decomposition to the vertical (cv1) and horizontal (ch1) coefficients.
   [U1, S1, V1]=svd(ch1) (2)
   [U2, S2, V2]=svd(cv1) (3)

4. Divide the watermark into two parts.
   W=W1+W2 (4)

5. Extract the red component of the watermark as well like for the image, with (:, :, 1).

6. Modify the singular values of vertical and horizontal plane in 2. Along with the inputted scale factor (α).
   S1 + αW1 = Uw *Sw *VwT (5)
   S2 + αW2 = Uw *Sw *VwT (6)

7. Two sets of modified DWT coefficients are made available by 4.
   Mod_c_h = U1 *Sw *V1T (7)
   Mod_c_v = U2 *Sw *V2T (8)

8. Apply the inverse Discrete Wavelet Transform, i.e. i-dwt on the two sets of modified coefficients in 5 (cv1 and ch1) and non-modified coefficients in 1 (ca1 and cd1).
   WI=idwt2 (ca1,Mod_c_h,Mod_c_v,cd1,'haar') (9)

9. Replace the first component of the image that is processed with the original image's first component.

## 4.2 Steps for Watermark Extraction using DWT and SVD Technique:

10. For the Extraction of the watermark: (in the red component). Apply one level Haar DWT to the watermarked image obtained in 9.

    [ca2, ch2, cv2, cd2] = dwt2 (WI,"haar') (10)

11. Apply SVD to the vertical and horizontal coefficients, where U and V are of original image and S is of the watermarked image from 2 and 4 respectively.

    [U1, Sw, V1] = svd (ch2) (11)
    [U2, Sw, V2] = svd (cv2) (12)

12. Compute the replaced coefficients by placing the U and V of the original watermark along with the singular value S used in 8.

    M_c_h = Uw *Sw *VwT (13)
    M_c_v = Uw *Sw *VwT (14)

13. Extract half of the watermark by

    W1* = (M_c_h – S1) / α (15)
    W2* = (M_c_h – S2) / α (16)

14. Combine the results of 4 to obtain the original watermark.

    W* = W*1 +W*2 (17)

## 4.3 Steps for Watermark Embedding using Contourlet Transform:

1. Resize the watermark.
2. Apply CT of level 2 on the watermark, directional sub bands would be made available by application of CT.
3. On the fourth directional subband apply LP in order to obtain bandpass image W.
4. Apply SVD on the bandpass image obtained after applying LP i.e. on W.

   $W=UW\times SW\times VWT$ (1)

5. Resize the cover image.
6. Dealt this cover image from RGB to YCbCr colour space.
7. For every individual colour component perform:
8. DCT for all the components.
9. CT of level 2 in order to get the directional subbands.
10. Apply LP decomposition on every fourth directional subband obtained from above performed CT.
11. Apply SVD on the bandpass component obtained after LP decomposition.

    $CPi = UPi \times SPi \times VPi$ (2)

12. Modify the singular values of the bandpass image, with SBPi and Sw obtained in the above steps and α as the scale factor.

    $SPMi = SPi + \alpha \times SW$ (3)

13. With the above modified singular value SPMi obtain the modified bandpass image with UPi and VPi as the orthogonal matrices of the bandpass of the cover image.

    $CPMi = UPi\times SPMi\times VPiT$ (4)

14. The modified bandpass image thus obtained above is applied with reconstruction LP.
15. All the above modified components are used to be applied in the ICT and inverse of CT is obtained.
16. Inverse DCT (IDCT) is also performed.
17. Transform the image back into RGB colour space to obtain the watermarked image.

## 4.4 Steps for Watermark Extraction using Contourlet Transform:

18. Covert the cover image and the watermarked image from RGB to YCbCr colour space.
19. Individually for each colour component for the cover image and the watermarked image perform:
20. DCT on all the components of the cover image and the watermarked image.
21. CT of level 2 in order to obtain directional subbands.
22. Apply LP decomposition on every fourth level directional subband obtained after CT.
23. Apply SVD on the bandpass image obtained after LP of the cover image and the watermarked image respectively

    $$CPi = UPi \times SPi \times VPiT \quad (5)$$

    $$CWWi = UWWi \times SWWi \times VWWiT \quad (6)$$

24. Extract the singular values of the watermark by applying the formula where SWWi is a singular matrix for watermarked image and SPi is the matrix for cover image and $\alpha$ is the scale factor

    $$Sext = SWWi - SPi / \alpha \quad (7)$$

25. Apply SVD on all the bandpass images of the watermark obtained after LP in order to obtain the extracted watermark coefficients

    $$Wext = UW \times Sext \times VWT \quad (8)$$

26. Apply reconstruction LP on the extracted watermark components obtained above.
27. Apply ICT on the resultant of the above step in order to obtain the watermark.

