

ABSTRACT

With the increasing use of digital media, there is an explosion of data exchange on the Internet. Consequently, digital data can be shared quickly and massively through the Internet. The copying and replication of information has aroused great interest in multimedia security and multimedia copyright protection, and it has become an important issue in the modern digital era. Due to the increasing popularity and accessibility of the Internet by people, digital data can be distributed to millions of recipients in no time. This problem is severe in case of digital video as illegal copies of original videos can be made on optical disks even with cheap commodity computers. This illegal copying of original video cause revenue losses to stakeholders like film distributors and movie theater owners and government in the form of tax. Digital video watermarking can be used to prevent video piracy and illegal distributions. Digital video watermarking is the process of embedding noise-tolerant signal, referred to as watermark, in the host signal, which is the video file. This dissertation focuses on the embedding of watermark bits into the video frames. A given video sample is split into frames for watermark embedding. A frame is first transformed into frequency domain using Discrete Fourier Transform. A watermark, modulated with PN sequence is transformed and added to the Fourier coefficients and then the inverse is obtained of modified Fourier Coefficients to get watermarked frame. This process is repeated to the entire length of video to get watermarked video. The video watermarking algorithm proposed is robust against the attacks of frame dropping, averaging and statistical analysis. Peak Signal to Noise Ratio (PSNR) is used as the quality metric for watermarking scheme. Mathematical techniques are presented and simulation are carried out using MATLAB.

Keywords: Watermarking, DFT, FFT, Spread Spectrum, m sequences.

CHAPTER 1

INTRODUCTION

1.1 Introduction

The rapid proliferation of multimedia over internet demands sophisticated technique for secure and efficient access to information. There is an emerging need to prevent unauthorized duplication and use of digital data. Watermarking refers to hiding a digital data into a host media in such a way so as to ensure minimal or no perceptual distortion of the host media [1]. The term watermarking symbolically indicates that the mark (digital data used as watermark) should be imperceptible for humans or any other system. Most often, the host signal is a digital content, like audio, video or images. Digital watermarking embedding refers to the method of inserting information into multimedia data, called original media or cover media e.g. text, audio, image or video. The embedded information or watermark can be a serial number or a random number, customized message, control signal, creation date, data creator information, binary image, textual data or other any other digital data. The Human Visual System (HVS) [2], is not perfect for images and video and cannot detect small changes in color intensities. Therefore it is possible to modify the pixel values without making the watermark visible. If the watermark is embedded suitably and selectively in such a way so as to cause minimal change in the host data, then the watermarked signal can be made almost similar to the original signal, thereby making the watermark imperceptible. Imperceptiveness of the watermark is the most critical aspect of watermarking process and it both the art and science of modern watermarking techniques. After embedding watermark, the watermarked media are sent over Internet or some other transmission channel capable of transmission of the said media. Whenever the copyright of the digital media is under question, the embedded information is decoded to identify copyright owner. Watermark decoding process extracts the digital watermark from the watermarked media (watermark extraction) or can detect the existence of watermark in marked content (watermark detection).

1.2 Problem Statement

Watermarking video content is important to avoid piracy and illegal manipulation. However, watermarking individual frames of video in spatial domain [3] is fragile and is subjected to various kinds of attacks. Also, with a little modification, the watermark gets destroyed and cannot be detected at the receiver end. Frequency domain watermarking is more robust as

compared to spatial domain watermarking. Also, frequency domain watermarking using spread spectrum uses correlation based analysis at the time of watermark detection, which provides a way for blind watermarking. In this technique, the original unmarked video is not required at the receiver end and therefore, is useful in a broad category of applications. As the case with all watermarking systems, the perceptual fidelity of the marked content must be as low as possible. Also, the marking scheme should be robust enough to handle attacks and video editing to greatest possible extent. In this work, a tradeoff between PSNR and robustness is derived by embedding watermark in frames of the original video.

1.3 Motivation

Intellectual property protection [4] is one of the greatest concerns of internet users today. Digital videos are considered a representative part of such properties so are considered important. There is a critical need of development of techniques that prevent malicious users from claiming ownership, motivating internet users to feel more safe to publish their work online. In this work an efficient and easily implemented technique for watermarking video files is presented. The proposed watermarking process embeds the watermark in the frequency domain, by modifying the Fourier Coefficients [5] of portions of the original image, selectively and repeatedly, thereby providing robust watermarking. The embedded data can be extracted using a denoising process without the need of the original unmarked content. Thus, it provides a way of blind watermarking which is much more convenient and desirable as compared to non-blind watermarking.

