

Content Based Fragile Watermarking Scheme
for
Image Authentication and Error Localization

By

Mufti Tausif – ur – Rahman
MSc 45
2005-NUST-MSc PhD- ComE-05

Submitted to the Department of Computer Engineering in
Partial Fulfillment of the requirements for the Degree of
Master of Science
In
Computer Systems

Thesis Supervisor

Brig Dr Muhammad Younis Javed
Head of the Department of
Computer Engineering

College of Electrical and Mechanical Engineering

Acknowledgement

First and foremost, I am grateful to Allah Almighty, without HIS countless blessings it would not be possible for me to take even a single step towards completion of this intricate work. Then, I am thankful to my parents and wife, their continuous prayers and support have helped me in every walk of my life including this research work. I am deeply indebted to all my teachers, especially my project supervisor, Brig Dr Younis Javed whose guidance, support and continuous encouragement made it possible for me to take this mammoth task to the end

Table of Contents

Chapter 1: Introduction	7
1.1 Introduction.....	7
1.2 General Description of Watermarking System	9
1.3 Salient Desirables of Watermarking	10
1.3.1 Capacity	11
1.3.2 Imperceptibility	11
1.3.3 Security	12
1.3.4 Robustness	13
1.3.5 Reversibility	14
1.4 Watermarking Applications.....	14
1.4.1 Validation, verification and authentication.....	15
1.4.2 Patent Safeguard.....	15
1.4.3 Footprint Tracing.....	16
1.4.4 Duplication Prohibition	16
Chapter 2: Basic Watermarking Schemes, Types and Attack Classifications	18
2.1 Introduction.....	18
2.2 Basic Watermarking Schemes.....	18
2.2.1 Informed / Blind Detection Schemes.....	18
2.2.2 Multiple embedding	19
2.2.3 Public and Private watermarking Schemes .	19
2.3 Types of Watermarks.....	20
2.3.1 Robust Watermarks.....	20
2.3.2 Fragile Watermarks.....	21
2.4 Attack Classifications.....	23
2.4.1 Watermark elimination.....	24
2.4.2 Host media forgery.....	26
2.5 Secret Information.....	27

2.6	Security Attacks Categories in General.....	28
Chapter 3: Literature Review and various approaches.....		
3.1	Introduction.....	29
3.2	Reviews of some approaches.....	30
3.3	The proposed Scheme's approach.....	33
Chapter 4: Design and Implementation of proposed content based fragile watermarking scheme.....		
4.1	Introduction.....	36
4.2	Design, implementation and functional details.....	36
4.2.1	Module Block Diagrams	37
4.2.1.1	Embedding block diagram description.....	37
4.2.1.2	Extraction/Authentication block diagram description.....	40
4.2.2	Proposed scheme's algorithms and flowcharts.....	42
4.2.2.1	Sender End Algo.....	42
4.2.2.2	Receiver End Algo.....	44
4.2.2.3	Embed Watermark A & B algo.....	46
4.2.2.4	Image Segmentation algo.....	48
4.2.2.5	Long Division.....	49
4.2.3	Proposed Scheme's Detailed functional description.....	51
Chapter 5: Evaluation results and discussion.....		
5.1	Technical Parameters	55
5.2	Testing Methodology	55
5.3	Image Quality Computation	56
5.4	Images and their histograms	57
5.5	Detailed evaluation and discussion.....	59
5.6	Transmitted and Watermarked Images	68
5.7	Tamper Detection and Error Localization capabilities.....	71

Chapter 6: User's Manual	78
6.1 Introduction.....	78
6.2 Functional Description.....	78
6.2.1 Sender End Module.....	78
6.2.1.1 Screen Description.....	79
6.2.2 Receiver End Module.....	85
6.2.2.1 Screen Description.....	86
Chapter 7: Conclusions and Future Works	91
7.1 Conclusions.....	91
7.2 Future Work.....	93
Bibliography	95

List of Figures

	<u>Figure Description</u>	<u>PAGE</u>
Figure 1.1	A general watermarking system.....	9
Figure 1.2	Relationship of Capacity, Imperceptibility and.....	12
Figure 2.1	Security Attacks Categories in General.....	28
Figure 4.1	Embedding block diagram.....	39
Figure 4.2	Extraction / authentication block diagram.....	41
Figure 4.3	Sender end flow chart.....	44
Figure 4.4	Receiver end flowchart.....	44
Figure 4.5	Watermark embedding flowchart.....	46
Figure 4.6	Image segmentation flowchart.....	48
Figure 4.7	Long Division.....	49
Figure 4.8	Calculation of 128 bit stream	52
Figure 4.9	Calculation of 32 bit stream	53
Figure 5.1	Camerman and its Histogram.....	57
Figure 5.2	Cell and its Histogram.....	57
Figure 5.3	Moon and its Histogram.....	58
Figure 5.4	Rice and its Histogram.....	58
Figure 5.5	Coin and its Histogram.....	58
Figure 5.6	Graph – Image PSNR versus Security Level	59
Figure 5.7	Graph – Comparison of proposed scheme.....	60
Figure 5.8	Watermarking Time(secs) versus Security Level.....	61
Figure 5.9	Watermarking Time(secs) versus Image Size.....	62
Figure 5.10	PSNR versus Image Size.....	63
Figure 5.11	Watermarking Time(secs) versus Generator Polynomial Length.....	64
Figure 5.12	PSNR versus Generator Polynomial Length.....	65
Figure 5.13	Watermarking Time(secs) versus Different Image Formats.....	66
Figure 5.14	PSNR versus Different Image Formats.....	67
Figure 6.1	Sender End – Main Screen.....	79

Figure 6.2	Prompt screen for selection of Input Image.....	81
Figure 6.3	Prompt screen for Saving the Watermarked Image.....	82
Figure 6.4	Screen showing successful Termination of the Watermarking Process.....	83
Figure 6.5	Screen showing the Receiver End Module.....	86
Figure 6.6	User Prompted to Enter Path of Received image File.....	87
Figure 6.7	User Prompted to Enter Path of Received image File.....	88
Figure 6.8	Screen Shot on Receipt of a Tampered / Modified Image.....	89

ABSTRACT

With the availability of some powerful image processing softwares such as 'Adobe Photoshop', one can remove/replace some features in a picture easily without any detectable trace. This type of operation may be regarded as tamper. However, in some cases, a person or an organization cannot afford an image to undergo any such operation, such as images for military, medical or judicative use etc. The validity and authenticity of an image is of utmost importance in such cases, so there is a need to guarantee the integrity of an image in an effective manner. Also in the second phase, one must be able to localize and identify an area which has been tampered or has developed some sort of an error while passing the transmission channel or during storage/retrieval process.

This thesis presents an image authentication and error localization technique, based on fragile water marking. The watermark is content based and the error localization method uses a "Cyclic Redundancy Checksum" to authenticate image as well as localize any channel induced or intentionally introduced error, down to block level (2x2 pixels). The scheme has been tested on various image formats as well as image sizes and has successfully localized error/tampering with good PSNR values.

1.1 **Introduction**

The continuous and fast paced advancements in the field of technology is being used by the human race both for the betterment as well as the destruction of this world. Every invention and development in the field of science is thus critically analyzed and subsequently used by both these school of thoughts to achieve their entirely opposite aims.

Digital recording, processing, manipulation and storage constitute one such area which has seen enormous amount of changes in the past few years. Very powerful and comprehensive software tools are now available which have not only vastly improved the capturing of image but also provide a very elaborate system of subsequent processing of these images, encompassing primarily their enhancement, improvement, manipulation and in some cases even merger as well. Moreover, the storage capacities and storage media have also come a long way. One CD can now store a colossal amount of multimedia files and clips. However, on the other hand the same power computer and softwares is also being negatively used by some people in order to distort, manipulate or even claim ownership of various images. Such nefarious activities have found their way chiefly in the following fields:

- a. **Medical image archiving.** The authentication data of the patients can be embedded at the time when their medical images are taken by the hospital to keep patient's case histories. However, when medical malpractices happen, cases on the basis of these records have to be resolved in court.

- b. **Imaging / sound recording of criminal events.**
Authentic imaging or recording of legally essential event or conversation could lead to breakthrough in criminal cases while maliciously tampered imaging / recording, if not detected, could result in wrong ruling.

- c. **Accident scene capturing** For insurance and forensic purposes, and for protecting the rights of the parties including the insurance company involved in accidents or natural disasters.

- d. **Broadcasting** During international crises, tampered or forged media could be used for propaganda and tilting public opinion. Therefore, broadcasting is an area where multimedia authentication is required.

- e. **Military intelligence** Multimedia authentication allows the military to authenticate whether the media they received do come from the legitimate source and to verify whether the content is original. Should the content be manipulated, an effective authentication scheme is expected to tell as much information about the manipulation (e.g. location of tampering) as possible.

It is now becoming more and more important to have the capabilities to ascertain the authenticity and originality of various images. Much work is being done in this field to ensure the rights of ownership. This thesis is the result of research, carried out with this backdrop in mind. Some of the specifics have been highlighted in

the abstract, whereas, the detailed methodology, concept and experimental results shall be covered in the subsequent chapters.

1.2 General Description of a Watermarking System

A watermarking system consists of three main components i.e. an embedder, a communication channel and a detector. The watermark information is embedded within the host signal before the watermarked signal is transmitted over the communication channel, so that the watermark can be detected at the receiving end, that is, at the detector. The mutilations found at the detector, if any, represent the modification applied to the watermarked signal when it is going through the communication channel

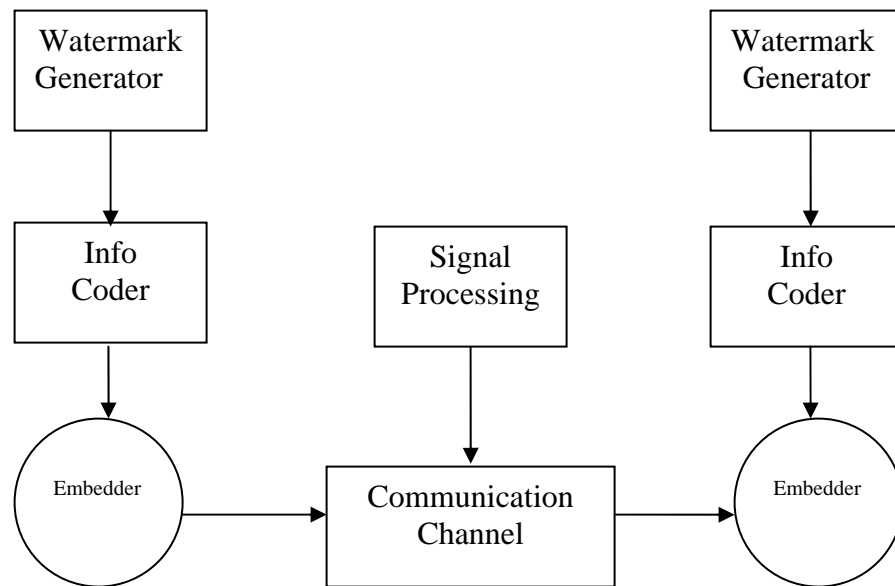


Figure 1.1: A general watermarking system

A general watermarking system framework is illustrated in Figure. 1.1 First of all, a watermark W_0 is generated by the

watermark generator possibly with a secret watermark generation key K_g . The watermark W_o can be a logo, or be a pseudo-random signal. Instead of directly embedding it into the host signal, the watermark W_o can be pre coded to optimize the embedding process, i.e. to increase robustness against possible signal processing operations or imperceptibility of the watermark. This is done by an information coder which may require the original signal S_o . The original signal S_o can be of any multimedia file, such as audio file, image file including text image and 3D image, video file or a combination Information of them. The outcome of the information coding component is denoted by symbol W that, together with the original signal S_o and possibly a secret key K , are taken as input of the embedder. The secret key K is intended to differentiate between authorized users and unauthorized users at the detector in the absence of K_g . The embedder takes in W , S_o and K , so as to hide W within S_o in a most imperceptible way with the help of K , and produce the watermarked signal S_w . Afterwards, S_w enters into the communication channel where a series of unknown signal processing operations and attacks may take place. The outcome of the communication channel is denoted by the symbol S'_w . At the receiving end, the detector works in an inversely similar way as the embedder, and it may require the secret key K_g , K , and the original signal S_o . Then the detector reads S'_w and decides if the received signal has the legal watermark.

1.3 Salient Desirables of Watermarking

The desirable qualities of watermarking system can be more specifically defined while taking into account the requirements of the system being designed. They may differ or vary from system to system. Moreover, since we are not living in an ideal world so some

impairments may be introduced in the signal because of D/A,A/D conversion, processing and compression/decompression techniques etc. In spite of all these factors some of the salient desirables of a watermarking system are discussed in the succeeding paras.

1.3.1 Capacity

Watermarking capacity refers to the amount of information that can be embedded into a host signal. This determines whether a technique can be feasibly used in a given scenario or not because of the fact that capacity for a certain technique can vary greatly among different signals, and even among sections in the same signal. This unevenly distributed embedding capacity is not desirable but only few solutions are proposed in the literature [1]. It is often a very desirable feature of an algorithm that it is flexible in adjusting capacity thus a good tradeoff can be found depending on the application at hand. Indeed, as long as expected capacity is achieved, efforts can be spent on optimizing other requirements.

1.3.2 Imperceptibility

This is one of the most important requirements of the watermarking system. It implies that the embedded watermark is too transparent to be detected visually. Ideally, no perceptible difference between the watermarked and original signal should exist [2, 3]. It is, therefore, desirable to design schemes that minimize the perceptual difference between the original signal and the watermarked signal.

1.3.3 Security

Watermarking security implies that the watermark should be difficult to remove or alter without damaging the host signal. The security requirement of a watermarking system can differ slightly depending on the application. A secure watermarking system is able to assure secrecy and integrity of the watermark information, and resist attacks. The attacks may be aimed at damaging the watermark, such as feature removal and addition in watermarked signal, or modifying the image differently like common image enhancement etc[4]. The importance and implication of each type of attack is different from application to application.

Each of the three important requirements of capacity, imperceptibility and security always struggle against two other important requirements. A higher capacity is usually obtained at the expense of either security or imperceptibility, or both. Same is the case with the other two factors. **Figure 2.2** shows this concept

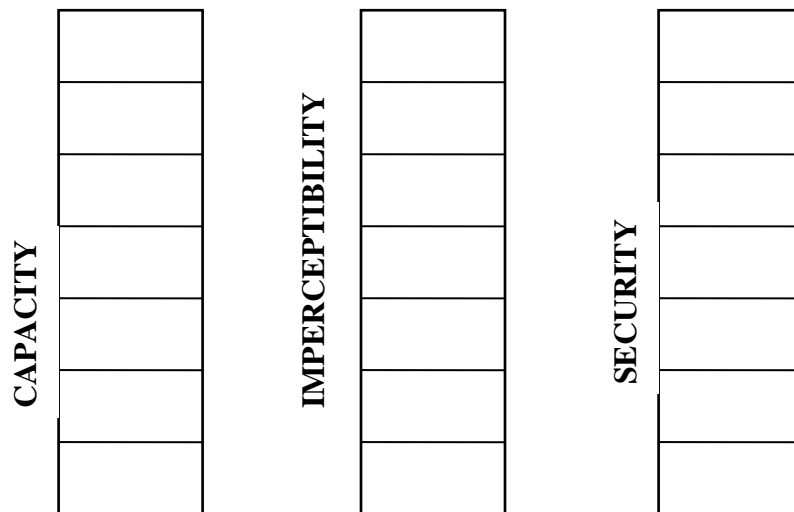


Figure 1.2: Relationship of Capacity, Imperceptibility and Security

From the above figure it is evident that we have specific number of building blocks available. If there is a requirement to enhance one of the factor in our application, one has to borrow the additional blocks from any of the other factors thereby weakening or reducing that aspect.

1.3.4 Robustness

Robustness accounts for the capability of the watermark to survive signal manipulations. In addition to attacks, even common signal processing operations pose a threat to the detection of watermark, thus making it desirable to design a watermark that can survive those operations. A good strategy may be embedding of a watermark into perceptually significant parts of the image. Thus, robustness is guaranteed when we consider the case of lossy compression which usually discards perceptually insignificant data, thus data hidden in perceptual significant portions is likely to survive lossy compression operation. On the other hand, as this portion of the host signal is more sensitive to alterations, watermarking may produce visible distortions in the host signal. There are some applications where imperceptibility requirement outweighs the robustness requirement. For example, a project for authenticating cultural heritage images would require special care for preserving the perceptual quality of images and can tolerate low level robustness. The exact level of robustness an algorithm must possess cannot be specified without considering the application scenario [5]. Not all watermarking applications require a watermark to be robust enough to survive all attacks and signal processing operations. Indeed, a watermark needs only to survive the attacks and those signal processing operations that are likely to occur during the period when the watermarked signal is in communication

channel. In an extreme case, robustness may be completely irrelevant in some case where fragility is desirable.

1.3.5 Reversibility

One of the most frequent uses of watermarking is the protection of copyrights and contents. However, watermarking may inflict degradation on the original signal. This is not permitted for the applications in medical and military fields which require highest precision. Therefore, reversible watermarking becomes important and is drawing more interests recently. Reversible watermarking implies watermarking schemes capable of successfully removing the watermark and thus restoring the watermarked signal to its original state.

Some of the early reversible schemes employ additive spread spectrum techniques that embed a spread spectrum signal as watermark information into the host signal , some latest reversible systems [6, 7] employ compression functions in the embedding stage. In most of the cases, reversible embedding increases the file size of lossless compressed (GIF etc...) signal except from a few recent techniques. Also, it adds to the computational expense, considerably complicates the application design when robustness and security of the watermark is of main concern, since it implies that only trusted users should be allowed to remove the watermark [1].