# CHAPTER 5:

# RESULTS:

## 5.1 Results of Watermarking using SVD and DWT Technique:

Image 1: Original Lena Image (Cover Image) Of size 512×512

Image 2: Original Watermark Image (Camera man) Of size 256×256

Image 3: Watermarked Image using SVD and DWT Technique
Of size 512×512

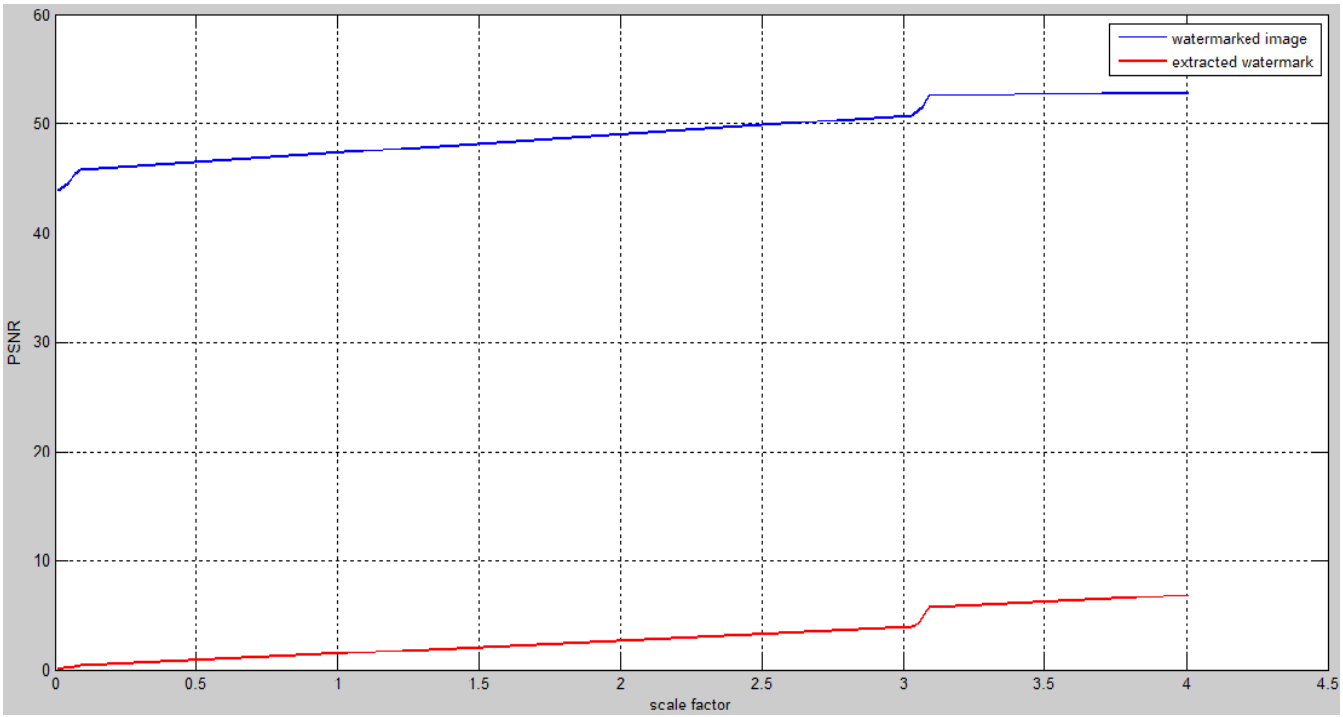
Image 4: Extracted Watermark from the Watermarked Image using SVD and DWT Technique Of size 512×512

## 5.2 Tabular results for the varying PSNR using SVD and DWT Technique:

| Scale Factor | 0.01 | 0.03 | 0.05 | 0.07 | 0.09 |
|---|---|---|---|---|---|
| Watermarked Image | 52.92 | 50.65 | 48.56 | 43.20 | 39.80 |
| Extracted Image | 6.80 | 5.70 | 4.90 | 4.20 | 3.90 |

Table: Variation in PSNR of Watermarked Image and Extracted Watermark with different Scale Factor.

## 5.3 Graph representing the varying PSNR of the Watermarked and the Extracted Watermark Image using SVD and DWT Technique:




Graph: Plot of Varying PSNR with Scale Factor of Watermarked Image and Extracted Watermark.

## 5.4 Result after checking the robustness of the SVD and DWT Technique with 'salt and pepper' noise and 'speckle' noise:

Image 5 : Effect of 'salt and pepper' noise on the Watermarked Image.

Image 6 : Effect of 'salt and pepper' noise on the Extracted Watermark.

## 5.5 Tabular results for the effect of 'salt and pepper' noise on the Watermarked Image and the Extracted Image in order to preserve the Robustness:

| Scale Factor | 0.01 | 0.03 | 0.05 | 0.07 | 0.09 |
|---|---|---|---|---|---|
| Watermarked Image | 21.80 | 21.65 | 20.40 | 19.88 | 19.20 |
| Extracted Image | 26.49 | 26.40 | 25.96 | 25.32 | 24.99 |

Table: Varying PSNR for 'salt and pepper' noise for different scale factor values.

Image 7 : Effect of ‘speckle’ noise on the Watermarked Image.

Image 8 : Effect of ‘speckle’ noise on the Extracted Watermark Image.

## 5.6 Tabular results for the effect of 'speckle' noise on the Watermarked Image and the Extracted Image in order to preserve the Robustness:

| Scale Factor | 0.01 | 0.03 | 0.05 | 0.07 | 0.09 |
|---|---|---|---|---|---|
| Watermarked Image | 20.98 | 20.32 | 19.93 | 18.67 | 17.60 |
| Extracted Image | 32.60 | 31.89 | 30.88 | 29.78 | 28.55 |

Table: Varying PSNR for 'speckle' noise for different scale factor values.

## 5.7 Results for Edge Detection using SVD and DWT Technique:

Image 9 : Edge Detection using ‘sobel’ transform for the Watermarked Image.