Moreover, using Frequency domain for watermark embedding, one can achieve a much more robust watermark as compared to spatial domain watermarks. Also, with spread spectrum technique, the watermark energy is distributed uniformly over the host signal thereby providing much more imperceptibility as compared to spatial domain techniques.

1.4 Research Approach

This work proposes a technique of watermarking in the frequency domain. A given video is first divided into frames for watermark embedding. Each frame is then divided into RGB color planes for the purpose of watermark embedding. The color plane frame is then transformed using Discrete Fourier Transform to obtain the DFT coefficients. The watermarking bits to be embedded in the frame are first spreaded and modulated with a chosen PN sequence [6] and

then added in the Fourier coefficients using the embedding algorithm. Each frame is separately watermarked with the content. The inverse Fourier Transform is then applied to get the watermarked color plane frame. All the three color planes are then tested to find out the minimum value of the PSNR. Finally, the one with the least value of the PSNR is selected and the other two color plane frames are kept unchanged. The watermarked frames are recombined to get back the watermarked video.

1.5 Layout of Dissertation

Chapter 1 presents an overview of the subject matter and gives the problem statement and the approach for the research. Chapter 2 provides the detailed overview of frequency domain watermarking processes and those that uses spread spectrum techniques, and the parameters for performance of the network. Chapter 3 presents the proposed technique for evaluation of tradeoff between watermark imperceptibility and robustness. Chapter 4 gives the simulation results and the plots for various values of watermark strength parameters. Chapter 5 concludes the dissertation.

CHAPTER 2

LITERATURE REVIEW

2.1 Introduction

Information System Security (ISS) is a term which encompasses the study of tools and techniques to hide a message into some other data file which cannot be obtained by an intruder, or to change the message into some form that cannot be recognized by some third party. The first case is called covert communication while the other one is called cryptography. The branch of ISS that enables covert communication is steganography. The roots of watermarking lies in the study of Steganography, which means "secret writing". This word comes from the old Greek language and can be translated as cover-writing. Steganography was basically a way of transmitting hidden or secret messages between allies, being used as early as 1000 B.C. Watermarking is a special type of steganography in which the watermark is entrenched in digital data to later provide a proof of possession of that data. Figure 2.1 shows the typical ISS tree along with its various branches.