1.4 Watermarking Applications

Watermarking is finding more and more uses. A previous increase in watermarking interest was most likely due to the increase in concern over copyright protection. A renewed interest is also triggered by the evidently promising use of watermarking for multimedia authentication purpose. Some of the fields effectively

using watermarking for various purposes are discussed in the succeeding paras :

1.4.1 Validation, Verification and Authentication

A diverse variety of powerful signal processing tools can be used effectively modify the visual or audio content of digital signals without leaving any perceptible traces, which causes the loss of signal credibility. A number of countermeasures have been taken to authenticate digital signals, i.e. to verify their integrity and that they have not been intentionally tampered with. Digital watermarking becomes a promising and powerful solution to both of the above problems with its unique merits. In watermarking authentication the mark is modified where the host is modified, which opens up the possibility of learning more about which portions, and even how the watermarked signal has been tampered with. Another potential benefit of watermarks is its use to trace attacks. Since the watermark undergoes the same manipulation as the host signal, it is possible for us to learn something about the manipulation the signal has undergone. Therefore, from a theoretical point of view, the embedding capacity has to be high to accommodate these needs. The security against tampering the content of the signal or forging a valid watermark in an unauthorized signal is of concern. Authentication watermarking requires the lowest robustness, sometimes just robust enough to certain modifications like compression.

1.4.2 Patent Safeguard

Illegal distribution of any kind of content causes a great loss to the rightful owner. Phenomenal use of internet has aggravated this problem manifolds. Watermarking for copyright protection is urged by the popularization of internet where digital multimedia files

may be distributed by illegal users. Copyright information such as company logo and relating information is embedded as whole or part of the watermark into the host signal. Watermarking for copyright protection expects embedded watermarks to survive various kinds of manipulations provided that the resulting files are still acceptable in terms of commercial value. Hence watermarking schemes for copyright protection are typically robust, while different levels of robustness depend on the specified application requirements. Because robust watermarks can be both imperceptible and inseparable from the host signal, it is practical to supply multimedia files that are embedded with robust watermarks to enable owner identification or proof of ownership. Schemes for copyright protection require the highest level of robustness, while the total embedding capacity does not have to be high in most scenarios. As long as enough copyright information is embedded, efforts can be made on optimizing other requirements.

1.4.3 Footprint Tracing

This is another important application and an aid for the protection of Copyright. Digital watermarks can help identify legal recipients of the file, and thus track if the file content left the authorized distribution path or not. If it is discovered that the file has been illegally distributed, the watermark can indicate the person responsible by tracking to the last authorized recipient. To enable transaction tracking, the content owner needs to maintain a detector and database. Some elaborate setup is required for that but it is worth the effort in the long run.

1.4.4 Duplication Prohibition

A kind of digital watermark, embedded within content, can indicate if play out or copying is permitted or not. By using

compliant devices, the embedded information can determine whether playing or copying is allowed. Thus illegal DVD burning or loading content for unauthorized distribution over the internet can be prevented. This is an efficient way to prevent illegal copies. However, fully functioning solutions have not yet been entirely approved by global producers although hectic efforts are underway in view of the importance of the issue.

Basic Watermarking Schemes, Types and Attack Classifications

2.1 Introduction

From previous chapters it may be seen that security, capacity and imperceptibility are all inter-linked. Increasing one of these factors is done at the cost of any of the other two. When a system is being designed, these factors are prioritized in view of the user requirement.

At this stage, it seems pertinent to discuss the basic approaches in the field of watermarking. This discussion shall also encompass the basic types watermarks and classifications of attacks in this regard, keeping in view the purview of the system being discussed, thus justifying the use of a fragile watermark.

2.2 Basic Watermarking Schemes

Now we look at some of the basic watermarking schemes being followed in the field of watermarking

2.2.1 Informed / Blind detection Schemes

If a watermarking scheme does not resort to the original signal in the watermark extraction process, it is blind watermarking. Conversely, a watermarking scheme is said to be informed if the extraction process needs the original signal to be available at the detector. In terms of capacity or security, blindness causes no ill

effect on performance. But it does feature loss of robustness [5]. Informed schemes are impractical when the availability of original signal cannot be warranted. For example, in authentication applications, obtaining original signal is unrealistic.

2.2.2 Multiple embedding Schemes

There may be requirements where it is desirable to insert two or more watermarks into a host signal to address more applications [5]. For example, if two robust watermarks are embedded in an image, one in lower frequency band and the other in higher frequency band, the watermarked image would be robust against a wide spectrum of image processing operations. In a joint owner identification and content image authentication project, each image contains two watermarks: a robust watermark with the identification information of the copyright owner and a fragile watermark with authentication information. Instead of concatenation, these watermarks should be layered up with fragile watermark lying upon robust watermark [8]. The insertion of multiple watermarks should not seriously deteriorate the quality of the host signal. Note that the applications that require robustness should not permit the situation that a pre-existing watermark is rendered undetectable after a new watermark is inserted. If this were the case, watermark insertion would be the most effective mean for an adversary to destroy any existing watermark without damaging the host signal. Multiple embedding could result in a more robust and higher capacity scheme, but it may incur more distortions.

2.2.3 Public and Private watermarking Schemes Category of public scheme implies those schemes which allow the general public to verify the watermark or in which a public key can be used

for extraction. On the other hand if only authorized users, for instance the holder of secret key, can detect or extract the watermark, the scheme is said to be private. While the public key can only be used to verify the watermark, the secret key is valid for embedding and extracting watermark. Similar to public key cryptography systems [2], public watermarking schemes do not need to transmit the secure key for verification. This is necessary in some applications where exchange of key is not possible. But on practical level, private systems may face a security risk. As the secret key is known to the widely spread detector devices and the fact that the security of a scheme is based on the secrecy of secret key instead of the obscurity of the algorithm, it is likely that attackers will take advantage of the available information and crack the whole system. However, usually public watermarking schemes are computationally more expensive.

2.3 Types of Watermarks

Only the basic types which shall be discussed in the succeeding paragraphs:

2.3.1 Robust Watermark

As the name implies, a robust watermark is the one that can withstand any kind of channel induced or intentional manipulations. During the time the watermarked signal is in the communication channel, manipulations and transmission distortion could apply to it. In the decoding stage, a watermark is expected to be reliably extracted from the received signal and embedded information is decoded.

The reliable extracting probability is important in helping estimate the level of security of a system. However, instead of designing a watermark that can resist all kinds of manipulations, a

specific scheme which can survive the manipulations that are likely to happen between the time of embedding and detection is more desirable. Since robustness as well as security is usually achieved at the cost of other requirements, it is not necessary to design a general system that is suitable for all applications. When possible manipulations are identified, a strategy can be chosen from a variety of the robust watermarking approaches.

Although robust watermarking is intentionally designed for copyright protection, it can also be used for data authentication. Essentially, a summary of the host signal is computed and inserted within the signal as a robust watermark. Origin information such as company logo can be also embedded if needed. At the receiving end, the watermark is extracted and used to authenticate and to prove integrity of the received signal. The mismatch or watermark absence, is taken as evidence of tampering.

Not all watermarking applications require a watermark to be robust enough to survive all attacks and signal processing operations. Indeed, a watermark needs only to survive the attacks and those signal processing operations that are likely to occur during the period when the watermarked signal is in communication channel. In an extreme case, robustness may be completely irrelevant in some case where fragility is desirable.

2.3.2 Fragile Watermark

The naming of this watermark is fragile is because of the basic characteristic of this watermark. A watermark is said to be fragile if the watermark hidden within the host signal is destroyed as soon as the watermarked signal undergoes any manipulation.

When a fragile watermark is present in a signal, it may be inferred, with a high probability, that the signal has not been altered. On the other hand, with a robust watermark, watermark loss is

taken as evidence that the signal has been tempered with, whereas the successful extraction of the hidden data is used to prove data integrity and, if needed, to analysis the attacks.

Based on quite different mechanisms as compared to robust watermarking authentication, fragile watermarking authentication has an interesting variety of functionalities including tamper localization and discrimination between malicious and non-malicious manipulations. Tamper localization is critical because knowledge of where the image has been altered can be effectively used to indicate the valid region of the image, to infer the motive and the possible adversaries. Moreover, the type of alteration may be determined from the knowledge of tampering localization.

In image watermarking, the schemes for tamper localization fall into two categories i.e. block-wise watermarking and pixel-wise watermarking. Block-wise watermarking divides an image into contiguous blocks and independently embeds the signature into each block, thus detects tampering with a resolution of the block size. When the block size is reduced to the size of one pixel, i.e. in pixel-wise watermarking, tampering localization can be traced to pixel resolution. It may seem desirable to use pixel-wise authentication which leads to more precise resolution. However, there exists a tradeoff between localization resolution and system security. The security may have to be compromised as the block size reduces, with greatest risk in the extreme case: the pixel-wise authentication. For example, in the counterfeiting attack , the adversaries own a watermarked image database marked with the same secret key and can thus create a counterfeit by swapping identically positioned blocks from different images. This attack can be effectively countered by selecting a larger block at the expense of the location accuracy.

Fragile watermarking is intolerant not only to malicious attacks but also to non malicious attacks which do not change the semantics of the media. Semi-fragile watermarking scheme is designed for better tolerance, since it is sometimes necessary that the hidden information survives a certain kind of allowed manipulations. These manipulations, normally content-preserving operations, are presented in many multimedia systems wherein fragile watermarking is not practical. While content-preserving manipulations are allowed, malicious operations such as removal of some features from the signal are still intolerable. An up-to-date review of semi-fragile watermarking algorithms can be found in [4]. In the case of fragile watermarking, watermark removal is not a pressing security problem, because the watermark is so fragile that it is easily distorted. On the contrary, adversary would be interested in modifying the host data without leaving any trace.

2.4 Attack Classification

As a large variety of attacks have been proposed, it is desirable to classify them into categories for better understanding their purposes and finding solutions. A straight forward way is to classify them according to the main objectives of the attackers which are, namely, watermark elimination and host media forgery. As its name implies, watermark elimination aims to remove watermark without leaving any trace while host media forgery aims to produce fake signals that can be correctly read by detector. Obviously, watermark elimination is explicitly designed for attacking robust watermarking systems while host media forgery is mounted against fragile watermarking systems. In the following section, the objectives and means that differ in two kinds of attacks without delving into details of specific attacks shall be studied. This viewpoint of essence study of attacks helps in identifying sets of

attacks an application may face, so that solutions can be systematically found.

2.4.1 Watermark elimination

A watermark is considered to be eliminated or removed from a signal if it is rendered undetectable. In all watermarking applications that require high level of robustness, it is necessary to prevent unauthorized removal and unauthorized insertion. To be more specific, an adversary against robust watermarking systems would attempt to eliminate copyright information so that the watermark cannot be recognized by detector, where after he may insert his own illegal watermark. Basically, there are four means to mount watermark elimination attacks: watermark estimation without cracking the security of the algorithm, investigation into the system weaknesses, watermark-detector synchronization distortion, and cracking security methods of the watermarking system. There are many ways to break in a watermarking system by estimating watermark features, such as the watermark generation key. Instead of watermark removal, an attacker may be more interested in watermark estimation so that the watermark information will be helpful in mounting future attacks. For example, if watermark estimation succeeds, the available information can be used to trick the detector and introduce ambiguity to the proof of ownership. The attacker may then subtract his watermark from the signal to create his own fake one and make the same ownership claim as the owner. A countermeasure to this attack is to generate the watermark from a signature of the original signal. Image semantic can be used to obtain a good estimate of the original image and thus gain access to the watermark. If the attacker has at his disposal several watermarked signals, he can estimate features about the watermark with higher possibility. The main concept is

based on the fact that some prior knowledge of the signal is available. A watermark must comply with the system requirements that maintain normal functioning of the system. But any weakness originated from these requirements or their combinations could endanger the secrecy of the watermark, since the weakness could be exploited by the adversary to circumvent the embedder or detector. For instance, the attack can be based on the requirement of watermark reversibility [10]. In particular, no matter the reversible watermark is additive or multiplicative the whole process could be perfectly inverted once a potential weakness is found. It has also been proved that for copyright protection applications. Further than making the watermark dependent on signal content, the schemes resistant to this reversibility attacks need to use one-way hash functions as well. In watermarking context, synchronization refers to the fact that detecting process finds the correct watermark information in the received signal. To disturb the synchronization between watermarks and detector, commonly used methods such as geometric attacks, temporal attacks and noise removal attacks have been successfully mounted. Most synchronization attacks are performed on video and audio multimedia signals to which synchronization is more important. In particular, geometric manipulation represents a low-cost yet effective attack to most existing watermarking schemes. As the security of a system is based on the choice of the secret key and possibly other secret parameters, instead of the secrecy of the algorithm, attackers may try to crack the system by exploring these key elements. When the watermark signal is generated from a generator, e.g. a linear sequence generator, then it is very possible that the generator parameters can be estimated from a few samples of the watermark signal. However, this attack will not succeed if the watermark generation is also dependent on the original signal. For example, a

hash value of the signal can be used and the hash can be either robust or fragile depending on the application requirements.

2.4.2 Host media forgery

As to the fragile watermarks for authentication and proof of integrity, the attacker is no longer interested in making the watermarks unreadable. Actually, disturbing this type of watermark is easy because of its fragility. The goal of the attackers is, conversely, producing a fake but legally watermarked signal. This host media forgery can be reached by either making undetectable modifications on the watermarked signal or inserting a fake watermark into a desirable signal. Many image watermarking attacks explore the block-wise method adopted by most of the fragile watermarking schemes. One of them is the Holliman-Memon attack or cut-and-paste attack [31] in which the blocks are exchanged either within one image or among multiple authenticated images. Alternatively, the attackers can aim at any watermark information leakage due to security flaw. As in watermark elimination, watermark estimation can help implement a copy attack, in which case the attacker tries to copy a watermark from one signal to another, hence the attacker has multiple pairs of original and watermarked signal, it is even more possible that watermark estimation would succeed. System weakness can also be exploited in host media forgery in forming an effective method to crack a watermarking system. For example in asymmetric systems where an attacker can have unlimited access to the detector, he can submit slightly modified versions of the authentic signal for verification until a modified signal passes the detector. The more devices or information publicly available, the easier it is for the attack to succeed. Indeed, the possibility of success depends on

the specific application scenario in which the watermarking scheme is served.

2.5 Secret Information

Although security requirements of a watermarking system vary greatly from application to application, it is impossible to discuss security issues in every aspect. In general, all attacks are driven by taking advantage of available knowledge about the watermarking system, even when the knowledge was supposed to be secret. It is important to understand the type of information that may be disclosed to attackers before effective measures can be used to prevent information leakage. Given the Kerckhoff's principle [9] which states that the security of a system cannot be based on the secrecy of the algorithm, it is highly recommended for a system designer to be aware of the potential danger of disclosing secret information, such as the choice of key. Here are three major types of secret information:

- The secret information about watermark, including the watermark generation key.
- The secret information about the embedding parameters.
- The secret information about the detecting/extracting parameters.

In practice, keeping information secret has always been difficult, thus it is preferable that not too much information needs to be kept secret. However, from the view point of the system designer, the more information is kept secret, the easier it is for him to design the system as he has only to care about fewer publicly-known things.

2.6 Security Attacks Categories in General

There are four categories of attacks:

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

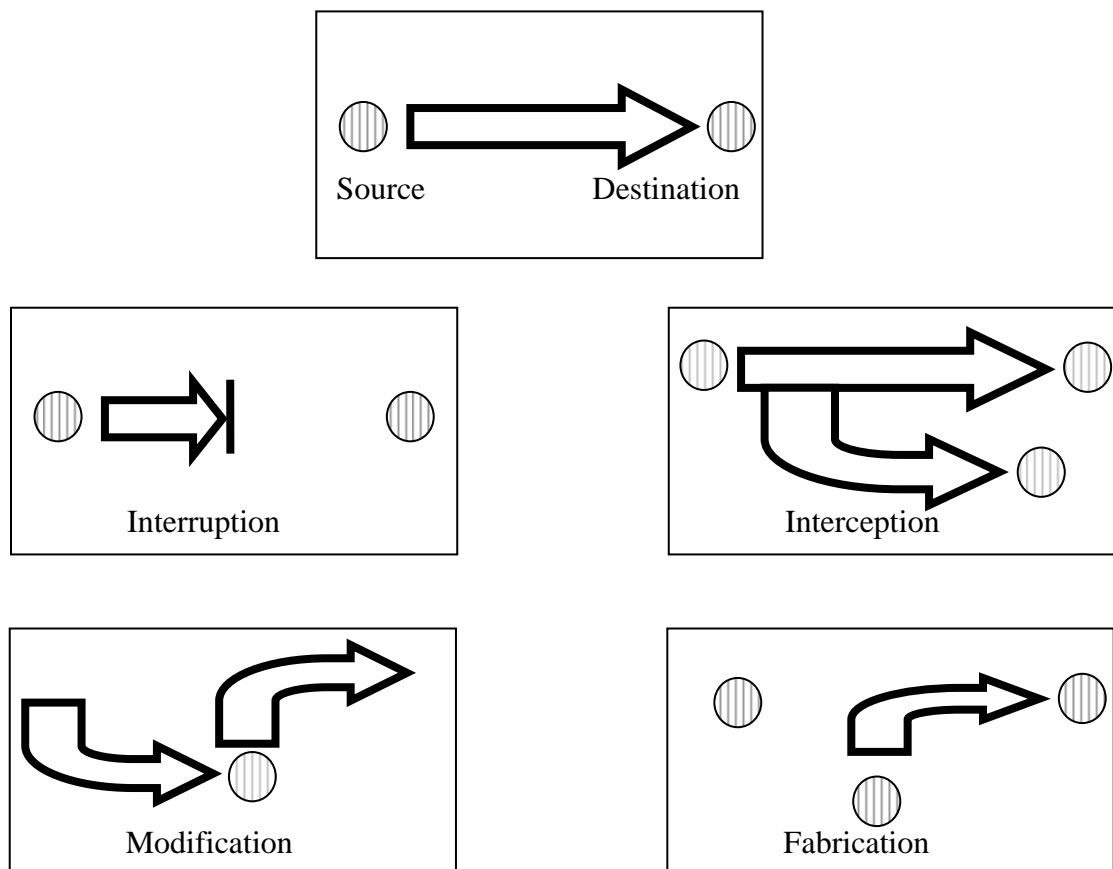


Figure 2.1 Security Attacks Categories in General

Literature Review and Various Approaches

3.1 Introduction

The Internet is not a secure medium. The security of digital images has thus always been a source of concern for industries that provides commercial applications of digital images. It is easy to tamper with digital images and video than analog versions due to the powerful editing programs available in the market today. In the past few years, many new techniques, fragile watermarking, have been developed for tamper detection and image authentication of digital images and video. Fragile watermarking is designed to detect every possible change in pixel values. The main goal of such fragile watermarking techniques is to have good localization properties while resisting to a wide spectrum of attacks including vector quantization attack, random alterations, collage attack, rotation and scaling. In fragile watermarking, the goal of the attacker is not to make the authentication watermark unreadable instead try to make the changes to the image while preserving the watermark. It is the diversity of the scheme that can thwart such attempts and detect even the minutest changes noticed.