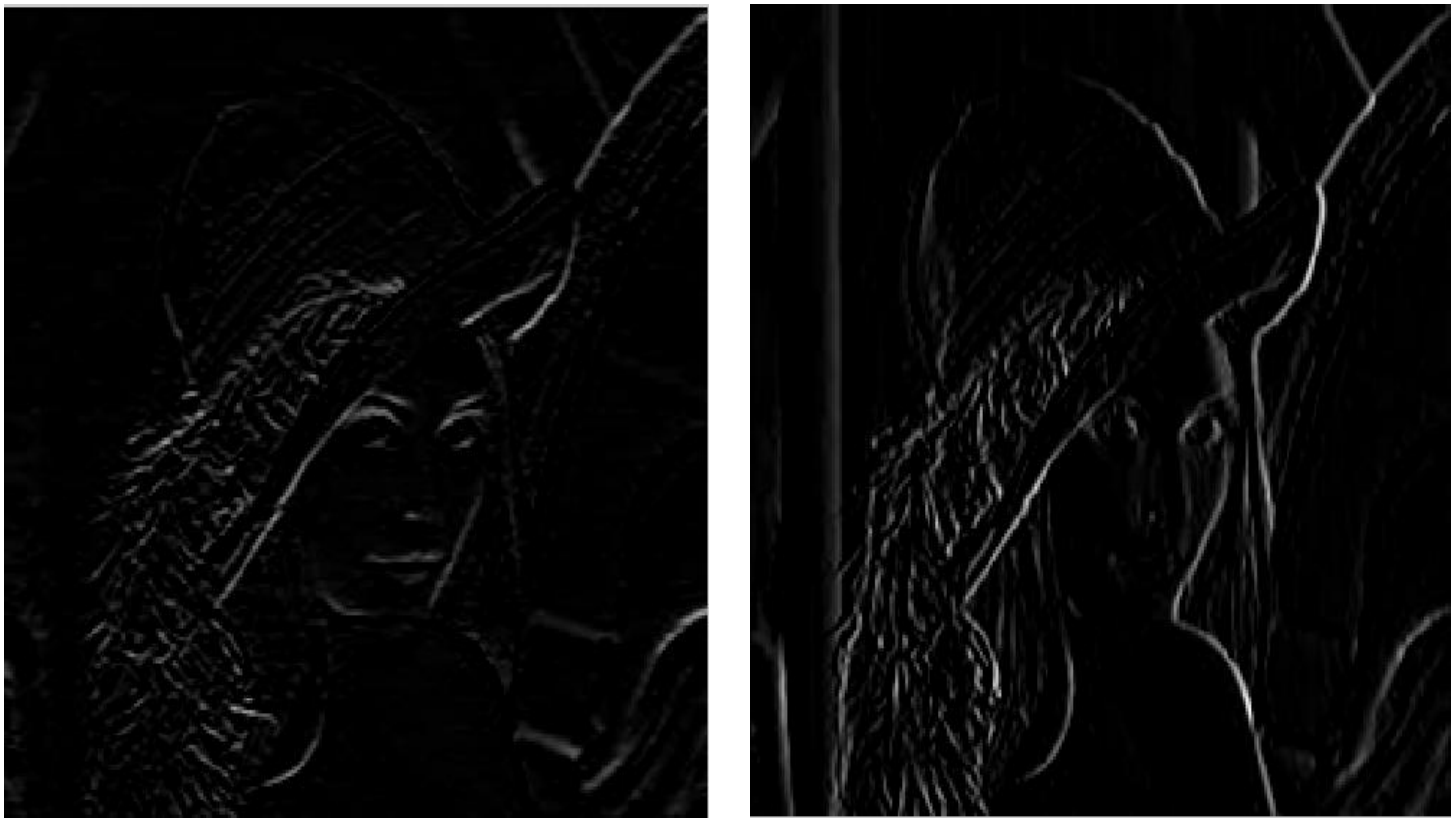


Image 10 : Horizontal Edge of the Watermarked Image and Vertical Edge of the Watermarked Image

Image 11 : Edge Detection of the Extracted Watermark using ‘sobel’ transform.

Image 12 : Horizontal edges of the Extracted Image and Vertical edges of the Extracted Image.

## 5.8 Tabular results for the effect of 'edge detection' on the Watermarked Image and the Extracted Image using SVD and DWT Technique:

| Scale Factor | 0.01 | 0.03 | 0.05 | 0.07 | 0.09 |
|---|---|---|---|---|---|
| Watermarked Image | 3.27 | 2.95 | 2.21 | 1.70 | 1.23 |
| Extracted Image | 14.30 | 13.78 | 13.53 | 12.84 | 12.33 |

Table: Varying PSNR for 'edge detection' for different scale factor values.

## 5.9 Results for Watermarking using Contourlet Transform:

Image 13: Original Lena Image (Cover Image)
Of size 512×512

Image 14: Original Watermark (Camera man) image

Image 15: Watermarked Image using Contourlet Transform Of size 512×512

Image 16: Extracted Watermark Image using Contourlet Transform Of size 512×512

## 5.10 Tabular results for the effect of Contourlet Transform on Watermarking Technique:

| Scale Factor | 0.01 | 0.03 | 0.05 | 0.07 | 0.09 |
|---|---|---|---|---|---|
| Watermarked Image (dB) | 48.24 | 48.02 | 47.92 | 47.80 | 47.32 |
| Extracted Image (dB) | 21.93 | 31.34 | 35.51 | 38.07 | 39.82 |

Table: Varying PSNR for Contourlet Transform for different scale factor values.

## 5.11 Result after checking the robustness of the Contourlet Transform Technique with ‘salt and pepper’ noise and ‘speckle’ noise:

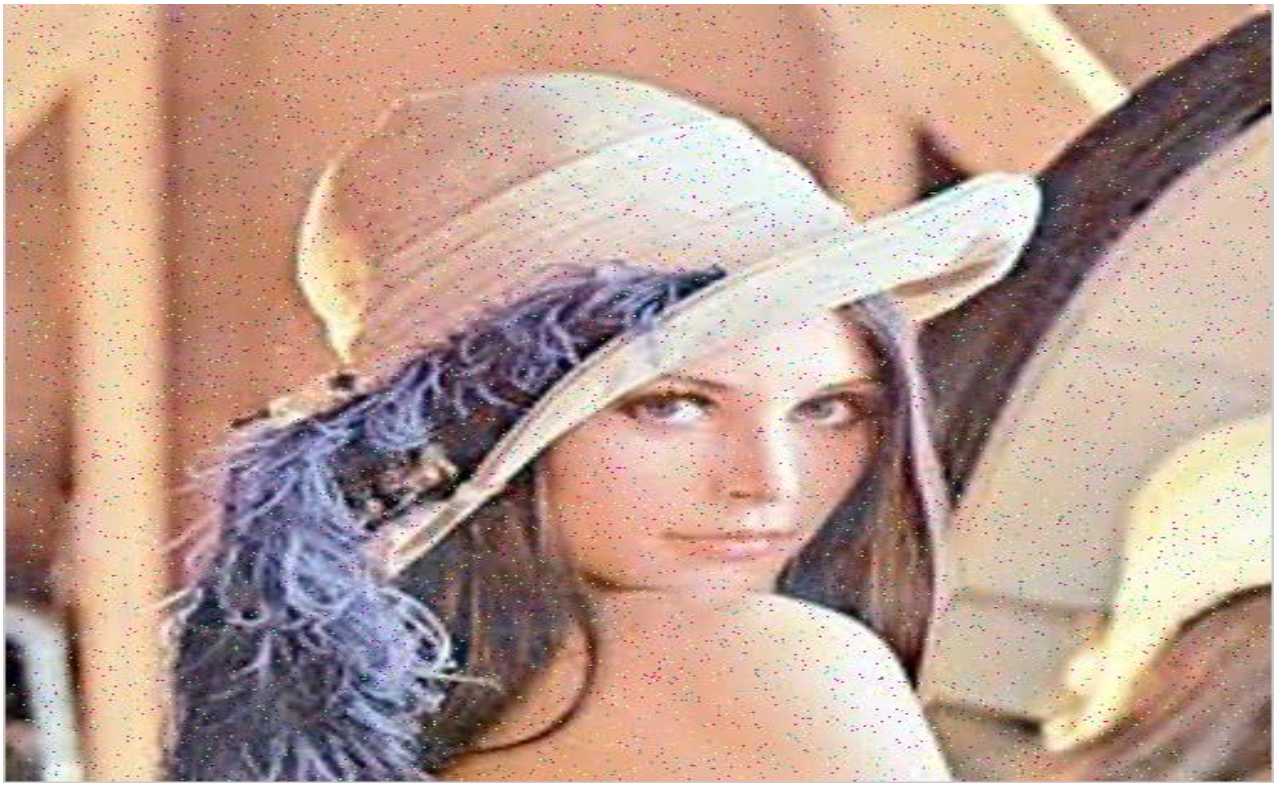


Image 17 : Effect of ‘salt and pepper’ noise on the Watermarked Image.

Image 18 : Effect of ‘salt and pepper’ noise on the Extracted Watermark.

## 5.12 Tabular results for the effect of 'salt and pepper' noise on the Watermarked Image and the Extracted Image in order to preserve the Robustness:

| Scale Factor | 0.01 | 0.03 | 0.05 | 0.07 | 0.09 |
|---|---|---|---|---|---|
| Watermarked Image | 21.99 | 21.65 | 20.40 | 19.88 | 19.20 |
| Extracted Image | 20.49 | 19.40 | 18.96 | 17.32 | 16.99 |

Table: Varying PSNR for 'salt and pepper' noise for different scale factor values.

Image 19 : Effect of ‘speckle’ noise on the Watermarked Image.

Image 20 : Effect of ‘speckle’ noise on the Extracted Watermark Image.

## 5.13 Tabular results for the effect of 'speckle' noise on the Watermarked Image and the Extracted Image in order to preserve the Robustness:

| Scale Factor | 0.01 | 0.03 | 0.05 | 0.07 | 0.09 |
|---|---|---|---|---|---|
| Watermarked Image | 20.98 | 20.32 | 19.93 | 18.67 | 17.60 |
| Extracted Image | 26.60 | 25.89 | 24.88 | 23.78 | 22.55 |

Table: Varying PSNR for 'speckle' noise for different scale factor values.

## 5.14 Results for Edge Detection using Contourlet Transform:

Image 21: Edge Detection using 'sobel' for Watermarked image using Contourlet Transform