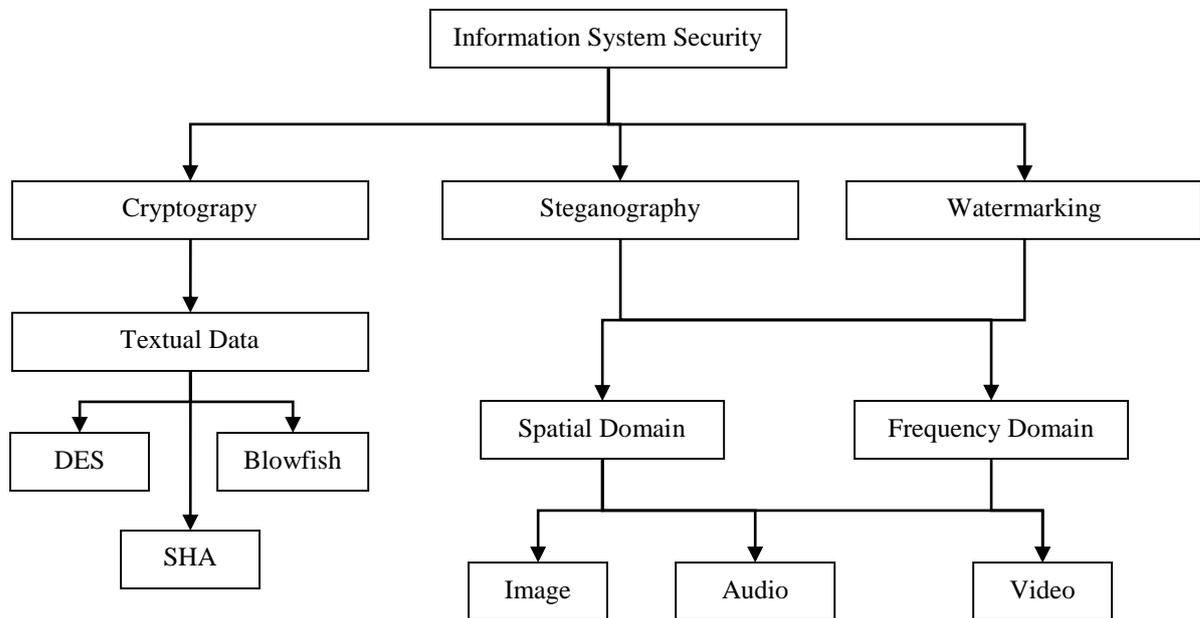


Fig. 2.1 Information Security Domains

2.2 Cryptography

Cryptography refers to the technique of encryption which is done by the sender and decryption at the receiver end. The term encryption refers to conversion of a message to some form called cipher text, using a reversible procedure, which cannot be interpreted by anyone. The reversible procedure is well-known to the receiver which can decrypt the encrypted message again into its original form. This encryption and decryption are usually performed using a secret key. When both sender and receiver have the same key, the technique is known as symmetric key cryptography. If the keys are dissimilar, the technique is known as asymmetric key cryptography. Asymmetric key cryptography is also known as public key cryptography in which the public key is known to everyone while the private key is known only to the anticipated recipient of the message.

As the Internet and other forms of electronic communication are becoming more and more popular to send and receive messages and to share data, electronic security is becoming increasingly vital. Cryptography is used to protect e-mail messages, corporate data and credit card information. One of the very well known cryptography systems used on the Internet is *Pretty Good Privacy (PGP)* as it is effective and free.

2.2.1 Data Encryption Standard (DES)

Data Encryption Standard (DES) is an extensively used methodology of data encryption that uses a private (secret) key and is considered so difficult to break that it is prohibited in many nations. There are a huge number of possible encryption keys that are generally used to resolve this matter. The key is chosen for every message randomly from among this vast number of keys. The sender and receiver both must use the same private key exactly like other cryptographic methods.

To each 64-bit block of data, 56-bit key is applied by the DES. There are various modes in which this process runs involving 16 rounds or operations. Many companies use “triple DES” despite of this being considered as a strong encryption algorithm which applies three keys one after the other but DES-encrypted message can still be broken. There are several awards being put upon for breaking this DES message. Whole over the world internet users trying out a range of keys finally deciphered the message, recognizing the key after trying out a huge number (in quadrillions) of possible keys. The messages which are sent using DES these days are more likely to encounter these kind of code breaking effort.

DES was founded at IBM labs in 1977 and it was adopted by the United States Department of Defense. It was defined in the ANSI X3.92 and X3.106 standards and also in the Federal FIPS 46 and 81 standards of encryption. In concern, the encryption algorithm could be used by aloof governments, the government of U.S. has banned the export of the software. Yet, free versions of the software are freely available on bulletin board services and several web sites. There is a general thinking that the encryption algorithm cannot be destroyed, NIST has indicated that DES will not get the certificate again as a standard for encryption and submissions concerning its replacement are being accepted. The standard which came in use after DES is known as the Advanced Encryption Standard (AES).

2.2.2 Secure Hash Algorithm (SHA)

The secure hash algorithm (SHA-1) was at first developed by the National Security Agency (NSA) as SHA-0 and later on submitted to the National Institute of Standards and Technology (NIST). However, there was a correction required in the original algorithm therefore NSA came up with the revised version of SHA-0 and named it as SHA-1. SHA-1 is based on hash function that considers a variable length message as input and produces a fixed length message as output which is referred to as the hash or the message digest of the original message. The SHA-1 algorithm is used with Digital Signature Algorithm (DSA) employed for digital signatures most importantly.

A good compression function needs to be devised in order to build a good and secured cryptographic hash function in which output bits are being affected by each input bits. SHA-1 (Secure Hash Algorithm) is a mostly used for SHA series of cryptographic hash functions, which is designed by the National Security Agency of USA and published as their government standard.

160-bit hash value is produced by SHA-1 algorithm. Original SHA that is SHA-0 also generate 160-bit hash value, but SHA-0 due to its shortcomings became outdated and is prohibited by the NSA after some time of its publication and was updated by SHA-1 Which is considered to be its revised version. SHA series produce 224-, 256-, 384- and 512-bit hash values which are responsible for providing the additional functions. The SHA-1 algorithm is quite same as MD family of hash functions. The major difference between MD family and SHA-1 is that SHA-1 uses input bits frequently during the hash function while MD4 and MD5 does not. The SHA-1

executes slowly but is surely much more secured than MD4 and MD5. This algorithm is originally proposed in May 1993 and the revised version was published in 1995. This SHA-1 algorithm is principally same as the MD4 and MD5 algorithms.