With the growing trend of storing the archives on digital format instead of paper, the importance of authentication at any later point of time is becoming more and more important and research in this field is gaining momentum

3.2 Review of Some Approaches

During the course of study for this paper, several research papers have been viewed. Some of the approaches, adopted by some of the people in this field, striving to achieve maximum security against image manipulation and tampering shall now be reviewed.

Wong [7] has proposed a method for image authentication and ownership verification and extended it to a more secure approach. The scheme detects any modification made to the image while indicating the specific locations that have been modified. It uses a key (secret key or private key) to embed a watermark that has been generated using 7 most significant bits (MSB) of the pixels in each block into the least significant bits (LSB). If the correct key is specified in the watermark extraction it shows the correct watermark indicating the image is authentic and image has not been changed. If the key is incorrect, the image was not watermarked, a new watermark is added or the image is cropped it extracts a random-noise like watermark. This effect is happening due to the property of the hash function used in the scheme. Also, the scheme is dependent on the key and thus it is impossible for an attacker to insert a new watermark or alter the existing watermark so that the resulting modified image will pass the authentication and ownership verification test. He uses a bitmap image as a watermark. The contribution of this bitmap is that if it has visual meaning then the authenticated image can be verified and tampered areas can be detected visually. Thus, the scheme has the satisfactory ability to localize modifications.

Shan Suthaharan and Seong-Whan Kim [16] present a gradient image dependent fragile watermarking scheme, which

thwarts VQ attack while providing superior localization properties. The proposed technique provides distinct input keys for each image block from a large ($1024!$ - 8770-bits entries) key space, a master key, a session key and block-wise permuted versions of a gradient image. The watermarking scheme requires only the master key (one time exchange) and the session key (each session) to be exchanged between the communication parties securely.

Phen-Lan Lin, Po-Whei Huang and An-Wei Peng [13] propose a fragile, block-wise, and content-based watermarking for image authentication and recovery. In this scheme, the watermark of each block is an encrypted form of its signature, which includes the block location, a content-feature of another block, and a CRC checksum. While the CRC checksum is to authenticating the signature, the mixture of the location indices of one block with the feature of a randomly selected block complicates the VQ attack. The encryption further strengthens the security. That all security parameters are user dependent and can be computed at both ends individually based on Diffie-Hellman key exchange method makes the scheme not only robust against collage attack but also truly oblivious. The experiments demonstrate that the scheme can detect and localize any tampering of size 8×8 pixels and above and can recover a 40% damaged image to an intelligible one with 24dB. As for incidentally manipulated images, the scheme can invalidate all the blocks but will not further degrade the images.

Anthony T.S. Ho in his paper, proposed a semi-fragile watermarking method for authentication of law enforcement images such as digital images captured at crime scenes and traffic enforcement situations, using the pinned sine transform (PST). The watermarking system can localize the portions of image that have

been tampered maliciously. The watermarking scheme is claimed to be very sensitive to any texture alteration in the watermarked images, which is crucial for crime scene image authentication.

Yazhou Liu , Wen Gao , Hongxun Yao , Shaohui Liu [15] proposed A texture-based tamper detection scheme by fragile watermarking technique. It is sensitive to malicious tamper such as replacing one's face in the image by another's and at the same time it's insensitive to other legal processing such as lossy JPEG compression and brightness/contrast changes.

Huiping Guo, Yingjiu Li, Anyi Liu and Sushil Jajodia In there paper, propose a fragile watermarking scheme to detect malicious modifications of database relations. In the proposed scheme, all tuples in a database relation are first securely divided into groups; watermarks are embedded and verified group by group independently. The embedded watermarks are claimed to not only detect but also localize, and even characterize, the modifications made to the database. In the worst case, the modifications can be narrowed down to tuples in a group.

Rafiullah Chamlawi, Asifullah Khan, Adnan Idris, and Zahid Munir in there paper, propose a secure semi-fragile watermarking, with a choice of two watermarks to be embedded. This technique operates in integer wavelet domain and makes use of semi fragile watermarks for achieving better robustness. A self-recovering algorithm is employed, that hides the image into some Wavelet sub bands to detect possible malevolent object manipulation undergone by the image (object replacing and/or deletion). The Semi-fragility makes the scheme tolerant for JPEG lossy compression and locate the tempered area accurately. In addition, the system ensures more

security because the embedded watermarks are protected with private keys. The computational complexity is reduced using parameterized integer wavelet transform.

F.-H. Yeh and G.C. Lee (Taiwan) [14] in their paper present a fragile watermarking approach that can localize tampered areas and resist the counterfeiting attack. Their algorithm eliminates communication security problem by using calculation signature function to create message signature for each block. Toral automorphism is applied for resisting counterfeiting attack. An image is divided into several non-overlapping blocks and signatures of each block are spread into other blocks that are chosen by toral automorphism with embedding keys. The effectiveness of our scheme is provided in experiments

Tsong-Yi Chen, Thou-Ho (Chao-Ho) Chen and Shang-Wei Lin In their paper, propose a new scheme that uses block-related index and three types of coefficient-scan strategy in image authenticating and recovering. Block related authenticating information makes the feature codes tolerant in compression processes. Besides, a block-classification method is used for classifying blocks into flat, vertical-detailed or horizontal-detailed types. With that, a useful coefficient-scan strategy is chosen for preserving and recovering block information.

3.3 The Proposed Scheme's Approach

Keeping the above discussion in view, we can now clearly define the approach parameters of the scheme being discussed in this paper may be enumerated below:

- a. The scheme uses a fragile watermark approach as compared to the robust one as it is mainly designed

for authentication purposes. Any modification / manipulation of the fragile watermark is helpful in the assessment and validation of an image with a to ascertain its tampering or otherwise

- b. The proposed watermarking scheme does not resort to the original signal in the watermark extraction process, it is blind watermarking in this respect. In terms of capacity or security, blindness causes no ill effect on performance. But it does feature loss of robustness[5]. Informed schemes are impractical when the availability of original signal cannot be warranted. For example, in authentication applications, obtaining original signal is unrealistic.
- c. Since only authorized users, for instance the holder of secret key, can detect or extract the watermark, this scheme has a private approach. Usually public watermarking schemes are computationally more expensive.
- d. The scheme's fragile watermark is based on the watermarks of individual sub block contents. This gives the scheme's watermark some diversity. Moreover, since each block has its own watermark, any tampering or modification can be detected down to the block level which in our case is 8x8 pixels.
- e. Compared to some of the other approaches discussed above, the scheme is considered to be more secure because of the method of embedding of any remainder of long division into the binary signature string (described in detail in chapter 4). The security is because of the fact that a secret generator polynomial of up to the length

of block signature may be used. This ensures minimum loss of information thereby giving good PSNR values of the watermarked image.

- f. The proposed scheme is also considered to be simple because of the fact that security is achieved by increasing the length of the secret generator polynomial. In comparison some of the other schemes rely on cryptology to enhance security, thereby increasing the complexity of code and processing at both the sender and the receiver ends.

Design and Implementation of Proposed Content based Fragile Watermarking Scheme

4.1 Introduction

This chapter shall give an introduction of the structure and dynamics of the content based fragile watermarking system developed with a view to provide a secure and tamper-detecting watermarking system. With the availability of some powerful image processing soft wares such as 'Adobe Photoshop', one can remove/replace some features in a picture easily without any detectable trace. This type of operation may be regarded as tamper. However, in some cases, we cannot afford an image to undergo any such operation, such as images for military, medical or judicative use etc. The validity and authenticity of an image is of utmost importance in such cases, so we need to guarantee the integrity of an image in an effective manner. Also in the second phase, we must be able to localize and identify an area which has been tampered or has developed some sort of an error while passing the transmission channel or during storage/retrieval process.

4.2 Design, Implementation and Functional Details

In this section, we shall be discussing the proposed scheme's design, implementation and functional details. The scheme mainly consists of two module namely:

- a. Sender End Module
- b. User End Module

First we give the block diagrams of both these modules. This shall be followed by the various algorithms and their respective flowcharts. Finally, the detailed functional description of the proposed scheme shall be given.

4.2.1 Module Block diagrams

The basic block diagrams of the sender and the receiver end modules are given in figures 4.1 and 4.2 respectively. The description shall be given separately for the embedding (sender) and the extraction / authentication (receiver) ends.

4.2.1.1 Embedding Block Diagram Description

The block diagram of embedding given in figure 4.1 is described as follows:

- a. Image is selected as per the path entered by the user.
- b. The user selects a security level from the available choice of two security levels 1 and 2, depending upon the security requirements.
- c. Image is divided into sub blocks of size 8x8 from left to right and top to bottom
- d. One sub block at a time is taken, starting from first block on the top left and proceeding from left to right and top to bottom till all the sub blocks are processed
- e. Each sub block is further divided into 2x2 (security level 1) or 4x4 (security level 2) sub sub blocks and average of pixel intensities of these 2x2 or 4x4 blocks is calculated.
- f. The average intensities are appended in front of each other from left to right to obtain a string of 128 bit

(security level 1) or 32 bit (security level 2), known as a sub block's signature

- g. Each block signature is divided by a secret generator polynomial decided upon by the sender and the receiver. Any remainder obtained as a result of this division is embedded into the LSBs of the signature string, e.g a 3 bit remainder is embedded into three LSBs of the signature string and so on. This makes the string perfectly divisible by the secret generator polynomial
- h. The string formed after the above step is the watermark of the 8x8 sub block and is embedded into the two LSBs of each pixel of the block starting from left to right and top to bottom.
- i. The image is watermarked after all the sub blocks of the image are processed in this manner. The watermark image is thus stored at the path specified by the user.

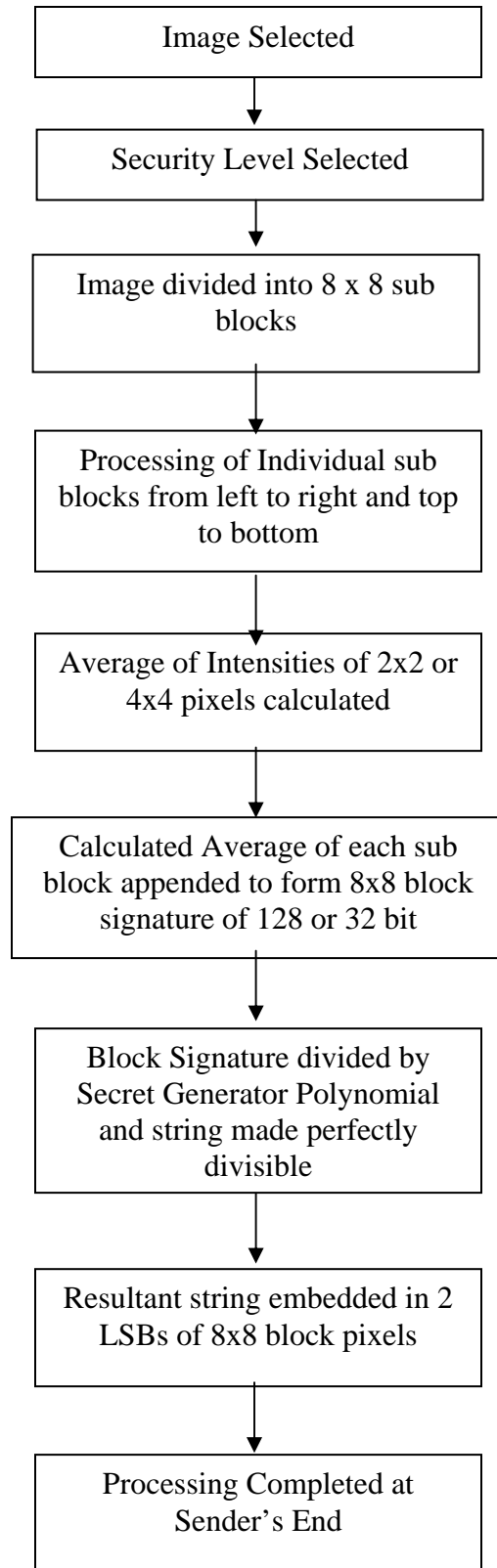


Figure 4.1 Embedding Block Diagram

4.2.1.2 Extraction / Authentication Block Diagram

Description

The block diagram of extraction, given in figure 4.2 is described as follows:

- a. The watermark image retrieved from the path specified by the user
- b. Image is divided into sub blocks of size 8x8 from left to right and top to bottom
- c. One sub block at a time is taken, starting from first block on the top left and proceeding from left to right and top to bottom till all the sub blocks are processed
- d. Each sub block's watermark is formed by extracting the values of two LSBs of each pixel from left to right and top to bottom.
- e. The extracted watermark is divided by the same secret generator polynomial decided upon by the sender and the receiver.
- f. If a remainder is obtained as a result of the above step, the block is modified or tampered and is marked so. If no remainder is obtained, the block is authentic.
- g. Entire image is processed in a similar manner.

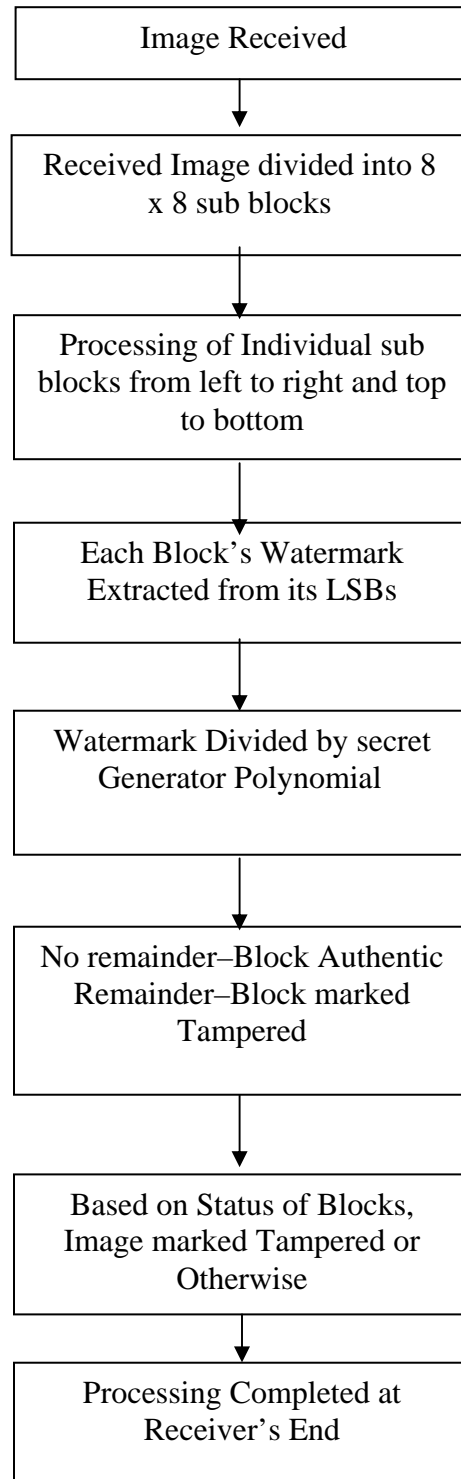


Figure 4.2 Extraction / Authentication Block Diagram

4.2.2 Proposed Schemes Algorithms and Flowcharts

This section gives in detail the various scheme algorithms and their respective block diagrams

4.2.2.1 Sender End Algorithm The sender end flowchart is given in Figure 4.3 below followed by the algorithm. The shaded processes are explained in separate flow charts:

Algorithm 1 – Reference flow chart Figure 4.3

- initialize all variables to zero
- acquire the image as per the path entered by the user
- calculate image size (rows and columns)
- image Segmentation
 - if image row \leq total image rows
 - if image columns less than or equal to total image columns
 - process image segment of size 8x8 pixels
 - if security level selected = 1
 - call function binary_signature 2x2 for calculating 128-bit binary signature
 - else if security level selected = 2
 - call function binary_signature 4x4 for calculating 32 bit binary signature
 - reset 2 LSBs of each pixel of image segment to zero
 - convert the binary signature string into number string
 - perform long binary division and note remainder
 - pad remainder on left with zeros
 - embed Remainder using XOR operation
 - embed Watermark for security level 1 or 2
 - increment Columns
 - increment Rows
- store watermarked image