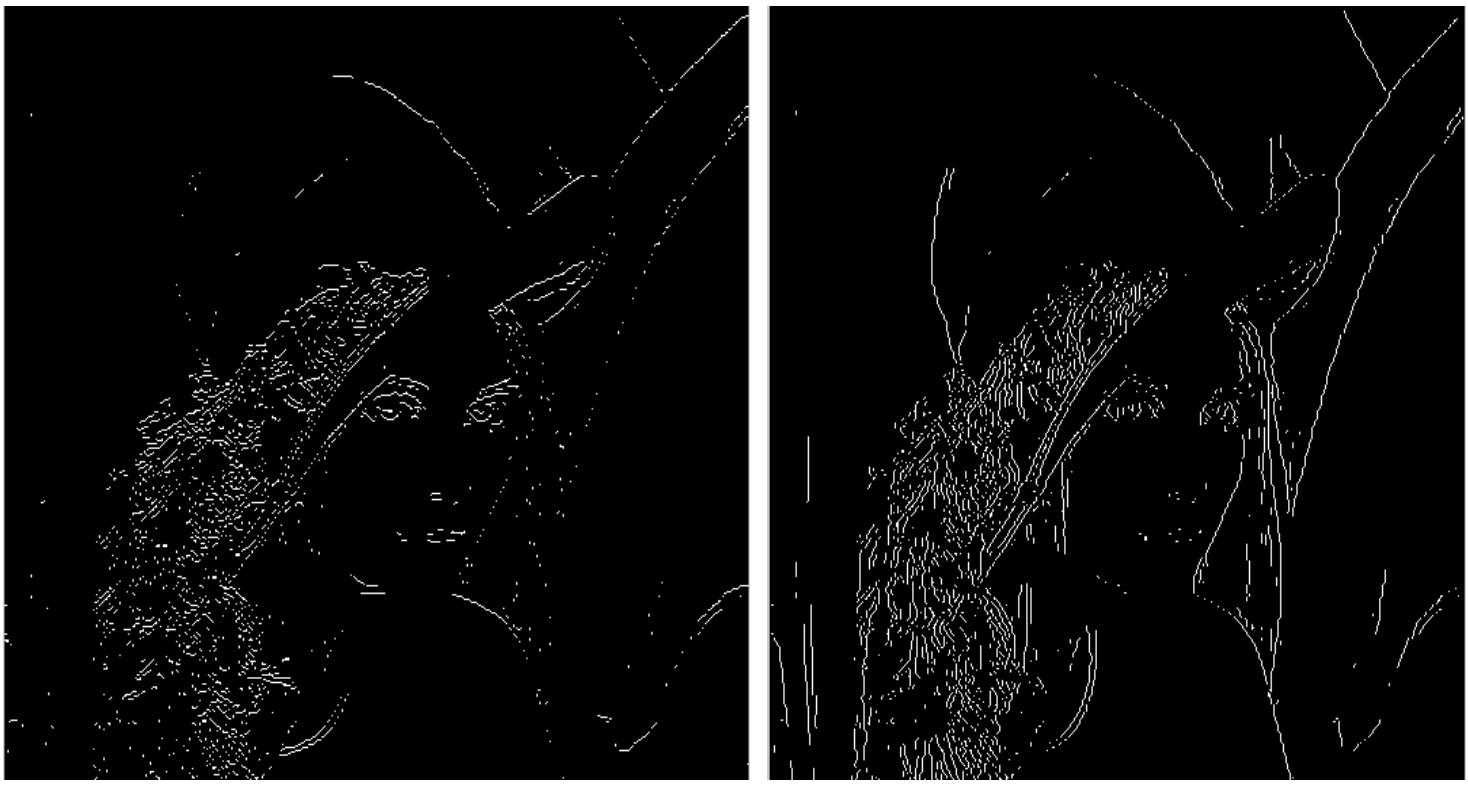


Image 22: Horizontal Edge for Watermarked Image and Vertical Edge for Watermarked Image.

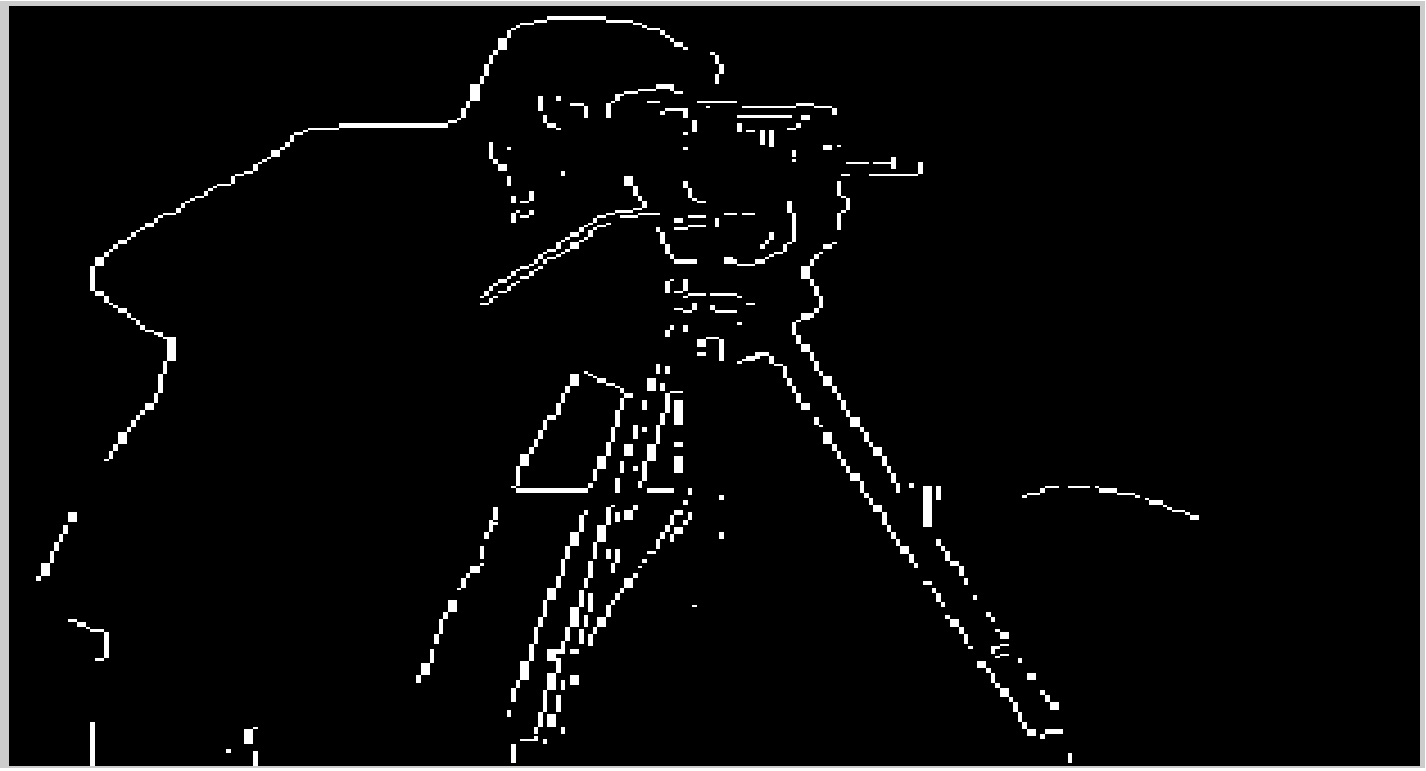


Image 23: Edge Detection using ‘sobel’ transform for the Extracted Watermark done using Contourlet Transform.

Image 24 : Horizontal edge for Extracted Image and Vertical edge for Extracted Image.