2.2.3 Blowfish

Blowfish is a block cipher which is symmetric in nature and can be efficiently used for encryption and protection of data. Its length of keys vary ranging from 32 bits to 448 bits, making it perfect for securing data. Blowfish cipher was designed in 1993 by Bruce Schneier. It is speedy and free that is open for all alternative to existing encryption algorithms. Blowfish is not patented, open and license free, available free for all users. Blowfish Algorithm is a Network, iterating an easy encryption function 16 times. The size of block is 64 bits, and the key can be of any length up to 448 bits. There is always a very complex initialization phase which needs to process before any encryption algorithm can begin executing, the data encryption on large microprocessors is very much proficient. Blowfish is particularly for those applications in which the key does not change frequently, for example.-a communications link or an automatic file encryptor. Blowfish is a variable length block cipher. It is significantly faster than other encryption algorithms when it is implemented on 32-bit microprocessors having large data caches.

A Feistel network was invented by Horst Feistel and used in many block cipher designs. It is a basic method of transforming any function called as F function into a permutation. The feistel network functions according to the points described below:

1. It Splits each block into halves
2. Right half becomes new left half
3. The new right half becomes the final result if the left half is XOR'd with the result of the right half and the key after applying f.
4. The previous rounds can be derived even if the function f is not invertible.

The Blowfish Algorithm changes or manipulates data in large blocks. Blowfish contains a scalable key of 32 bits to 256 bits and has a 64-bit block size. It uses simple operations that are proficient on microprocessors. For e.g.:-exclusive-or, addition, modular- multiplication table lookup. It comprises pre-computable sub-keys. It does not make use of variable-length shifts or bit-wise permutations, or conditional jumps.

The subkeys are generally pre compiled on large memory systems for faster execution. If precompilation of the subkeys is not done, it will result in slower operation. Precomputations is not always required for encryption it can even be done without that. It consists of a variable number of iterations. The complexity of brute force attack enhances if the applications have small key size. Also a differential attack makes a large number of iterations surplus. Therefore number of iterations can be lessened without harming the security with the reduced key size. Subkeys are used in this which are considered to be a one-way hash of the key thereby making it possible to use long passphrases without compromising the security. It also uses a design that is simple to understand and does not contain any linear structures that reduces the complication of exhaustive search.. This provides the facility of analysing and increase the confidence in the algorithm. However in practice, this algorithm is known as feistel block cipher.

Blowfish is a 64-bit block cipher having a variable-length key. The algorithm comprises of two parts: a key-expansion part and a data- encryption part. Key extension converts a key of maximum upto 450 bits into various subkey arrays adding a huge number of bytes. Data encryption is done through a 16-round Feistel network. Every round consists of a key dependent permutation along with key and data-dependent substitution. The entire operations are XORs and additions on 32-bit words. The few more operations are done which consists of four indexed array data lookups per round.

2.3 Steganography

Steganography is the science of hiding information. while cryptography makes data unreadable by the third party. There are various steganographic methods that are well- known to us, some of them are- invisible ink and microdots, secreting a hidden message in the second letter of each word of a large body of text and spread spectrum radio communication. The purpose of steganography is to hide the data from a third party. Computers and networks are used in multiple ways to hide information which are as follows:

1. Covert channels- For Example- Internet Control Message Protocol, or ICMP is used by Loki and some distributed denial-of-service tools which provides the communications channel between the hacker and a compromised system
2. Hidden text within Web pages

3. Hiding files in plain sight – For Example, the better place to hide a file is c:\winnt\system32 directory
4. Null ciphers- For Example, First letter is used to form a hidden message in an harmless text

Steganography today, however, is considerably more sophisticated than the examples suggested above. It allows users to hide a considerable amount of information in either audio or video files. These forms of steganography frequently are used in combination with cryptography. It ensures that the information is fully protected as it is both encrypted and hidden from the adversary who needs to first find and then decrypt the information. There are various uses for steganography besides the mere originality. This is considered to be the most extensively used applications of digital watermarking. Considering the history, watermark is the replication of an image, text or logo on paper stock. It ensures that the origin of document is from the authenticated source. A digital watermark can bring about the same function. for example, a graphic artist might post sample images on the Web site with an embedded signature so that he/she is not cheated upon and if others try to portray their work they can prove their ownership. Steganography files can also be used to allow communication within an underground community. There are several reports, for example- various religious minorities exploits this technique for embedding messages for a group within images that are posted on various well known sites.