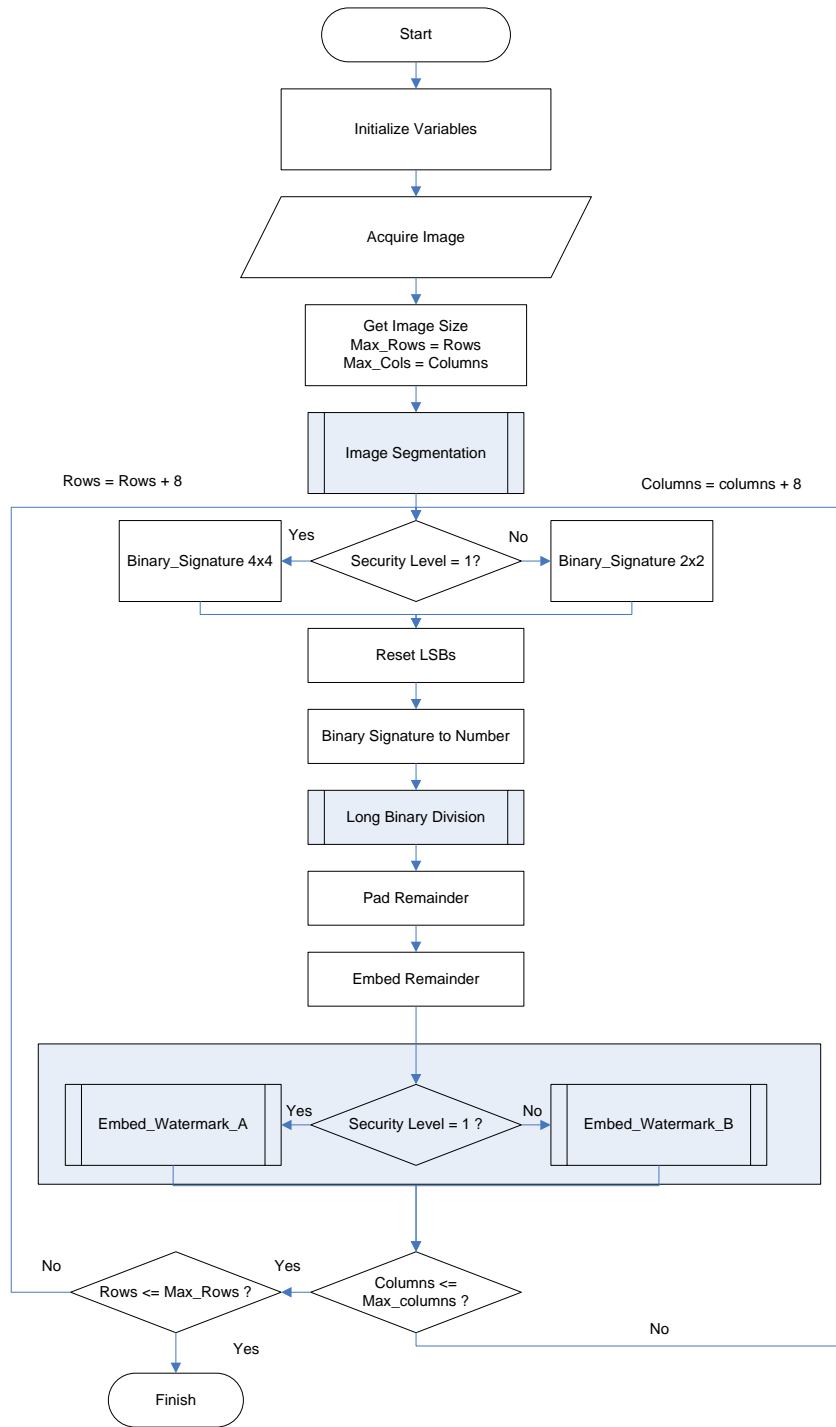


Figure 4.3 Sender End Flow Chart

4.2.2.2 RECEIVER END ALGORITHM

The receiver end flowchart is given in Figure 4.4 below followed by the algorithm. The shaded processes are explained in separate flow charts:

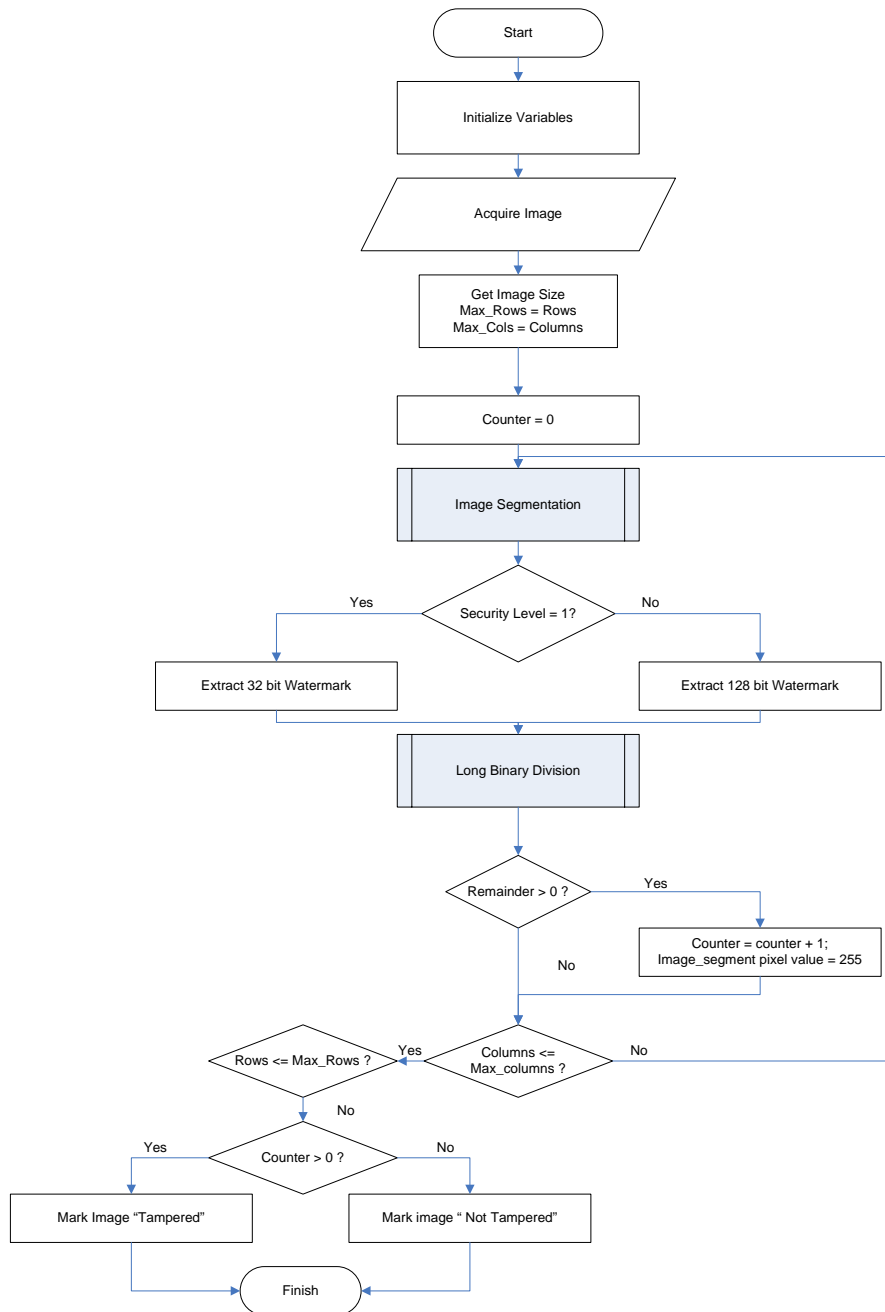


Figure 4.4 Receiver End Flowchart

Algorithm 2 – Reference flow chart Figure 4.4

- initialize all variables to zero
- acquire the image as per the path entered by the user
- calculate image size (rows and columns)
- counter = 0
 - if image row \leq total image rows
 - if image columns less than or equal to total image columns
 - process image segment of size 8x8 pixels
 - if security level selected = 1
 - extract 128 bit watermark
 - else if security level selected = 2
 - extract 32 bit watermark
 - perform long binary division
 - if remainder = 0
 - increment columns
 - increment rows
 - else if remainder > 0
 - counter = counter + 1
 - value of image segment block pixels = 255
 - increment columns
 - increment rows
 - if counter > 0
 - mark image “Tampered”
 - else if counter = 0
 - mark image “Not Tampered”

4.2.2.3 EMBED WATERMARK A & B

The flowchart of Figure 4.5 and subsequent algorithm explain the embedding process:

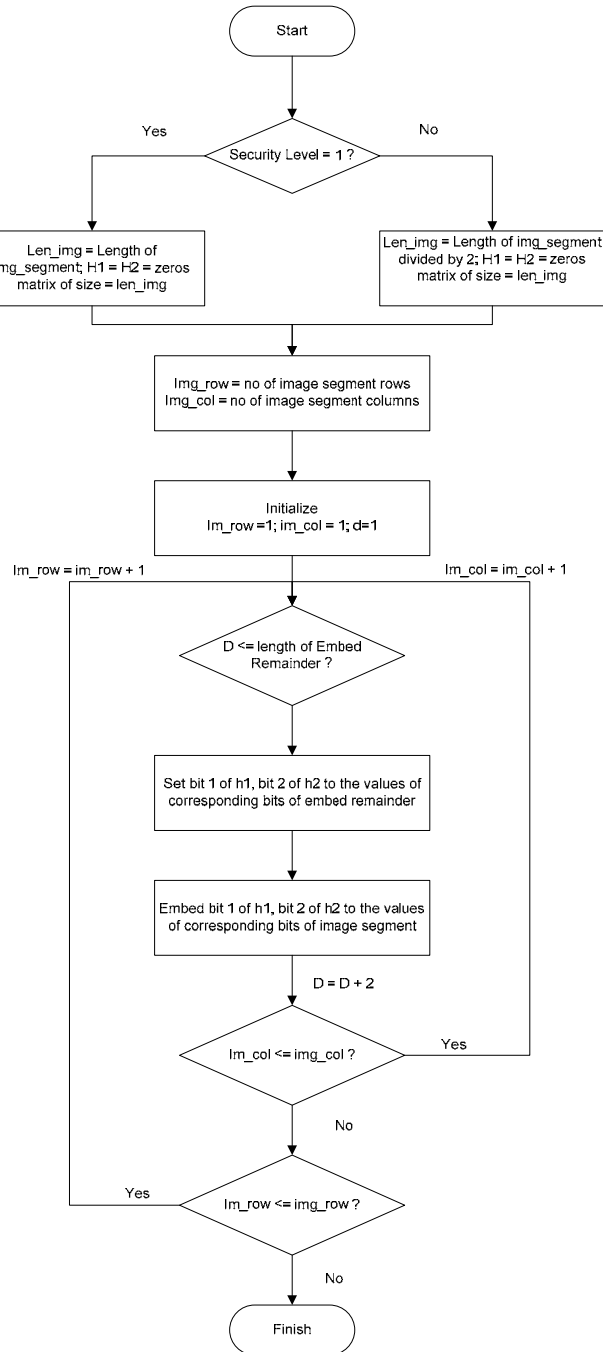


Figure 4.5 Watermark Embedding Flowchart

Algorithm 3 – Reference flow chart Figure 4.5

- initialize
- if security level = 1
 - len_img = length of image segment
 - h1 = h2 = zeros matrix of size = len_img
- else if security level = 2
 - (len_img = length of image segment) / 2
 - h1 = h2 = zeros matrix of size = len_img
- initialize variables
 - img_row = number of image segment rows
 - img_col = number of image segment columns
 - im_row = im_col = 1; d = 1
- if d <= length of embed remainder
 - set bit 1 of h1, bit 2 of h2 to corresponding values of embed remainder
 - embed h1, h2 in bit 1 and 2 positions of image segment
- if im_col <= img_col
 - increment img_col by 1
- else if im_col > img_col
 - increment im_row by 1

4.2.2.4 Image Segmentation

The flowchart of Figure 4.6 and subsequent algorithm explain the embedding process:

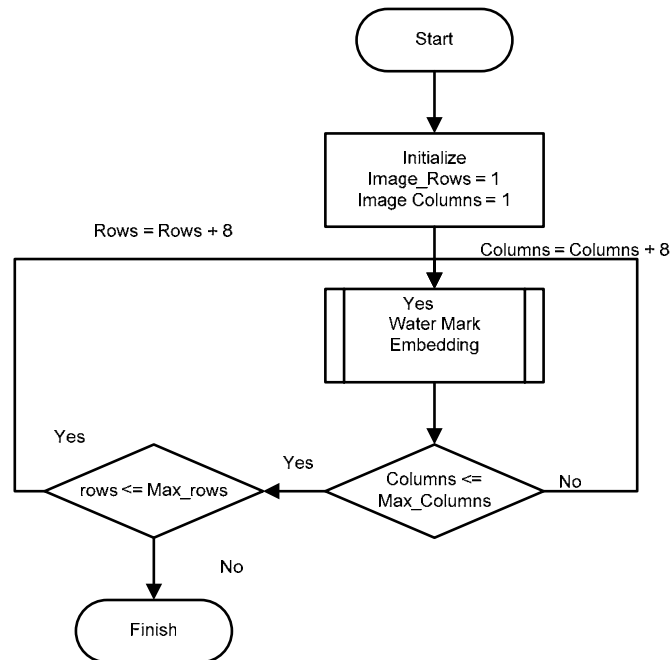


Figure 4.6 Image Segmentation Flowchart

Algorithm 4 – Reference flow chart Figure 4.6

- initialize variables
 - image_rows = 1
 - image_columns = 1
- embed_watermark A&B
- if columns <= max_columns
 - if rows <= max_rows
 - rows = rows + 8; loop
 - if rows > max_rows
 - finish
- else if columns > max_columns
 - finish

4.2.2.5 LONG DIVISION

The flowchart of Figure 4.7 and subsequent algorithm explain the long division process:

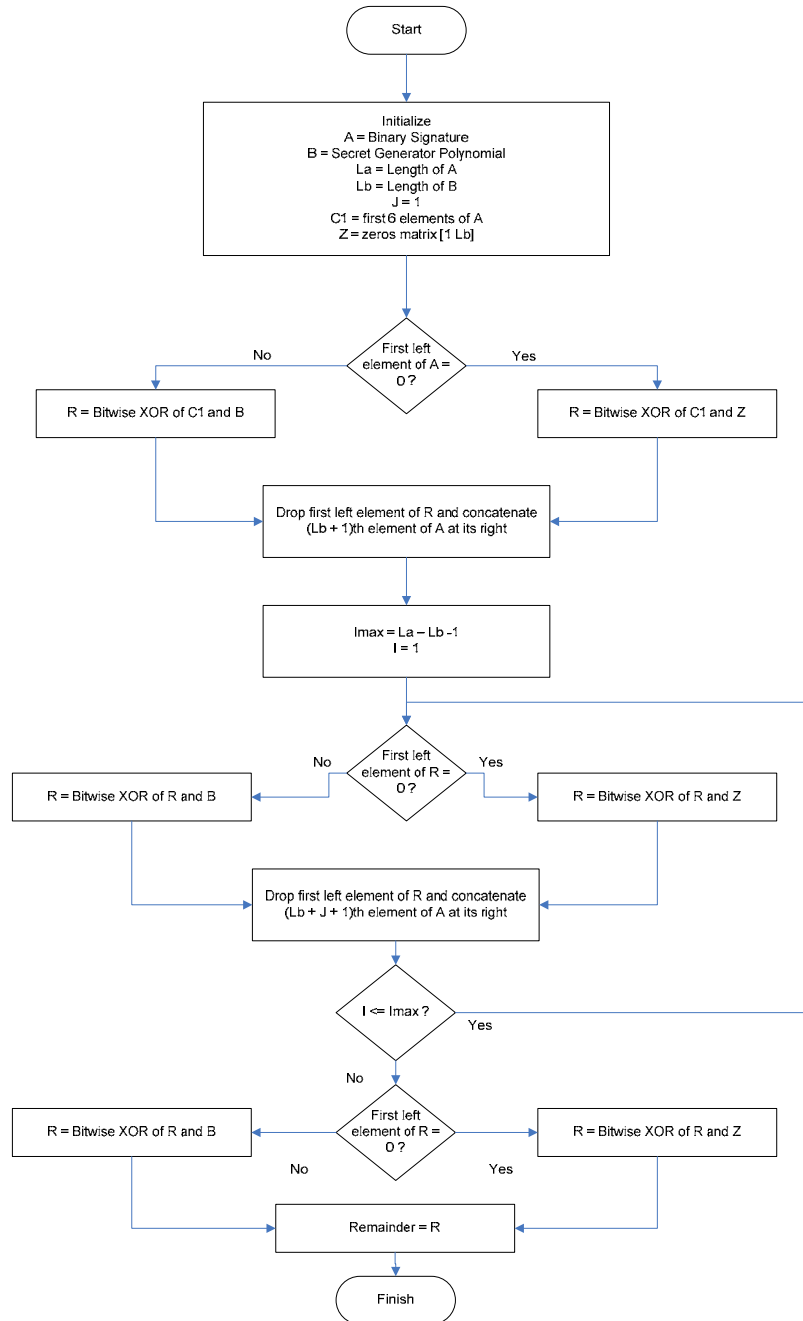


Figure 4.7 Long Division

Algorithm 5 – Reference flow chart Figure 4.7

- initialize variables
 - A = binary signature of segment
 - B = secret generator polynomial
 - La = length of A
 - Lb = length of B
 - J = 1
 - C1 = first 6 elements of A
 - Z = zeros matrix of size = Lb
- if first element of A = 0
 - R = bitwise XOR of C1 and Z
- else if first element of A not 0
 - R = bitwise XOR of C1 and B
- drop first left element of R and concatenate (Lb + 1)th element of A to its right
- if $l \leq l_{max}$
 - loop
- else if $l > l_{max}$
 - if first element of R = 0
 - R = bitwise XOR of R and Z
 - else if first element of R not 0
 - R = bitwise XOR of R and B
- remainder = R
- finish

4.2.3 Proposed Scheme's Detailed Functional Description

The proposed algorithm implements the idea of authentication and error location of an image, using fragile content based watermark. The redundancy check performs a cyclic redundancy check which not only authentications but localizes the tampered/erroneous block as well.

4.2.4.1 Image Segmentation

The image to be watermarked and transported is divided into non overlapping blocks of 8x8 pixels each. Each block has its own watermark, generated on the basis of its contents (described below). The content based watermark of each block is embedded into two LSBs of each pixel

4.2.4.2 Security Level

The security level may be selected by the user depending on the amount of security vis a vis the quality of watermarked image required. Level 1 is more secure level as the length of the content to be embedded is 128 bits. However, since all the 64 (8 x 8) LSBs are used for embedding of the content, the quality of the watermarked image is slightly reduced. On the other hand level 2 is slightly lesser secure because of the content length of 32 bit , but it also helps produce a better quality watermarked image.

4.2.4.3 Content based watermark.

The watermark for each block described in the step above in generated using the contents of the 8x8 pixels of the blocks. The contents used in our case are the intensity values of each pixel.

The intensity value of each pixel is represented in the 8-bit binary value. Each 8x8 blocks is further divided into 2x2 sub blocks in security level 1 and 4 x 4 sub blocks in security level 2. Thus average intensity a 2x2 or 4x4 sub block is calculated. For an 8x8 block we get 16 values of average intensities for security level 1 and 4 values for security level 2. We arrange these 16 or 4 valves into a bit stream of 128 bits (16x 8 = 128 bits) or 32 bits(4 x 8 = 32) respectively. The concept of division, sub division and calculation of 128 and 32 bit streams becomes clearer when we refer to Figure 4.7 and Figure 4.8.