## 5.15 Tabular results for the effect of 'edge detection' on the Watermarked Image and the Extracted Image using Contourlet Transform Technique:

| Scale Factor | 0.01 | 0.03 | 0.05 | 0.07 | 0.09 |
|---|---|---|---|---|---|
| Watermarked Image | 4.27 | 3.95 | 3.21 | 2.70 | 2.23 |
| Extracted Image | 15.30 | 14.78 | 14.53 | 13.84 | 13.33 |

Table: Varying PSNR for 'edge detection' for different scale factor values.

# REFERENCES:


[1] J. Cox, M. L. Miller, and J. A. Bloom, "Watermarking applications and their properties," in Proceedings of the International Conference on Information Technology: Coding and Computing (ITCC '00), pp. 6–10, Las Vegas, Nev, USA, 2000.

[2] H.-T. Wu and Y.-M. Cheung,"Reversible watermarking by modulation and security enhancement," IEEE Trans. Instrum. Meas., vol. 59, no. 1, pp. 221–228, Jan. 2010.

[3] F. Hartung and M. Kutter, "Multimedia watermarking Technique", IEEE proceeding on Signal Processing", Volume 87, NO.7, pp.1079-1107, July 1999.

[4] N. Memon,& P.W. Wong (1998). Protecting digital media content. Communications of the ACM, 41(7), 35-43.

[5] J. Dittmann, A. Mukherjee & M. Steinebach, (2000, March 27 - 29). Media-independent watermarking classification and the need for combining digital video and audio watermarking for media authentication. Paper presented at the Proceedings of the international conference on information technology: Coding and computing, Las Vegas, Nevada.

[6] C. I. Podilchuk and E. J. Delp, "Digital Watermarking:Algorithms and Applications," IEEE Signal Processing Magazine, , pp. 33-46, July 2001.

[7] I. J. Cox, M. L. Miller, and J. A. Bloom, "Digital Watermarking", Morgan Kaufmann Publishers, 2002.

[8] ] R. G.vanSchyndel, A. Z. lTrkel and C.F.Osbome, "A Digital Watermark" IEEE, 1994.

[9] J. Sang and M. S. Alam,"Fragility and robustness of binary-phase only filter- based fragile/semi-fragile digital image watermarking," IEEE Trans. Instrum. Meas., vol. 57, no. 3, pp. 595–606, Mar. 2008.

[10] F. Deguillaume, S. Voloshynovskiy, T. Pun, "Secure hybrid robust watermarking resistant against tampering and copy-attack", Signal Processing, vol. 83, pp.2133-2170, October 2003.

[11] Forrest M. Hoffman, "An Introduction to Fourier Theory".

[12] L.B. Almeida, "An introduction to the angular Fourier transform," Acoustics, Speech, and Signal Processing, 1993. ICASSP-93, 1993 IEEE International Conference on , vol.3, pp.257-260, Apr 1993.

[13] L.B. Almeida, "The fractional Fourier transform and time-frequency representations ," Signal Processing, IEEE Transactions on , vol.42, no.11, pp.3084-3091, Nov 1994.

[14] A.I. Zayed, "On the relationship between the Fourier and fractional Fourier transforms," Signal Processing Letters, IEEE , vol.3, no.12, pp.310-311, Dec 1996.

[15] Chen Wen-Hsiung, C.Smith, S.Fralick, "A Fast Computational Algorithm for the Discrete Cosine Transform," Communications, IEEE Transactions on , vol.25, no.9, pp. 1004- 1009, Sep 1977.

[16] Tribhuwan Kumar Tewari and VikasSaxena, ”An Improved and Robust DCT Based Digital Image Watermarking Scheme.”, International Journal of Computer Applications , June 2010.

[17] Ameya K Naik and RaghunathHolambe, “ A Blind DCT Domain Digital Watermarking for Biometric Authentication”, International Journal of Computer Applications, February 2010.

[18] Jianmin Jiang and GuocanFeng , "The spatial relationship of DCT coefficients between a block and its sub-blocks," Signal Processing, IEEE Transactions on , vol.50, no.5, pp.1160-1169, May 2002.

[19] Ken Cabeen and Peter Gent, “Image Compression and the Discrete Cosine Transform”.

[20] Z Yuehua., C. Guixian and D. Yunhai, “An Image Watermark Algorithm Based on Discrete Cosine Transform Block Classifying”, ACM Int. Conf., pp. 234-235, 2004..

[21] Hyesook Lim, Vincenzo Piuri, Earl E. Swartzlander,”A Serial-Parallel Architecture for Two-Dimensional Discrete Cosine and Inverse Discrete Cosine Transforms”, Dec 2000, Vol. 49, pp. 1297-1309.

[22] H. Lim, C. Yim, E.E. Swartzlander Jr., “Finite Word-Length Effects Of An Unified Systolic Array For 2-D DCT/IDCT”, 1996 IEEE International Conference on Application-Specific Systems, Architectures and Processors (ASAP'96).

[23] C. Candan, M.A. Kutay, H.M. Ozaktas, “The discrete fractional fourier transform,” IEEE Transactions on Signal Processing, May 2000, Vol. 48, pp. 1329-1337.

[24] Reduction of discrete cosine transform/ quantisation/ inverse quantization/ inverse discrete cosine transform computational complexity in H.264 video encoding by using an efficient prediction algorithm.

[25] A. Nikolaidis and I. Pitas, “Asymptotically optimal detection for additive watermarking in the DCT and DWT domains,” IEEE Trans. Image Process., vol. 12, no. 5, pp. 563–571, May 2003.

[26] Chen Yongqiang1, Zhang Yanqing2, and Peng Lihua3, “A DWT Domain Image Watermarking Scheme Using Genetic Algorithm and Synergetic Neural Network”, Proceedings of the 2009 International Symposium on Information Processing (ISIP’09) Huangshan, P. R. China, August 21-23, 2009, pp. 298-301

[27] S. Mallat, “The theory for multiresolution signal decomposition: The wavelet representation,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 11, no. 7, pp. 654–693, Jul. 1989.