2.4 Watermarking

Watermarking process refers to the technique of hiding a message into a host document to as to later claim the ownership of the digital document. Possible examples of watermarking is to embed a text or small image into a video or audio file. Watermarking is closely related with Steganography with the difference being that the latter provides a means of covert communication over a medium. In setganography, the sender and the receiver are the only two parties who knows about the communication being done, except possibly the attacker.

The goal of watermarking and stenography is entirely different.

Though, the field of digital watermarking was developed as an important technique in the last 15 years ,it is now being used for various applications.

Watermarks and watermarking techniques can be divided into various categories in many ways. Watermarking techniques is generally categorized in four types which are as follows:

1. Text Watermarking
2. Image Watermarking
3. Audio Watermarking
4. Video Watermarking

Watermarking in many ways is similar to signal processing, hence can be explained through a communication model. Some other techniques of watermarking can be described using a geometric model.

The traditional and communications based watermarking are described in much similar ways. Watermarking is used wherever a secure communication model is majorly needed. It is a process of communicating a message from the watermarking embedded to the watermarking receiver

In a general secure communication model, is depicted as shown in figure 2.2.

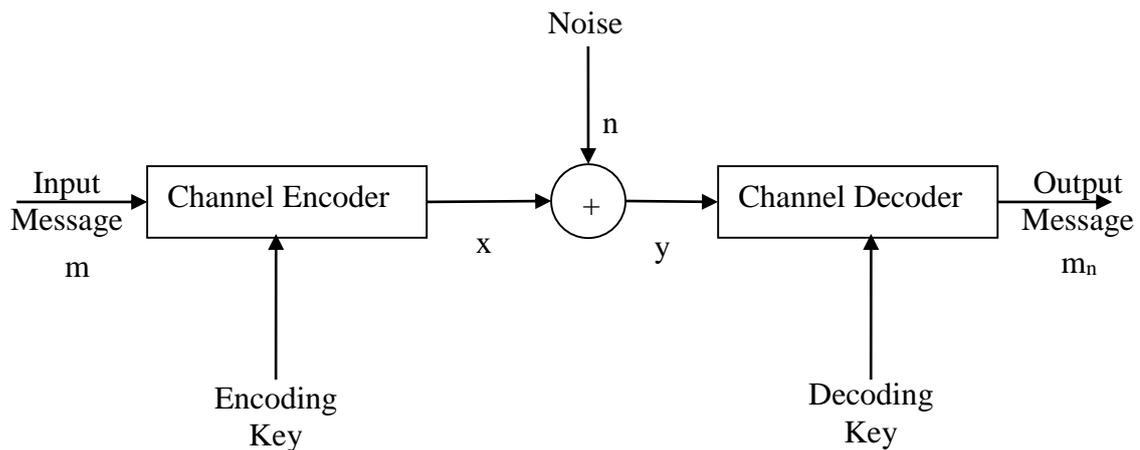


Fig 2.2 Standard model of communication with key based encoding

2.5 Desirable Characteristics of Watermarking Scheme

A digital video watermarking algorithm should optimize for the following important factors:

1. **Robustness:** It is the ability of the watermark to resist the attack by the invader to completely destroy or alter the properties such as size, quality, rotation or other visual characteristic of the video.
2. **Security:** It is the the ability of the watermark to resist the invading attempt from the unauthorized person to remove or demolish it using cryptoanalysis while video it not modified.
3. **Perceptual fidelity:** It is the apparent visual quality of the marked video when compared to the original which is an unmarked video.

2.6 Watermarking Types

2.6.1 Fragile Watermarking

Fragility is the opposite of robustness. Fragile watermarking refers to the watermarking which is least robust. Fragile watermarking finds application in tempering detection. A fragile watermark is destroyed when the host document is subjected to any change. For example, a fragile watermark in image is destroyed if the pixel intensities are changed. In this case, if the image is transferred to some recipient, he/she may check it and deduce that the image is been subjected to some change and it is not the original copy which is been created.