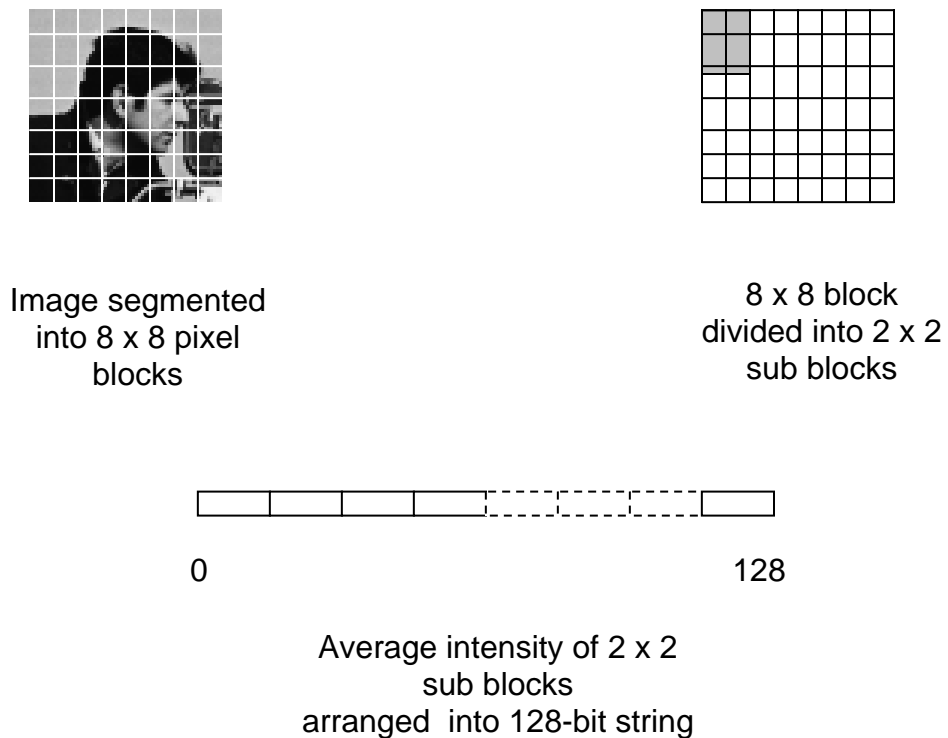


Figure 4.8 128 bit string formation

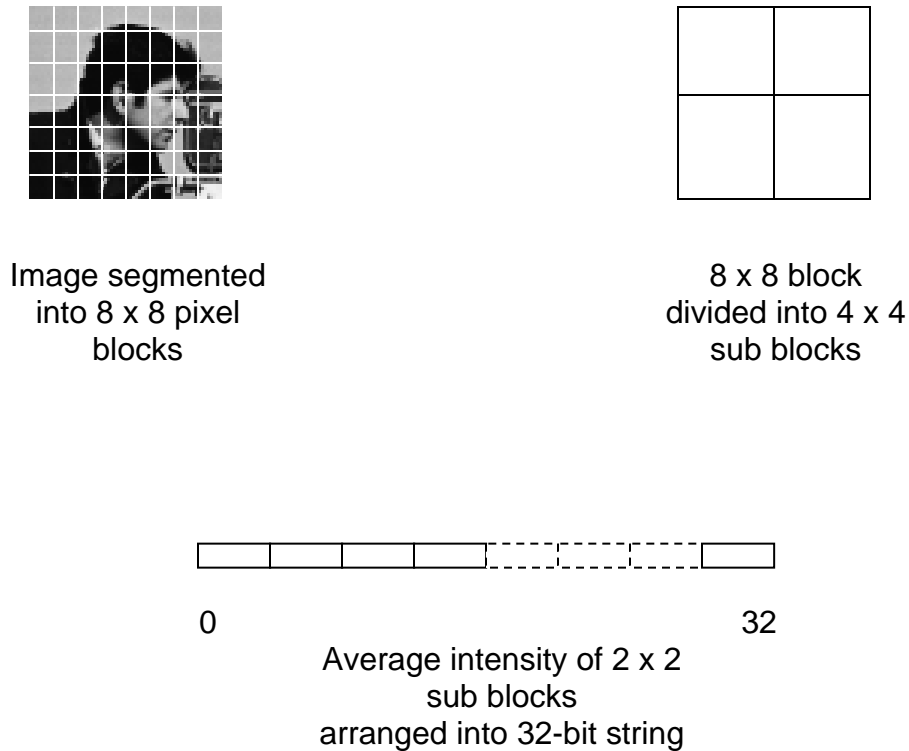


Figure 4.9 32 bit string formation

4.2.4.4 Generator matrix and Polynomial Arithmetic

Polynomial arithmetic based on Exclusive OR operations is used in the steps to follow. The 128 or 32 bit string obtained in the above step is treated as a polynomial, e.g string '10110' may be represented in polynomial form as X^4+X^2+X . A generator polynomial (referred to as $G(x)$ from now onwards) is decided upon by the sender and the receiver. This is the only thing that the receiver requires to authenticate the image. The $G(x)$ decided upon is used to divide the 128 or 32 bit string obtained from the previous step, using the long division method, at the sender's end. If the string is not fully divisible by $G(x)$ then we obtain a remainder $r(x)$. This remainder, when added to LSBs of 128 or 32 bit string produces another 128 or 32 bit string which is now completely

divisible by $G(x)$, leaving no remainder. This fully divisible string is the content based watermark of this 8x8 block and is embedded in the two LSBs of each pixel from left to right and top to bottom.

In this way, all 8x8 blocks of the image are scanned and embedded with their respective content based, fully divisible watermarks.

4.2.4.5 Authentication and Error Location

The watermarked image, produced in the previous step is authenticated at the receiver's end and any tampering or errors detected need to be localized.

4.2.4.6 Image segmentation

Image is again segmented into 8x8 pixel, non overlapping blocks, as was done previously.

4.2.4.7 Watermark Extraction

The content based watermark of an 8x8 pixel block is extracted by extracting the two LSBs of each of the 8x8 pixels from left to right and top to bottom. This produces a 128 bit string of watermark. The string thus obtained is divided by $G(x)$ which has already been decided upon by the sender and the receiver. If the long division by $G(x)$ produces no remainder then the block has neither been tampered, nor any error has been introduced in it and it may be termed as authentic. However, if a remainder is obtained as a result of long division by $G(x)$ then the block is termed as tampered. The image is scanned in this manner from left to right and top to bottom to identify any corrupted or tampered blocks.

Evaluation Results and Discussion

This chapter gives details of various test and trial conducted to ascertain the efficacy of the system developed. The system has been thoroughly checked in several aspects, the details of which shall be given as the chapter progresses.

5.1 Technical Parameters

Given below are the technical details of the system on which the system has undergone test and trials:

- a. Computer Type – Desktop
- b. Processor – P IV Intel
- c. RAM – 256 MB
- d. Programming Language – MATLAB 7

These technical details have been given so that it may be appreciated that the various response times of the software (given in succeeding paras) may vary as and when the system specs given above are varied.

5.2 Testing Methodology

The tests conducted and various responses have been noted by taking three readings of results and then average of these results have been documented in this chapter.

5.3 Image Quality Computation

Signal-to-noise (SNR) measures are estimates of the quality of a reconstructed image compared with an original image. The basic idea is to compute a single number that reflects the quality of the reconstructed image. Reconstructed images with higher metrics are judged better. In fact, traditional SNR measures do not equate with human subjective perception. Several research groups are working on perceptual measures, but for now we will use the signal-to-noise measures because they are easier to compute. Just remember that higher measures do not always mean better quality.

The actual metric we will compute is the peak signal-to-reconstructed image measure which is called PSNR. Assume we are given a source image $f(i,j)$ that contains N by N pixels and a reconstructed image $F(i,j)$ where F is reconstructed by decoding the encoded version of $f(i,j)$. Error metrics are computed on the luminance signal only so the pixel values $f(i,j)$ range between black (0) and white (255).

PSNR in decibels (dB) is computed by using

$$\text{PSNR} = 20 \log_{10} \left(\frac{255}{\text{RMSE}} \right)$$

The actual value is not meaningful, but the comparison between two values for different reconstructed images gives one measure of quality.

5.4 Images and their histograms

In this section we give various images used in our test and trials along with their respective histograms. The histograms would help us understand the response given by different images after they are watermarked by our system.

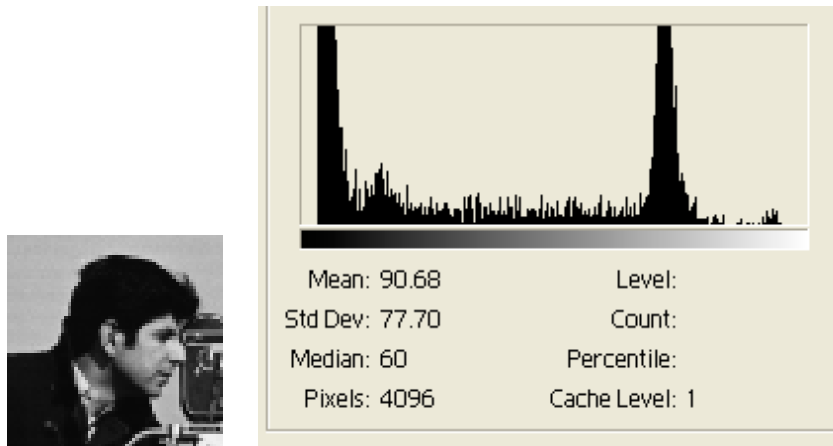


Figure 5.1 Cameraman and its Histogram

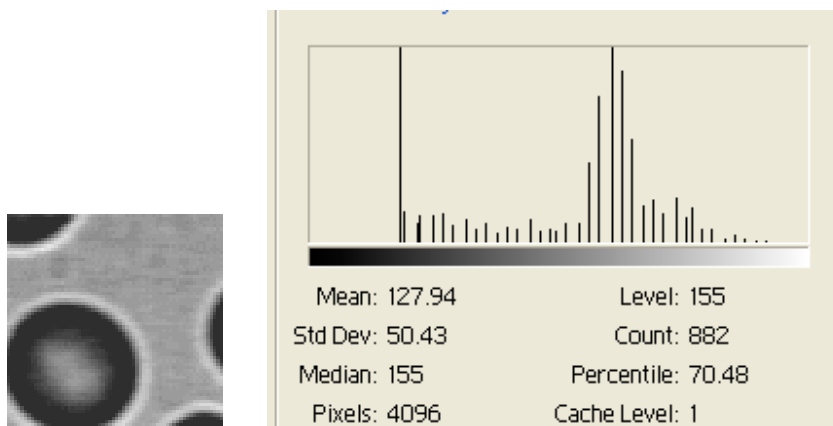


Figure 5.2 Cell and its Histogram

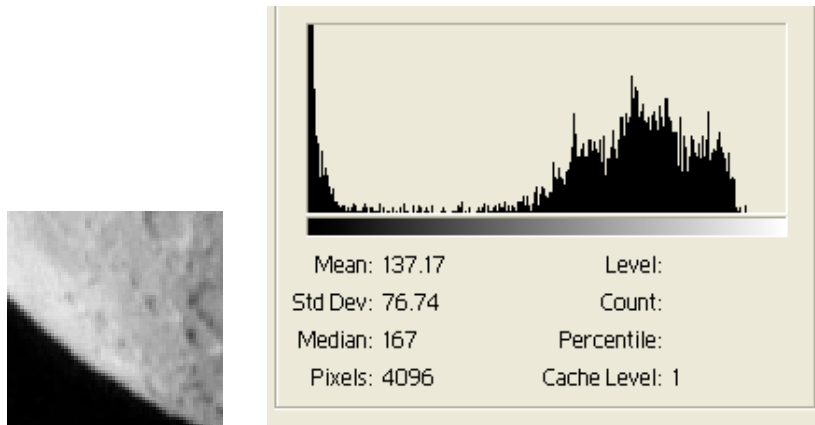


Figure 5.3 Moon and its Histogram

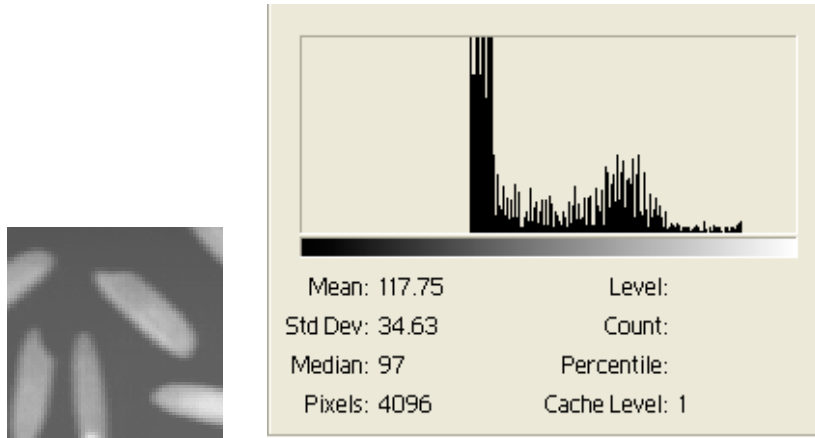


Figure 5.4 Rice and its Histogram

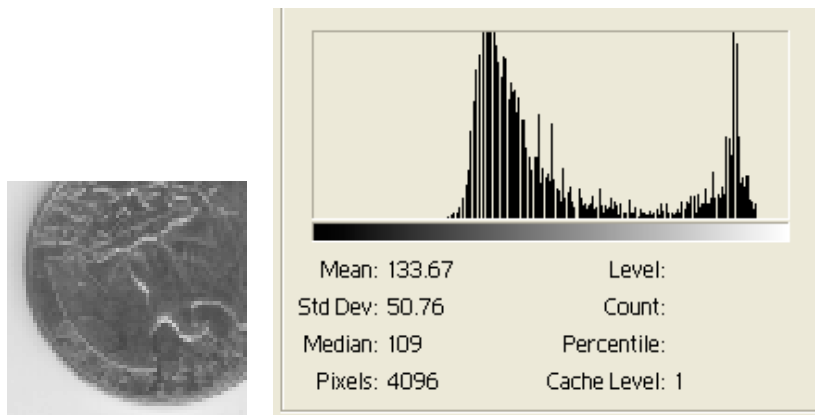


Figure 5.5 Coin and its Histogram

5.5 Detailed Evaluation and Discussion

In this section we shall be giving the various readings and responses of the software under different headings:

1. Image PSNR versus Security Level

S.no	Image	Security Level 1	Security Level 2
1.	Cameraman	46.69 db	50.61 db
2.	Cell	46.67 db	50.08 db
3.	Moon	47.31 db	51.13 db
4.	Rice	49.23 db	51.84 db
5.	Coin	48.11 db	49.69 db

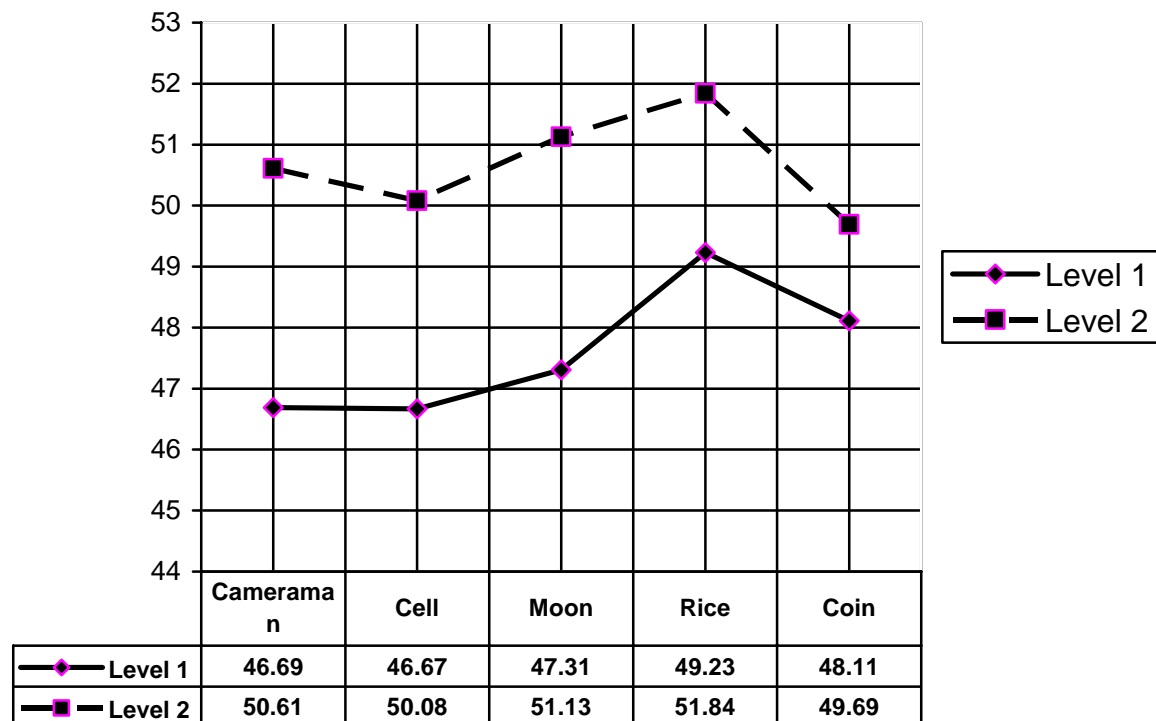


Figure 5.6 Graph – Image PSNR versus Security Level

The PSNR values of all the images have improved in security level 2 as compared to security level 1. The improvement of PSNR can be seen to be from 4% to 9%. This is because of the fact that the content based watermark in security level 1 is of 128 bit length thus making it more secure but at the cost of some of the image quality. On the other hand the watermark in security level 2 is of 32 bit length preserving image quality but at the cost of security.