[28] Andrew B. Watson, Gloria Y. Yang, Joshua A. Solomon, and John Villasenor,”Visibility of Wavelet Quantization Noise”, IEEE Transactions on image processing ,Vol. 6,No. 8, August 1997.

[29] A. Grzeszczak ,M.K. Mandal, S. Panchanathan, “VLSI implementation of discrete wavelet transform”,Dec 1996 Vol. 4, pp. 421-433.

[30] Xiu-bi Wang, “Image enhancement based on lifting wavelet transform,” 4th International Conference on Computer Science & Education, 2009. ICCSE '09., pp. 739-741.

[31] C.M. Patil, S. Patilkulkarani, “Iris Feature Extraction for Personal Identification Using Lifting Wavelet Transform”, International Conference on Advances in Computing, Control, & Telecommunication Technologies, 2009. ACT '09, Dec 28-29, pp. 764 – 766.

[32] O.G Pla, Lin E.T, and Delp E.J, “A Wavelet Watermarking Algorithm Based on a Tree Structure”, Tech. Rep., Polytechnic University of Catalonia, Spain, 2004.

[33] E. Elbasi and A. M. Eskicioglu, “A DWT-based robust semiblind image watermarking algorithm using two bands,” in Security, Steganography, and Watermarking of Multimedia Contents VIII, vol. 6072 of Proceedings of SPIE, San Jose, Calif, USA, January 2006.

[34] X. J. Wu, Hu. Z. Gu, and J.Huang “A Secure Semi-Fragile Watermarking for Image Authentication Based on Integer Wavelet Transform with Parameters”, Tech. Rep., Sun Yat-Sen University, China, , 2005.

[35] Christian Tenllado, Javier Setoain, Manuel Prieto, Luis Piñuel, Francisco Tirado, "Parallel Implementation of the 2D Discrete Wavelet Transform on Graphics Processing Units: Filter Bank versus Lifting," IEEE Transactions on Parallel and Distributed Systems, vol. 19, no. 3, pp. 299-310, Mar. 2008.

[36] M.Antonini, M. Barlaud, P. Mathieu, I.Daubechies, “Image coding using wavelet transform”, IEEE Transactions on Image Processing, VOL. 1, pp. 205 – 220, Apr 1992.

[37] S. Arivazhagan and L. Ganesan, “Texture segmentation using wavelet transform”, Vol. 24, Issue 16, Dec 2003, pp. 3197-3203.

[38] TinkuAcharya and ChaitaliChakrabarti, “A Survey on Lifting-based Discrete Wavelet Transform Architectures”, The Journal of VLSI Signal Processing Volume 42, Number 3,pp. 321-339.

[39] M. Vishwanath, R.M. Owens, "A Common Architecture For The DWT and IDWT," asap, pp.193, 1996 IEEE International Conference on Application-Specific Systems, Architectures and Processors (ASAP'96), 1996.

[40] V. Aslantas, L. A. Dog˘an, and S. Ozturk, “DWT-SVD based image watermarking using particle swarm optimizer,” in Proc. IEEE Int. Conf. Multimedia Expo, Hannover, Germany, 2008, pp. 241–244.

[41] Emir Ganic Ahmet M. Eskicioglu, “Robust DWT-SVD Domain Image Watermarking:Embedding Data in All Frequencies".

[42] G. Bhatnagar and B. Raman,“A new robust reference watermarking scheme based on DWT-SVD,” Comput. Standards Interfaces, vol. 31, no. 5, pp. 1002–1013, Sep. 2009.

[43] E. Ganic and A. M. Eskicioglu,“Robust DWT-SVD domain image watermarking: Embedding data in all frequencies,” in Proc. Workshop Multimedia Security, Magdeburg, Germany, pp. 166–174, 2004.

[44] Q. Li, C. Yuan, and Y.-Z. Zhong,“Adaptive DWT-SVD domain image watermarking using human visual model,” in Proc. 9thInt. Conf. Adv. Commun. Technol., Gangwon-Do, South Korea, pp. 1947–1951, 2007.

[45] R. Liu and T. Tan, “An SVD-based watermarking scheme for protecting rightful ownership,” IEEE Trans. Multimedia, vol. 4, no. 1, pp. 121–128, Mar. 2002.

[46] Luc Lamarche; Yan Liu; Jiying Zhao., "Flaw in SVD-based Watermarking," Electrical and Computer Engineering, 2006. CCECE '06. Canadian Conference on , vol., no., pp.2082-2085, May 2006.

[47] Loukhaoukha, K. Chouinard, and J.-Y, “Hybrid watermarking algorithm based on SVD and lifting wavelet transform for ownership verification” , 11th Canadian Workshop on Information Theory, 2009. CWIT 2009., pp. 177 -182.

[48] XiushanNie, Ju Liu, Xianqing Wang, Jiande Sun, "Watermarking for 3D Triangular Meshes Based on SVD," iih-msp, pp.430-433, 2009 Fifth International Conference on Intelligent Information Hiding and Multimedia Signal Processing, 2009.

[49] V.C. Klema, “The Singular Value Decomposition: Its Computation and Some Applications,” IEEE Trans. Automatic Control, Vol. 25, pp164-176, 1980.

[50] Xiao Bing KANG and Sheng Min WEI, “Identifying Tampered Regions Using Singular Value Decomposition in Digital Image Forensics”, 2008 International Conference on Computer Science and Software Engineering.

[51] V. Klema, A.Laub, “The singular value decomposition: Its computation and some applications”, IEEE Transactions on Automatic Control, Apr 1980, Vol. 25, pp. 164-176

[52] David Gleich, Leonid Zhukov, "SVD based Term Suggestion and Ranking System," icdm, pp.391-394, Fourth IEEE International Conference on Data Mining (ICDM'04), 2004.