Tampering localization in Fragile watermarking

In most of the case, the tempering detector comes up with the result that the document which is subjected to being test is tempered or original. However, it is much more beneficial is the tempering detector can specify which portions/parameters of the document are being changed. High resolution tempering localization refers to the technique of specifying which regions of the host image are being changed by the tempering operation.

Tampering detection with low false positive

A good tempering detection systems results in low false positive and low false negative alarms. A false negative refers to the condition that tempering detector output a originality message whereas tempering is not done actually. A false positive alarm refers to the condition that tempering detector output a tempering message even when a tempering is done. Designing a tempering detector is a difficult and complex task in case of fragile watermarking.

Blind and Non Blind detection

A blind watermarking technique refers to the technique of watermark detection in which, at the time of watermark detection, the original unmarked document copy is not needed. On the other hand, in non-blind watermarking technique, the original image is needed at the detector end for watermark detection. Generally, Blind watermarking is needed for watermark detection and Non-blind watermarking techniques are needed for watermark detection and extraction. Blind watermarking typically uses correlation based techniques for watermark detection.

2.6.2 Semi Fragile Watermarking

Fragile methods are mainly applied to content authentication and integrity evidence, because they are fragile to almost all modifications. In contrast, semi-fragile methods are robust to incidental modification such as JPEG compression, but delicate to other modifications. The authentication watermark can be classified into fragile watermark and semi-fragile watermark according to their fragility and sensitivity. The fragile watermark is very sensitive and designed to detect every possible alteration in marked image, so if it fits to verify the integrity of data, and is viewed as an alternative verification solution to a standard digital signature scheme. In most multimedia applications, minor data modifications are satisfactory as long as the content is authentic, so the semi-fragile watermark is developed and widely used in content verifying. Semi-fragile watermark fragile to malevolent modifications while robust to incidental manipulations is drawing many attentions in image authentication. Though, watermark security has not received enough notice yet. The main advantage of employing semi fragile watermarking over digital signature and fragile watermarking technology is that there is greater prospective in characterizing the tamper distortion, and to design a method which is robust to various kinds of processing. Lossless and lossy compression, light additive noise, smoothing and format conversion are typically tolerable modifications since image content interpretation is not exaggerated but there is no need to get assured of the exact representation during exchange and storage. The alteration on the documents can occur unintentionally or can be fixed intentionally. The so-called unintentional or unplanned alterations typically occur from such varied facts as bit errors during transmission, storage, and signal processing operations such as filtering, sharpening, contrast enhancement, and compression. The major distinction, is whether the content is altered as in malicious and intentional attacks or only the illustration,

but the content of the document is not altered, as occurs in unintentional, non-malicious cases. Deliberate or malicious alterations, on the contrary, are understood to be due to an explicit fake attempt by a pirate with the clear purpose of changing the contents of a document. The line of separation between these two attacks categories is, however, not always clearly specified, as it depends very much on the application domain.

2.6.3 Robust Watermarking

The purpose of the watermark is to supply some additional information about the digital media, to attain control over the copy process of a particular digital media to verify image integrity. The information used by the watermark can be accessed using a detection algorithm provided that the secret key is known. A significant property of a watermark is its robustness with respect to digital media distortions. That is the watermark is legible from images that underwent common image processing operations, such as filtering, histogram manipulation, lossy compression, noise adding and a variety of geometrical transformations. Watermarks intended for copyright protection, access control or fingerprinting must also be embedded in a secure form. This implies that an attacker who knows all details of the embedding algorithm except the secret key should not be able to disturb the watermark further than detection. In further applications, such as adding additional captions to images or subtitles in several languages to movies, there is little incentive for intentional removal of the watermark, and the detecting key can be made public. The amount of bits carried by the watermark could be as low as one bit to several hundred bits. There is a swapping between the capacity and the robustness of the watermark. This tradeoff is depicted in the figure 2.3 given below.