2. Comparison of Proposed Schemes PSNR

The graph in figure 5.7 below provides a comparison of the proposed scheme with some of the other available schemes. The graph shows a very satisfactory performance by the proposed scheme

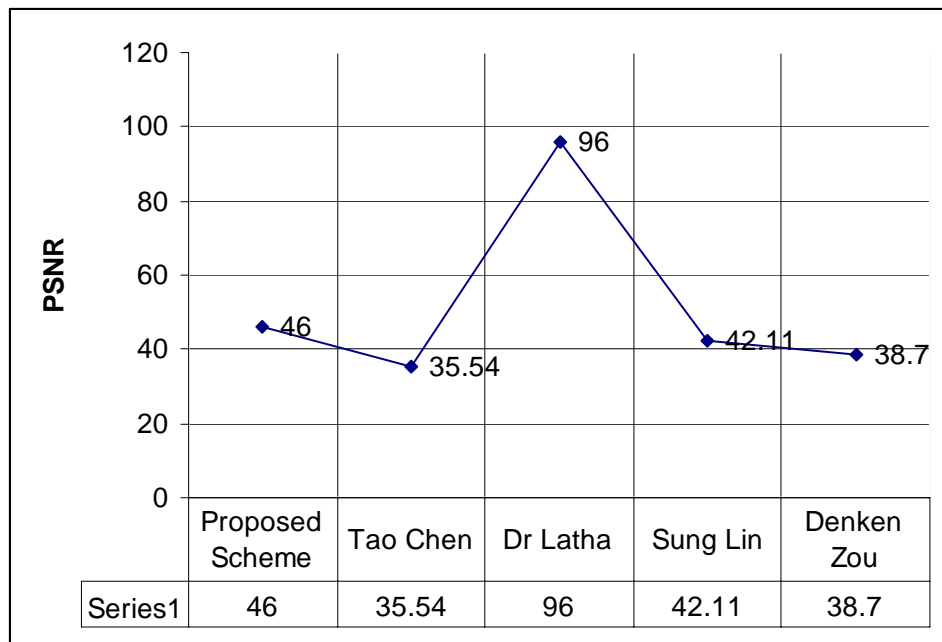


Figure 5.7 Graph – Comparison of Proposed Scheme

3. Watermarking Time(secs) versus Security Level

S.no	Image	Security Level 1	Security Level 2
1.	Cameraman	6.57	3.45
2.	Cell	6.62	3.39
3.	Moon	6.85	3.37
4.	Rice	6.62	3.62
5.	Coin	6.86	3.57

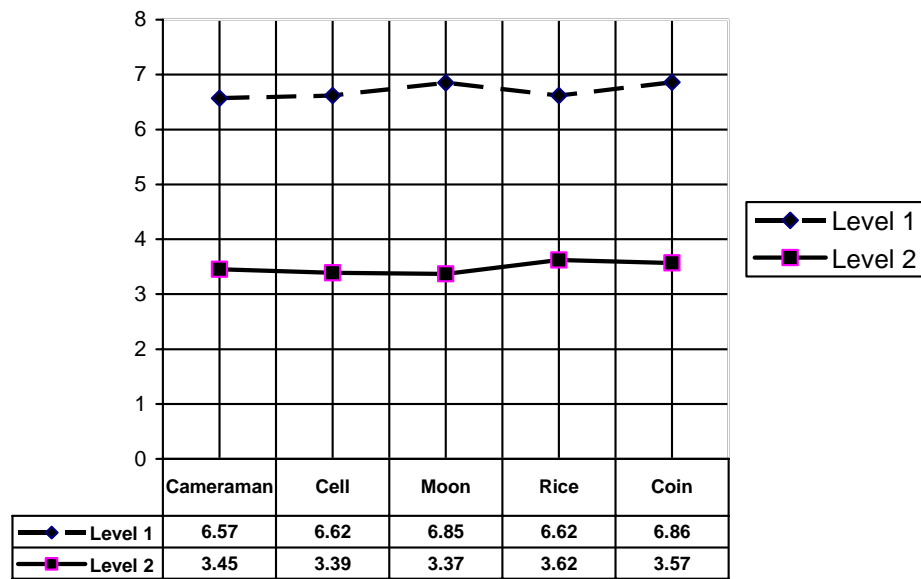


Figure 5.8 Watermarking Time(secs) versus Security Level

The reduction in watermark embedding time from security level 1 to security level 2 by almost 50% is approximately the same for all the images. This is because of the fact that in security level 1 the watermark string to be embedded is 128 bit in length as opposed to the length of 32 bit string of security level 2 which obviously takes lesser time to embed.

4. Watermarking Time(secs) versus Image Size

S.no	Image Size	Security Level 1	Security Level 2
1.	64 x 64	5.93	3.5
2.	128 x 128	21.87	10.6
3.	256 x 256	84.96	39.84

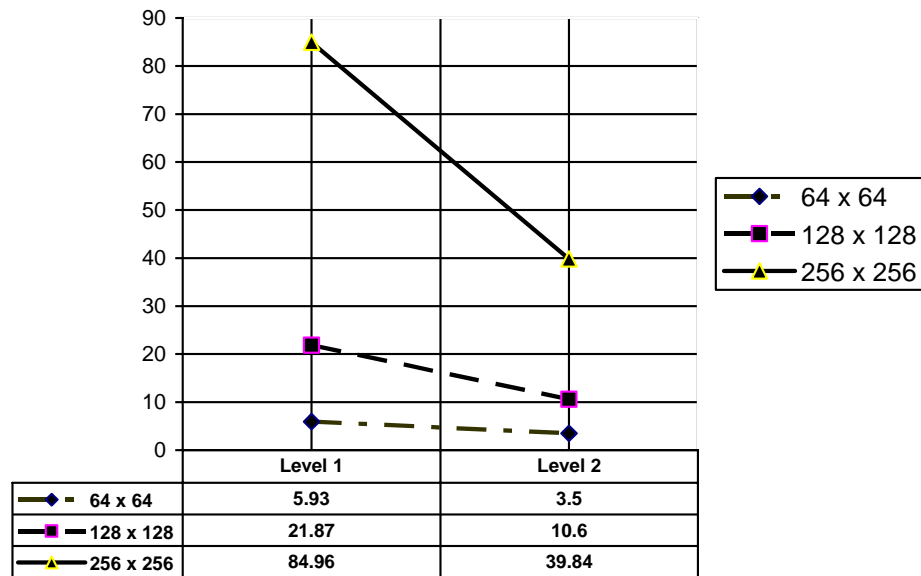


Figure 5.9 Watermarking Time(secs) versus Image Size

The change in the watermark embedding time is more or less proportional to the image size. As the image size increases so does the processing time. However, the processing times for each security level changes in the same manner as has already been described above. This almost proportional increase or decrease of embedding time with image size can help extrapolate embedding times for other image sizes as well.

5. PSNR versus Image Size

S.no	Image Size	Security Level 1	Security Level 2
1.	64 x 64	47.51 db	50.95 db
2.	128 x 128	47.61 db	50.55 db
3.	256 x 256	47.61 db	50.55 db

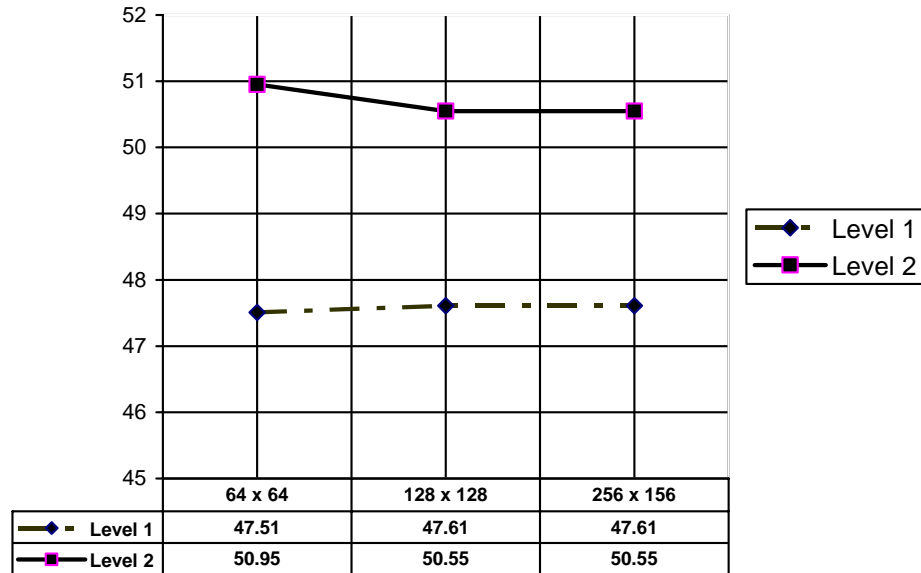


Figure 5.10 PSNR versus Image Size

We can clearly see from above that irrespective of image size, the system produces watermarked images of same PSNR value quality which is a positive aspect. This also implies that the scheme can effectively be used for any image size provided time is not a constraining factor. The improvement in PSNR value when using security level 2 is almost 7%.

6. Watermarking Time(secs) versus Generator Polynomial Length

S.no	Gen Poly Length	Security Level 1	Security Level 2
1.	8	5.90	3.14
2.	16	5.88	3.12
3.	32	5.90	3.11
4.	64	6.17	

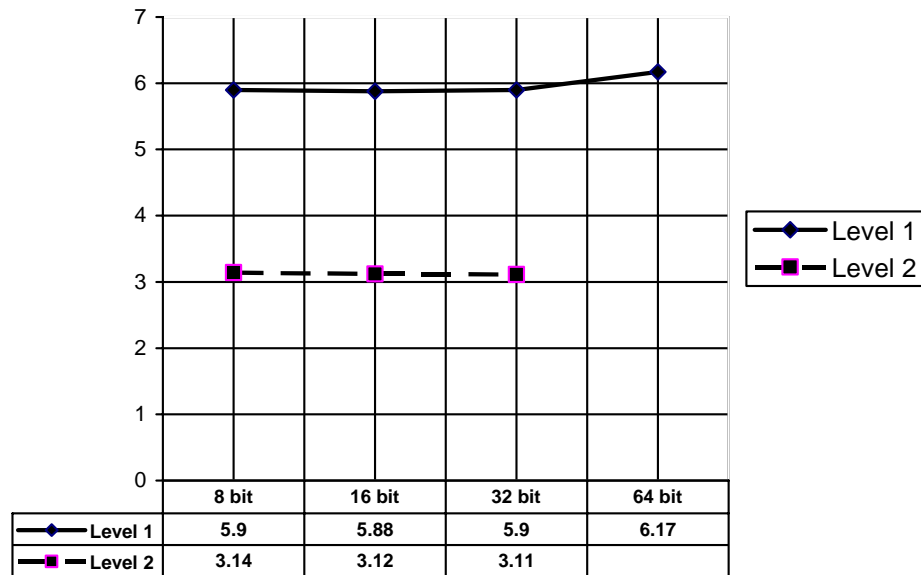


Figure 5.11 Watermarking Time(secs) versus Generator Polynomial Length

The above table and graph shows that the length (in bits) of the secret generator polynomial has no bearing on the watermark embedding time. We know that secret generator polynomial is the hub of system security and if its length has no bearing on the timing then we can use larger length polynomial making the embedded image more secure without any additional delay. This is one of the most important and useful aspects of the proposed schemes because by virtue of this very characteristic, we do not require any additional encryption as the system is secure to 2^n th degree of

secret generator polynomial selected. Security level 2 is processing wise almost 85% faster.

7. PSNR versus Generator Polynomial Length

S.no	Gen Poly Length	Security Level 1	Security Level 2
1.	8	47.59 db	50.99 db
2.	16	47.49 db	51.01 db
3.	32	47.31 db	50.75 db
4.	64	47.16 db	

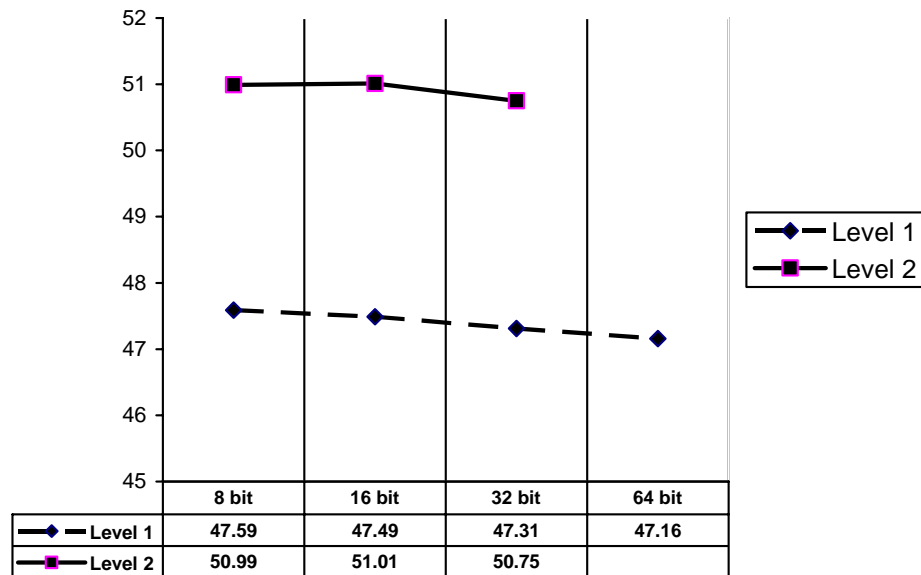


Figure 5.12 PSNR versus Generator Polynomial Length

Like watermark embedding time, the length of secret generator polynomial has no bearing on the PSNR and quality of the watermarked image as well. This is again a very positive point because a good PSNR of the watermarked image implies transparency which is one of the most desirable features of the

watermark as discussed in the early chapters in detail. The improvement of PSNR from security level 1 to 2 is almost 7% which was the case previously as well so we can use a larger degree of secret generator polynomial, giving us more security but no loss of image quality.

8. Watermarking Time(secs) versus Different Image Formats

S.no	Image Size	Format	Security Level 1	Security Level 2
1.	64 x 64	TIF	6.51	3.23
2.	64 x 64	JPEG	6.34	3.44
3.	64 x 64	BMP	6.38	3.22
4.	128 x 128	TIF	23.25	10.79
5.	128 x 128	JPEG	23.18	10.81
6.	128 x 128	BMP	23.09	11.00

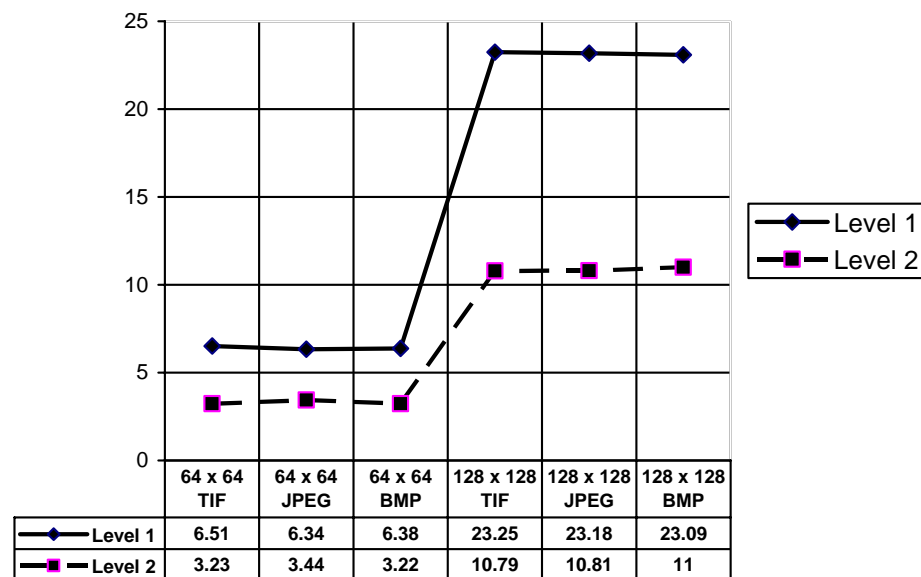


Figure 5.13 Watermarking Time(secs) versus Different Image Formats

The system treats various image formats in a similar manner as is clearly evident from above readings and graph. This also proves the diversity of the proposed scheme in watermarking various image formats uniformly.

9. PSNR versus Different Image Formats

S.no	Image Size	Format	Security Level 1	Security Level 2
1.	64 x 64	TIF	47.51 db	50.95 db
2.	64 x 64	JPEG	47.22 db	50.61 db
3.	64 x 64	BMP	47.53 db	50.8 db
4.	128 x 128	TIF	47.61 db	50.55 db
5.	128 x 128	JPEG	47.43 db	50.44 db
6.	128 x 128	BMP	46.92 db	49.94 db

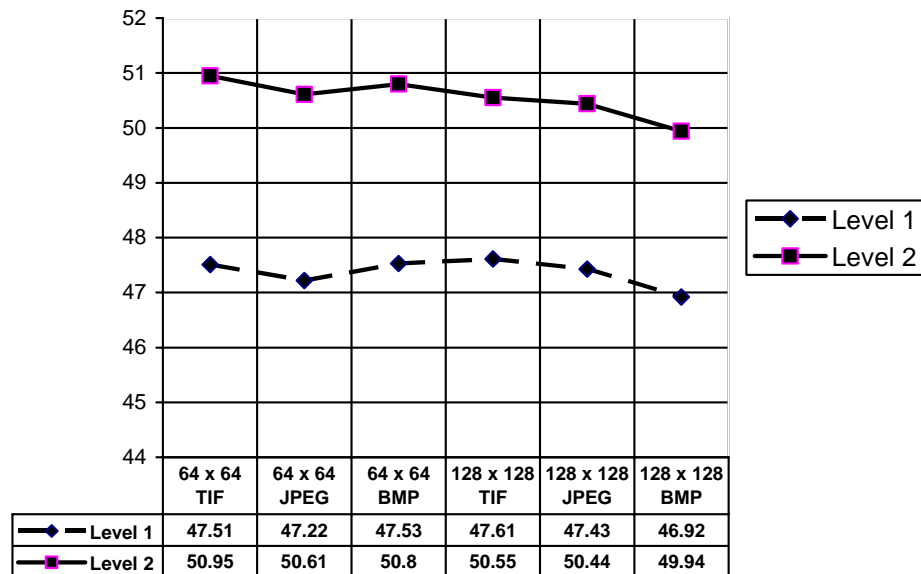


Figure 5.14 PSNR versus Different Formats Image

The system's strong uniform response in terms of PSNR and quality of watermarked image is evident from the above readings and graph.