[53] S. Esakkirajan, T. Veerakumar, P. Navaneethan, "Best Basis Selection Using Singular Value Decomposition," icapr, pp.65-68, 2009 Seventh International Conference on Advances in Pattern Recognition, 2009.

[54] V. I. Gorodetski, L. J. Popyack, V. Samoilov and V. A. Skormin, "SVD-based Approach to Transparent Embedding data into digital images," International Workshop on Mathematical Methods, Models and Architechtures for Computer network Security (MMMACNS 2001), St. Petersbug, Russia, May 21-23, 2001.

[55] Farid Ahmed, "A dual Fourier-wavelet domain authentication-identification watermark," Opt. Express 15, 4804-4813 (2007).

[56] NikolayPolyak, William A. Pearlman, "Wavelet decomposition and reconstruction using arbitrary kernels: a new approach," icip, vol. 3, pp.866, 1998 International Conference on Image Processing (ICIP'98) - Volume 3, 1998.

[57] YoucaiGao; Jinwei Wang; ShiguoLian; , "Optimum detection for Barni's multiplicative watermarking in DWT domain," Communications and Networking in China, 2008. ChinaCom 2008. Third International Conference on , vol., no., pp.1308-1311, 25-27 Aug. 2008.

[58] K. Raghavendra and K.R. Chetan, "DWT Based Blind Digital Video Watermarking Scheme for Video Authentication", International Journal of Computer Applications ,August 2010.

[59] Zhu-zhiJia, Hong-yu Zhu, Wan-sheng Cheng, "A Blind Watermarking Algorithm Based on Lifting Wavelet Transform and Scrambling Technology," icece, pp.4576-4579, 2010 International Conference on Electrical and Control Engineering, 2010.

[60] DashunQue; Li Zhang; Ling Lu; Liucheng Shi; , "A ROI Image Watermarking Algorithm Based on Lifting Wavelet Transform," Signal Processing, 2006 8th International Conference on , vol.4, no., 16-20 2006.

[61] Miyazaki, A.; Uchiyama, F.; , "An Image Watermarking Method using the Lifting Wavelet Transform," Intelligent Signal Processing and Communications, 2006. ISPACS '06. International Symposium on , vol., no., pp.155-158, 12-15 Dec. 2006.

[62] M. Jayalakshmi, S. N. Merchant, and U. B. Desai, "Digital watermarking in contourlet domain," in 18th International Conference on Pattern Recognition, ICPR 2006, pp. 861–864, August 2006.

[63] M. N. Do, M. Vetterli, "The contourlet transform: An efficient directional multiresolution image representation",IEEE Trans. on Image Process ng, vol.14, pp.2091-2106, December 2005.

[64] M. N. Do and M. Vetterli, “The contourlet transform: An efficient directional multiresolution image representation,” IEEE Transactions on Image Processing, vol. 14, no. 12, pp. 2091–2106, 2005.

[65] S. Zaboli and M. S. Moin, “CEW: A non-blind adaptive image watermarking approach based on entropy in contourlet domain,” in 2007 IEEE International Symposium on Industrial Electronics, ISIE 2007, pp. 1687–1692, esp, June 2007.

[66] D. D.-Y. Po and M. N. Do, “Directional multiscale modelling of images using the contourlet transform,” IEEE Transactions on Image Processing, vol. 15, no. 6, pp. 1610–1620, 2006.

[67] M. N. Do and M. Vetterli, “Pyramidal directional filter banks and curvelets,” in Proceedings of IEEE International Conference on Image Processing (ICIP ’01), vol. 3, pp. 158–161, Thessaloniki, Greece, October 2001.

[68] M. N. Do andM. Vetterli, “Framing pyramids,” IEEE Transactions on Signal Processing, vol. 51, no. 9, pp. 2329 2342, 2003.

[69] S. Phoong, C.W. Kim, P. P. Vaidyanathan, and R. Ansari, “New class of two-channel biorthogonal filter banks and wavelet bases,” IEEE Transactions on Signal Processing, vol. 43, no. 3, pp. 649–665, 1995.

[70] S. Voloshynovskiy, S. Pereira, T. Pun, J. J. Eggers, and J. K. Su, “Attacks on digital watermarks: Classification, estimationbased attacks, and benchmarks,” IEEE Communications Magazine, vol. 39, no. 8, pp. 118–125, 2001.

[71] Tao Huang, Lele Qin, “ Image denoising research based on lifting wavelet transform and threshold optimization”, 2009 3rd IEEE International Symposium on Microwave, Antenna, Propagation and EMC Technologies for Wireless Communications ,pp. 1218 – 1220.

[72] C.I. Podilchuk and W. Zeng, "Image-Adaptive Watermarking Using Visual Models," IEEE Journal on Selected Areas in Comm., vol 16, no. 4, pp. 525-539, May 1998.

[73] F. Hartung and M. Kutter, “Multimedia Watermarking Technique”, IEEE Proceeding on Signal Processing”, Volume 87, NO.7, pp.1079-1107, July 1999.

[74] Stefano Gnavi, Barbara Penna, Marco Grangetto, Enrico Magli, and Gabriella Olmo,” Wavelet Kernels on a DSP: A Comparison between Lifting and Filter Banks for Image Coding”, EURASIP Journal on Applied Signal Processing Volume 2002 (2002), Issue 9, Pages 981-989.

[75] G. H. Golub and C. F. Van Loan, Matrix Computations, 3/e, Johns Hopkins University Press, Baltimore, 1996.

[76] WimSweldons, “The Lifting Scheme: A New Philosophy in Biorthogonal Wavelet Constructions”.

[77] VassiliosSolachidis, Ioannis Pitas, ”Watermarking Polygonal Lines Using Fourier Descriptors”, vol. 24, no. 3, pp. 44-51, May/June 2004 .

[78] Z. Wang, Q. Li, “Information content weighting for perceptual image quality assessment”, IEEE Trans. on Image Processing, vol.20, pp.1185-1198, May 2011.