5.6 Transmitted and Watermarked Images

In this section we give original and watermarked images of various image size and formats whose PSNR values have already been given in the table above. The images make it evident that the visual quality of the watermarked images is good thus fulfilling the requirement of transparency of watermark in a befitting manner.

a. TIF Images



Original
64x64 Image



Watermarked Image
PSNR 47.51



Original
128x128 Image



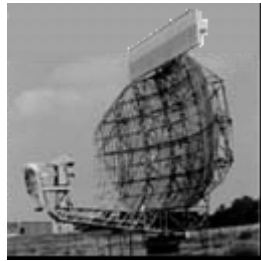
Watermarked Image
PSNR 47.61



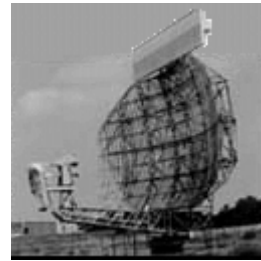
Original
64x64 Image



Watermarked Image
PSNR 47.53



Original
128x128 Image



Watermarked Image
PSNR 46.92

c. JPEG Images



Original
64x64 Image



Watermarked Image
PSNR 47.22



Original
128x128 Image



Watermarked Image
PSNR 47.43

5.7 Tamper Detection and Error Localization Capabilities

This section gives the results in respect of a very important aspect of this system i.e. tamper detection and error localization. The pictures given below say it all:

5.7.1 Camerman



Original Image



Watermarked Image PSNR 46.69

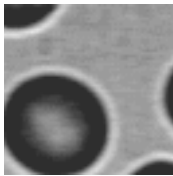


Tampered Image

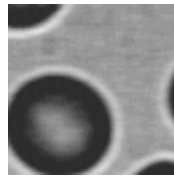


Tampered Areas Identified

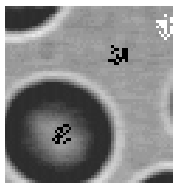
5.7.2 Cell



Original Image



Watermarked Image PSNR 46.67

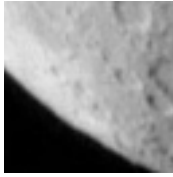


Tampered Image



Tampered Areas Identified

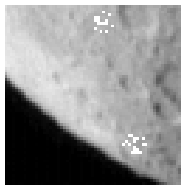
5.7.3 Moon



Original Image



Watermarked Image PSNR 47.31



Tampered Image



Tampered Areas Identified

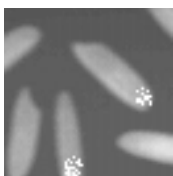
5.7.4 Rice



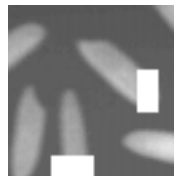
Original Image



Watermarked Image PSNR 49.23



Tampered Image



Tampered Areas Identified

5.7.5 Coin



Original Image



Watermarked Image PSNR 48.11



Tampered Image



Tampered Areas Identified

All the pictures shown above portray the proposed scheme's capabilities of precisely localizing and highlighting the tampered or modified areas of the transmitted watermarked images. The watermarked images have been intentionally tampered. The utility and importance of the proposed scheme becomes even more clear when we have a look at some of the screen shots of the proposed scheme where some the tampered areas of some images of medical and legal fields have been accurately localized and highlighted.



Original Image

Received Image



Tampered Areas Identified





Original Image

Received Image



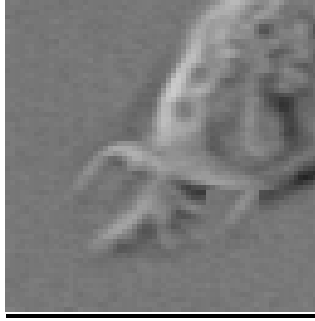
Tampered Areas Identified





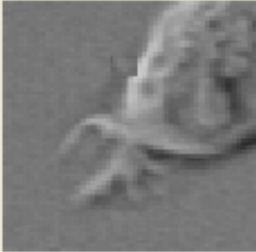
Original Image



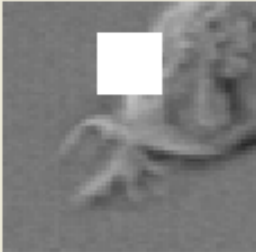


Original Image of a cell

Received Image



Tampered Areas Identified



The complex block contains two images on a light beige background. The top image is labeled "Received Image" and shows the same cell as the original image. The bottom image is labeled "Tampered Areas Identified" and shows the same cell with a white square overlaid on the central nucleus area, indicating a tampered region.

User's Manual

6.1 Introduction

This chapter shall describe in detail the functioning of the proposed model. This would entail explanation of various screens and buttons.

As has already been described, the proposed scheme consists of two separate modules given below:

- a. User End's Module
- b. Sender End's Module

We shall be describing each of these modules separately and in the process it is hoped that the simple functioning and utility of the proposed scheme would become clearer

6.2 Functional Description

We now start the functional description of the proposed scheme and at this point we separately explain the functioning of both the modules

6.2.1. Sender End Module

This is the module where the actual calculation of content based watermark and its subsequent embedding is carried out. From the user's point of view, this is done with the help of a simple front end given in Figure 6.1 below:

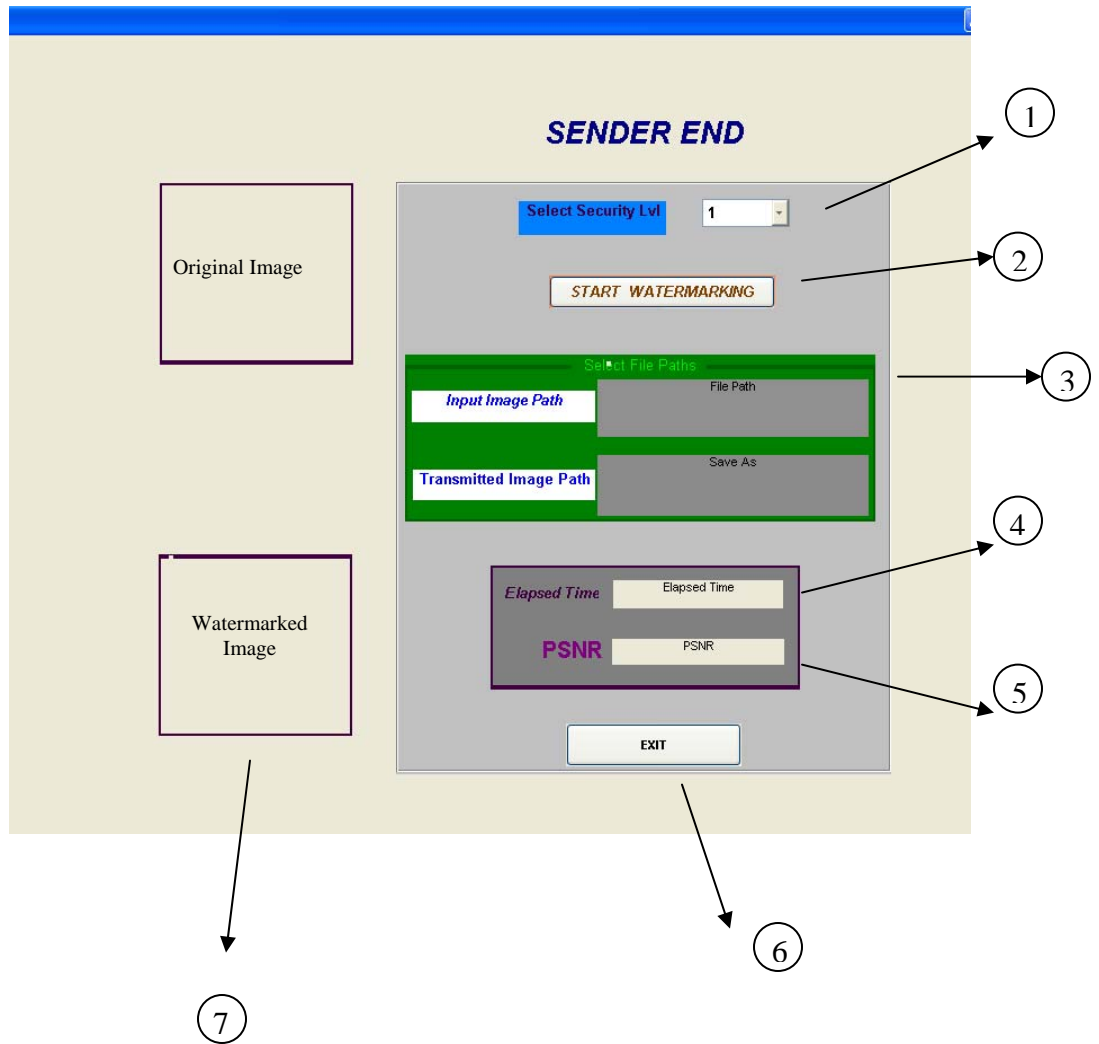


Figure 6.1 Sender End – Main Screen

6.2.1.1 Screen Description

The screen shown in Figure 6.1 above is labeled on various places. Given below is the detailed description of the screen with reference to these labels:

- a. **Label 1.** This is the popup menu for the selection of security level. Through this menu, the user can select any of the two security levels. The security level may be selected by the user depending on the

amount of security vis a vis the quality of watermarked image required. Level 1 is more secure level as the length of the content to be embedded is 128 bits. However, since all the 64 (8 x 8) LSBs are used for embedding of the content, the quality of the watermarked image is slightly reduced. On the other hand level 2 is slightly lesser secure because of the content length of 32 bit , but it also helps produce a better quality watermarked image. Moreover, the processing of calculation and embedding of the content based watermark in security level one obviously takes more processing time because of larger amount of calculations involved.

- b. **Label 2.** This is the button to start the process of content based watermark's calculation and embedding. When this button is pressed, the screen shown in Figure 6.2 below appears.

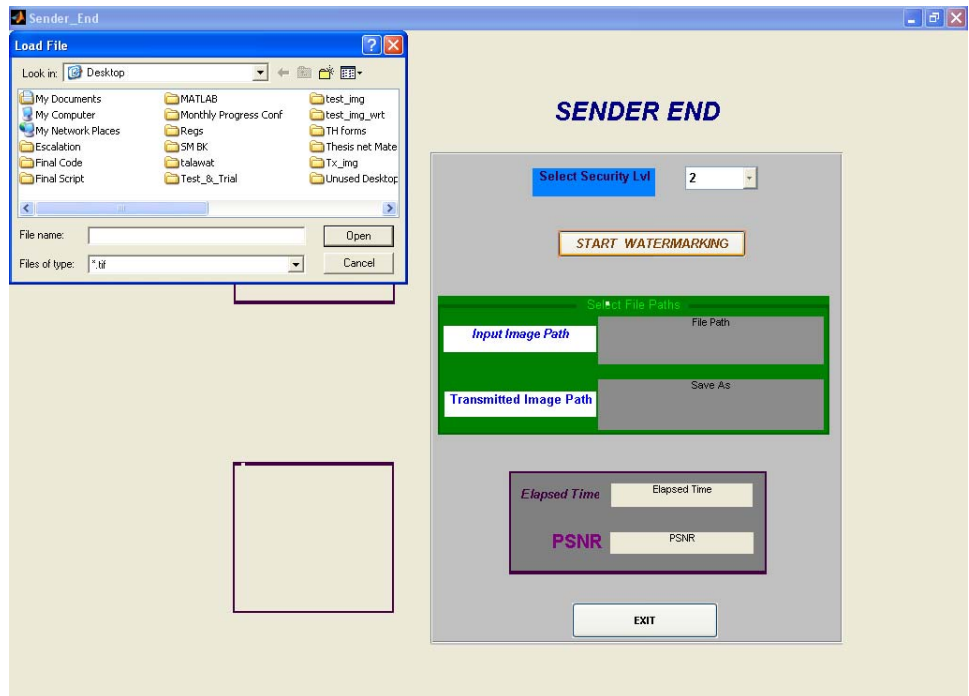


Figure 6.2 Prompt screen for selection of Input Image

This screen prompts the user to browse and select the path for the image file to undergo the watermarking process. Once the user selects his file, the screen shown in Figure 6.3 below automatically appears.

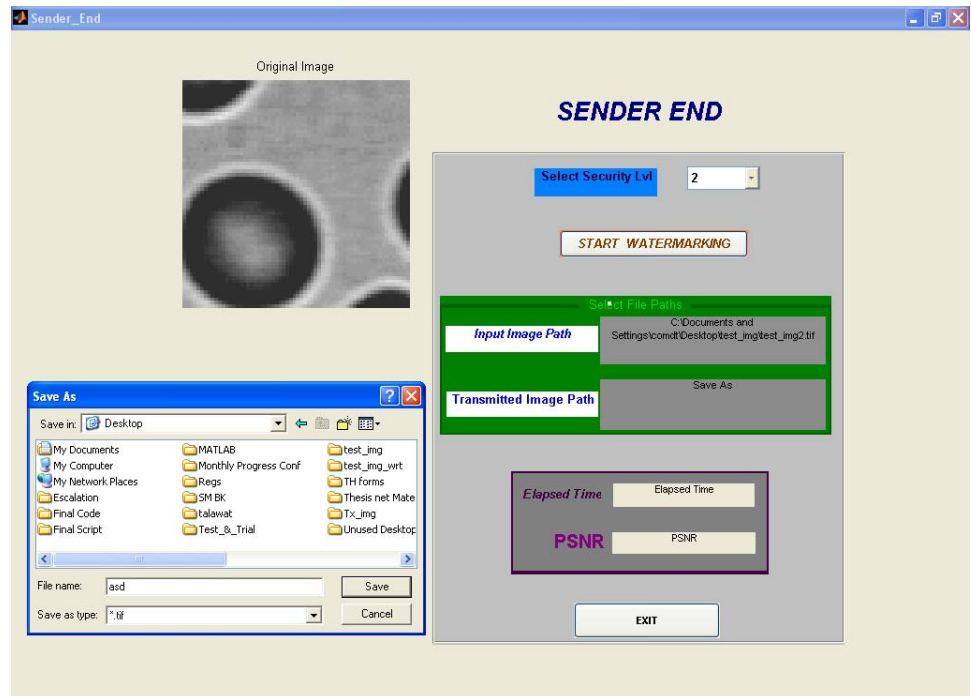


Figure 6.3 Prompt screen for Saving the Watermarked Image

This screen prompts the user to choose the path, where he wants to save the watermarked image before subsequent transfer to the receiver. It may also be noted here that the image selected by the user has appeared on the left side of the screen under the heading of “Original Image”. Moreover, the path of this image is also available on the screen’s right side in a pane which shall be discussed subsequently.

As soon as the user selects this path, the process of calculation and embedding of the watermark starts. The time that this process takes depends upon the selection of security level and the size of the image.

Once this process completes, the screen shown in Figure 6.4 below appears. This screen marks the successful termination of the watermarking process

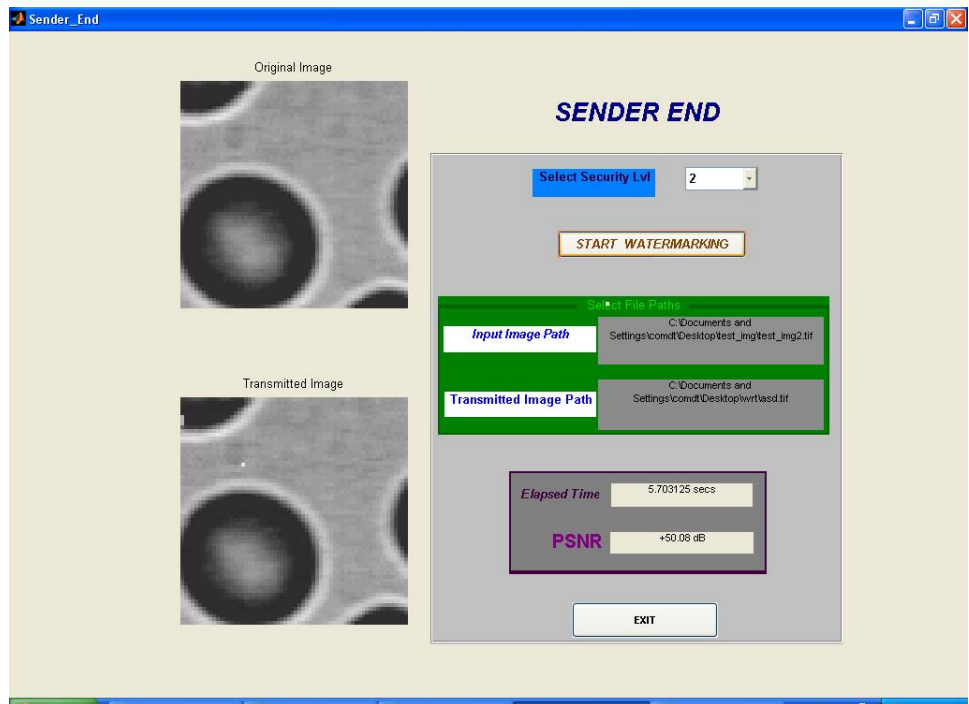


Figure 6.4 Screen showing successful Termination of the Watermarking Process

This screen not only shows the Transmitted / Watermarked image on the right side of the image below the original image but also shows other procedural values given in the succeeding explanation.

- c. **Label 3.** This label points towards the panel where the paths of the image to be watermarked and the consequent watermarked image are given. Once the user selects the path of image to be watermarked (explained above) the path of that image appears in the upper panel labeled "Input Image Path". Later when the user selects the path for saving the watermarked image and the watermarking process successfully terminates, the path where the watermarked image has been saved as per user's direction appears in the lower panel labeled "Transmitted Image Path". These panels are useful in the confirmation of selected paths.

- d. **Label 4.** This label points to the segment of the screen where we can find the time elapsed between the selection of image by the user and the termination of the watermarking process. The time is shown in seconds. The time elapsed for the same image may be different, depending upon the selection of security level. Security level one takes comparatively more time as compared to security level 2 just because of the enhanced amount of processing involved.

- e. **Label 5.** This is the label of the PSNR value which is the PSNR value of the watermarked image. Signal-to-noise (SNR) measures are estimates of the quality of a reconstructed image compared with an original image. The basic idea is to compute a single number that reflects the quality of the reconstructed image. Reconstructed images with higher metrics are

judged better. In fact, traditional SNR measures do not equate with human subjective perception. Several research groups are working on perceptual measures, but for now we will use the signal-to-noise measures because they are easier to compute.

- f. **Label 6.** This is the label of the 'Exit' button. This button may be used to exit the program at any time. We may also execute an 'Exit' from the program by pressing the 'X' button at the top right corner of the screen with the help of the mouse button.

- g. **Label 7.** This label points to two empty frames. These frames are used subsequently to house the original as well as watermarked image on the successful termination of the watermarking process as shown in Figure 6.4 above.

6.2.2. Receiver End Module

Now we come over to the receiver end module. This is the module at the receiver's end. The receiver end module is primarily required to ascertain the authenticity or otherwise of the received image. In case an image is modified or tampered in any way due to transferring channel or intentional manipulation, this module can also locate and highlight the modified / manipulated areas of the image.

We shall first have a look at the user's front end of this module which is shown in Figure 6.5 below:

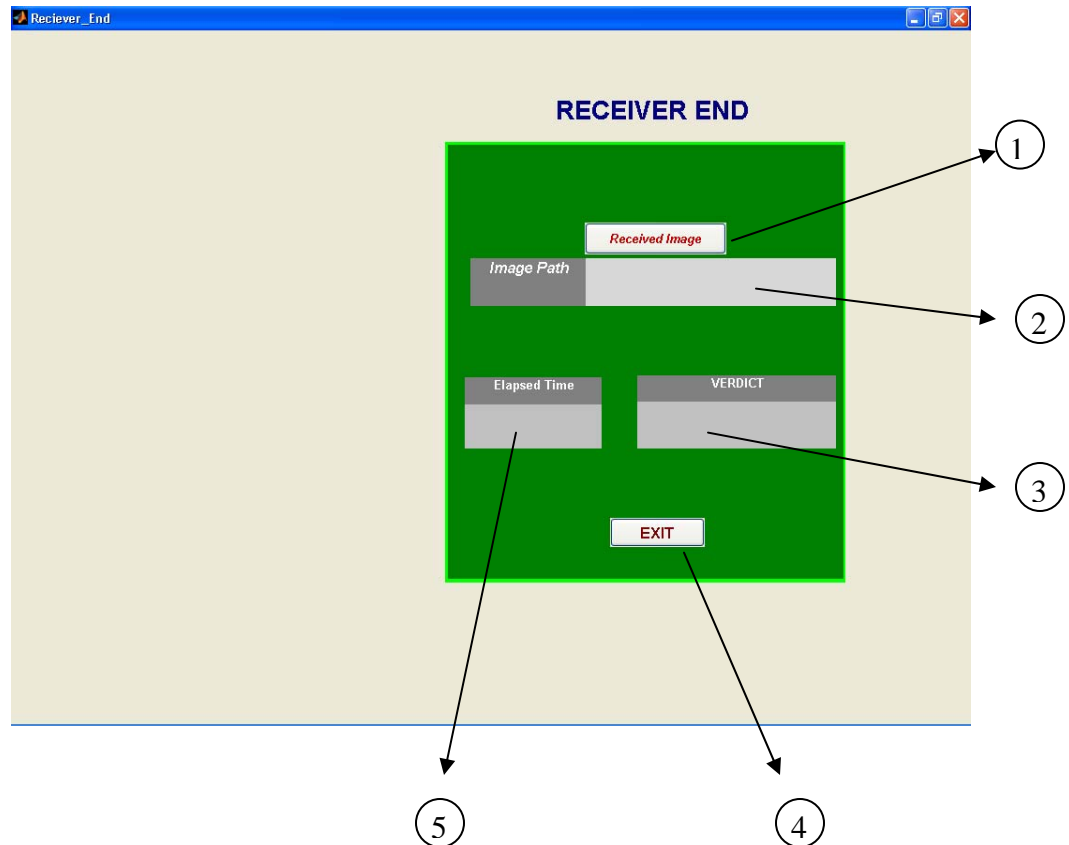


Figure 6.5 Screen showing the Receiver End Module

6.2.2.1 Screen Description

From the above Figure we can get an idea of the appearance of the receiver end module. However in the ensuing paras, we shall be giving the explanation of the various labels as well as the working of this module.

- a. **Label 1.** This label points to the button labeled “Received Image”. This is the main functional button of this screen. When this button is pressed, the user is prompted to enter the path where the received image file is stored. This prompt screen is shown in Figure 6.6 below:

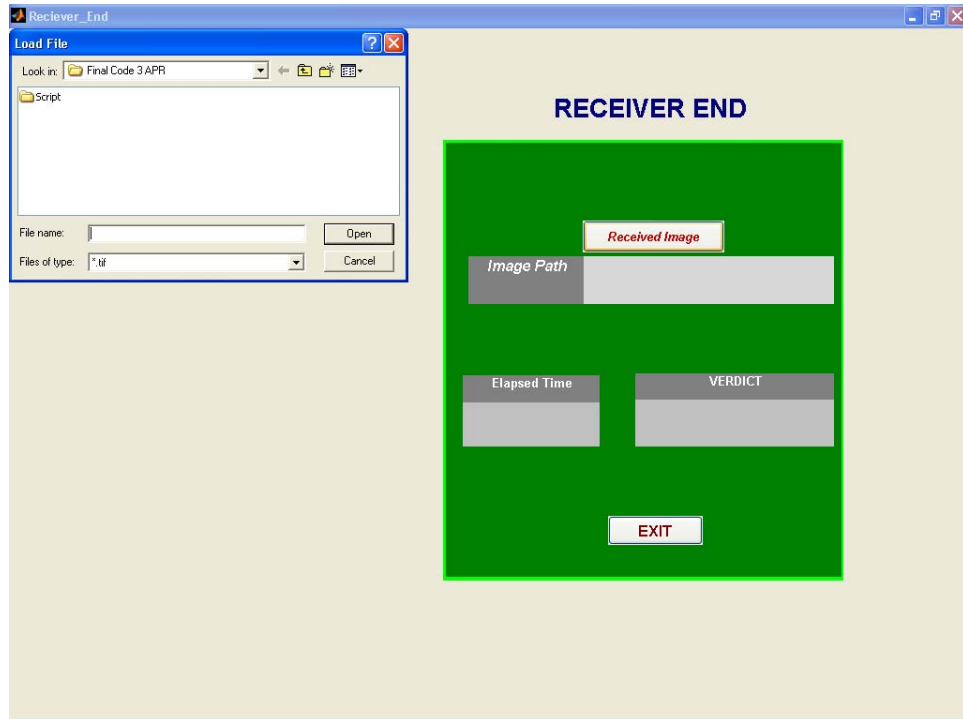


Figure 6.6 User Prompted to Enter Path of Received image File

Through this screen, the user enters the path of the received image file with the help of browsing and presses the "Open" button. Once this button is pressed the screen changes to the Figure shown in Figure 6.7 below:



Figure 6.7 User Prompted to Enter Path of Received image File

We can see from this screen that two images are shown on the left side of the screen. These are captioned “Received Image” and “No Tampering Detected” respectively. On the right side of the screen, under the columns captioned “Verdict” and “Elapsed Time”, we see the entries of “NOT TAMPERED” and “0.20 secs” respectively. This entire screen thus means that the received image has not been tampered or modified in any way and it has taken the system “0.20 secs” to process and reach a verdict.

In Figure 6.7 above, we took the example of an image which reached its destination safe and sound. However in Figure 6.8 below we see a screen achieved by pressing the

“Open” button as explained above but shows that modification / tampering of image has taken place:

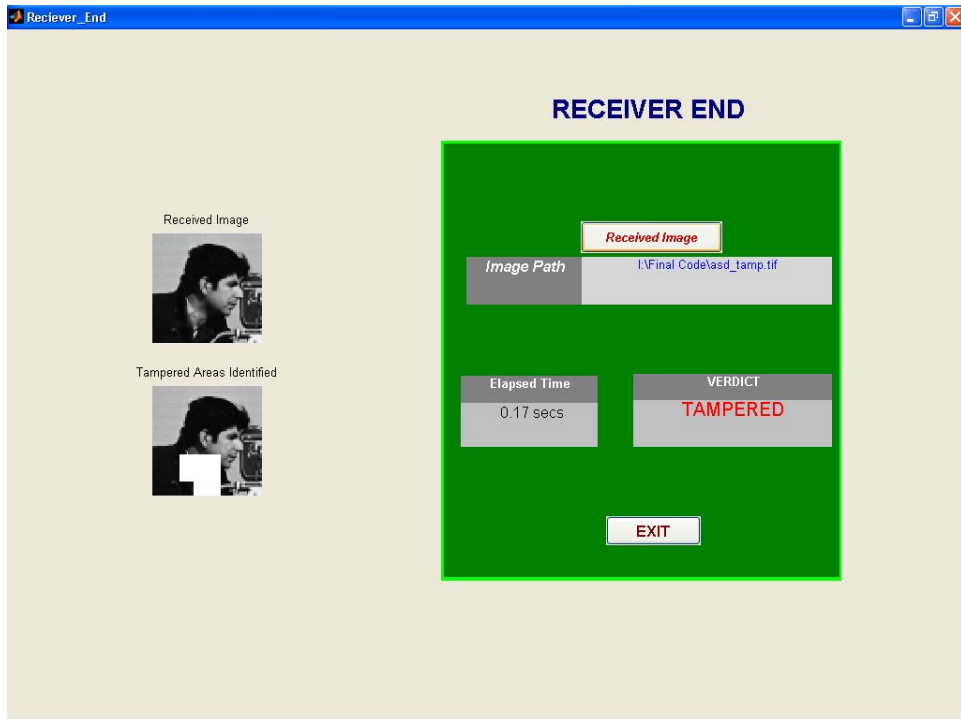


Figure 6.8 Screen Shot on Receipt of a Tampered / Modified Image

This figure again shows two images on the left side of the screen but this time they are captioned “Received Image” and “Tampered Areas Identified” respectively. For the purpose of explanation of this case, the received image has been intentionally smudged in the “neck” portion of the image. We see in the lower image that the system has successfully located and identified in white, the tampered / modified areas of the image, thereby giving a final verdict of “Tampered” in 0.17 secs.

- b. **Label 2.** This label points to the area of the screen which shows the path of the received file as selected by the

user through browsing. It is helpful in confirming the path selected by the user which may be erroneously selected during the fast browsing process.

- c. **Label 3.** This is the label showing the important slot of the verdict given by the system. This is the verdict in respect of the received image. If the image is not tampered or modified in any way, the system grades it so and in this slot gives the verdict of “Not Tampered”. However, if the system considers that the received image has either been intentionally tampered or modified by channel noise etc, it posts the verdict of “Tampered” here.

- d. **Label 4.** This is the label of the ‘Exit’ button. This button may be used to exit the program at any time. We may also execute an ‘Exit’ from the program by pressing the ‘X’ button at the top right corner of the screen with the help of the mouse button.

- e. **Label 5.** This is the label for the slot of time elapsed in processing the final verdict of “Tampered” or “Not Tampered”. The time is given in seconds.

Conclusions and Future Work

7.1 Conclusion

Technological advancements at an astronomical pace manifest continuous research and hard work to keep pace. Positive developments in this field are almost always accompanied by efforts made by someone, somewhere, to find and manipulate their loopholes. Intentional and continuous efforts are required to be a step ahead of these negative efforts.

This document has made an effort to present a simple yet effective watermarking scheme to authenticate images and at the same time, identify and localize any errors induced due to channel noise or intentional manipulations. This has been done primarily by calculating and using content based signature of the image. The signature is then mathematically manipulated with the help of a secret generator polynomial, decided upon by the sender as well as the receiver, to arrive at the fragile content based watermark which is embedded in the image. The subsequent authentication process of the watermarked image requires similar mathematical processing resulting in the precise localization and highlighting of any tampered / modified areas in the received image. The scheme has been tested on various image formats as well as image sizes and has successfully localized error/tampering with good PSNR values.

The document starts by giving an elaborate introduction of watermarking and its desirable features. This also includes various fields where watermarking and image authentication find useful applications. The thesis then develops by including basic watermarking schemes and types. This is pertinently accompanied

by information about various types of attacks and their classifications which people can or do come across. To enhance the understanding of the subject, some pertinent literature mainly comprising research in this field already being conducted, is reviewed.

After development of this background knowledge, the document comes to the design and implementation of the proposed scheme. This includes a very comprehensive explanation of the scheme, starting from basic block diagrams, followed by all the system flowcharts and algorithms, leading ultimately to the detailed functional description of the proposed scheme.

The proposed scheme has been given an exhaustive analysis and experimentation. This included several sizes, types and formats of images which were manipulated or tampered in different ways like cutting, pasting, adding noise and various manipulations through image processing software etc. All these details including the results and relevant discussion are also a part of the document.

Finally, a section has been reserved to present a simple and screen by screen user manual to help any user interested in using the proposed scheme for watermarking any image or trying to authenticate any image watermarked through this scheme

It is felt that the proposed authentication scheme can be very effectively used in the authentication of images in the fields of medical archiving, adjudicative use, insurance companies, military and survey groups etc. All of these fields cannot afford any manipulation or tampering of the images lying in their records.

Moreover, it is now high time that all the important documents and images etc must be converted to digital format rather than archiving them in paper form because of deterioration of paper quality over a period of time. This would make authentication schemes even more important in the future.

7.2 **Future Work**

Although the proposed scheme has performed fairly well under various testing parameters, nevertheless, there is always room for improvement. It is therefore suggested that the scheme presented may be made more comprehensive by incorporating the following:

- a. **Recovery**. This would entail the recovery of the areas of image, corrupted as a result of channel noise or malicious attacks etc. The author has worked on the concept of recovery with partial success and would be happy to share it with anyone working on similar lines. This info would include the ideas to work on the recovery process which is definitely feasible.
- b. **JPEG Images**. The watermarking process of JPEG format images has given very satisfactory results under the proposed scheme. However, the authentication process has not given very consistent performance as far as this image format is concerned. This aspect also requires some further deliberation.
- c. **Cropping attacks**. The system has proved its usefulness against image modification, tampering

including cut, copy, paste attacks, modification through image processing softwares and noise etc. However, cropping of image attacks still are not detected consistently by the proposed scheme.

Bibliography

- [1] C. Podilchuk and E. Delp. Digital Watermarking Algorithms and Applications. *IEEE Signal Processing Magazine*, vol. 18, no. 4, July 2001.
- [2] S. Voloshynovskiy, S. Pereira, T. Pun, J. Eggers and J. Su. Attacks on Digital Watermarks: Classification, Estimation-based Attacks and Benchmarks. In *IEEE Communications Magazine*, vol. 39, no. 8, pp. 118-127, 2001.
- [3] P. W. Wong. A Public Key Watermark for Image Verification and Authentication. In *Proc. IEEE Int. Conf. Image Processing*, vol. I, pp. 455-459, Chicago, USA, 1998.
- [4] P. Tao and A. M. Eskicioglu. A Robust Multiple Watermarking Scheme in the Discrete Wavelet Transform Domain. In *Proc. of the SPIE*, vol. 5601, pp. 133-144, Oct. 2004.
- [5] I. J. Cox, M.L. Miller and J.A. Bloom. *Digital Watermarking*. Morgan Kaufmann, San Francisco, USA, 2002.
- [6] M.U. Celik, G. Sharma, A.M. Tekalp and E. Saber. Reversible Data Hiding. In *Proc. of the IEEE Int. Conf. on Image Processing*, vol. II, pp 157-160, USA, 2002.
- [7] P. W. Wong and N. Memon. Secret and Public Key Image Watermarking Schemes for Image Authentication and Ownership Verification. In *IEEE Trans. on Image Processing*, vol. 10, no. 10, pp. 1593-1601, 2001.
- [8] D. Coltuc, P. Bolon and J. Chassery. Fragile and Robust Watermarking by Histogram Specification. In *Proc. of the SPIE Security and Watermarking of Multimedia Contents IV*, vol. 4675, pp. 701-710, USA, 2002.
- [9] C. Honsinger, "Data embedding using phase dispersion"
- [10] P.S.L.M Barreto, H.Y.Kim and V.Rigmen, "Towards secure public-key block wise fragile authentication watermarking"
- [11] J. Fridrich, "Security of fragile authentication watermarks with localization"

[12] C.H.Lin and W.S.Hsieh, "Applying projection and B-spline to image authentication and remedy"

[13] ¹Phen-Lan Lin, ²Po-Whei Huang, ³An-Wei Peng, " A Fragile Watermarking Scheme for Image Authentication with Localization and Recovery"

[14] F.H. Yeh and G. C.Lee "Toral Fragile Watermarking For Localizing And Recovering Tampered Image"

[15] Yazhou Liu , Wen Gao , Hongxun Yao , Shaohui Liu "A Texture-based Tamper Detection Scheme by Fragile Watermark"

[16] Shan Suthaharan¹, Seong-Whan Kim² "A Gradient image dependent fragile watermarking for improved security and localization"

[17] Yeung & Mintzer, "Invisible Watermarking for Image Verification", J. of Electronic Imaging, pp.578-591, July 1998.

[18] P.W.Wang, "A public Key Watermark for Image Verification and Authentication, " IEEE Inte'l conf. Image Processing, pp.455-459, Oct. 1998.

[19] Wu & Liu, "Watermarking for Image Authentication," IEEE Inte'l conf. Image Processing, pp.437-441, Oct. 1998.

[20] Tao Chen, Jingchu, "Combined Digital Signature and Digital Watermark Scheme for Image Authentication"

[21] Dr Latha Parameswaren, "Content based watermark for image authentication using independent component analysis"

[22] Sung Lin Tsan, "Automatic Image Authentication and recovery using fractal code embedding"

[23] Denken Zao, Tai Wah, "Content based image authentication system with lossless data hiding"