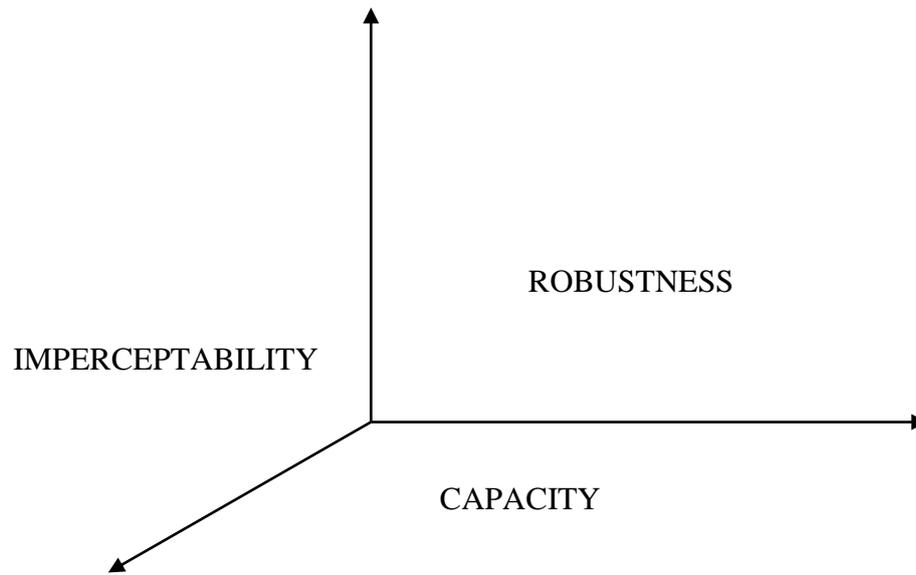


Fig 2.3 The tradeoffs between imperceptibility, capacity and robustness.

2.7 Classification of Watermarking Schemes

2.7.1 Spatial Domain Watermarking

Various methods enable watermarking in the spatial domain. The simplest method is just to turn over the lowest-order bit of chosen pixels. This goes well only if the image is not subject to any alteration. A more robust watermark can be entrenched by superimposing a symbol over an area of the picture. The resultant mark is either noticeable or not, depends upon the intensity value. Picture cropping, a common operation of image editors, can be used to get rid of the watermark. Spatial watermarking can also be done using color separation. Using this way, the watermark appear in only one of the color bands. This render the watermark visibly delicate such that it is difficult to detect under regular viewing. Though, the mark appears right away when the colors are separated for printing. This render the document useless for the printer unless the watermark is removed from the color band. This approach is used on the commercial purposes for journalists to inspect digital pictures from a photo stock house before buying unmarked versions.

2.7.2 Transform Domain Watermarking

Transform Domain watermarking techniques refers to the techniques in which the domain of the watermarking is changed from one base to another for watermark embedding purpose and later-on, transformed to the base domain so as to get the watermarked signal. An example of conversion is the conversion from time domain to frequency domain which can be done using Fourier Transform. Transformation to some other domain like wavelet domain is also possible using Discrete Wavelet Transform. In most of the cases of digital watermarking, transform domain watermarking results in much more robust watermarking as compared to base domain watermarking.

For example, in a particular case of image watermarking, watermark can be added in Spatial Domain (in pixels) or in Frequency Domain (in Fourier Coefficients of Pixels). Table 2.1. shows a small comparison between the two different techniques. The comparison between the two as mentioned in the table is a generic one and specialized techniques may vary from the table.

**TABLE 2.1
COMPARISON BETWEEN VARIOUS WATERMARKING TECHNIQUES**

Parameter	Spatial Domain	Frequency Domain
Computation Cost	Low	High
Robustness	Fragile	Robust
Perceptual Quality	High Control	Low Control
Capacity	High (Depending on the size of the cover file)	Low
Example of Applications	Mainly Authentication	Copyright

2.8 Watermarking Attacks

One classification of the wide class of existing attacks contains four classes of attacks:

1. Removal attacks
2. Cryptographic attacks
3. Geometric attacks
4. Protocol attacks.

2.9 Spread Spectrum Based Watermarking Scheme

This work concentrates mostly on one class of watermarking schemes viz. Spread Spectrum (SS) based watermarking schemes. In SS watermarking, the watermark (or the copyright information) is modulated using a Pseudo Random (PR) digital sequence. This is corresponding to spreading the information in the frequency domain. The seed of the PR sequence generator can be used as a key to identify the legitimate copyright owner. This modulated sequence is then added to the video signal (host signal), at an imperceptibly small strength, to generate the watermarked signal. A matched filter can be used for recovering the watermark. Like any communication system, the intentional and incidental tampering on the watermarked signal acts as interference. Further to this, the host signal itself is interference to the watermark detector.

In spread spectrum communications, the band-limited source information is modulated on to a pseudo-random (PR) chip sequence before transmission. The chip sequence has an extended bandwidth, in general a few 1000s time broader than the source bandwidth. It has nearly flat spectrum in this band. The modulation leads to the spreading of the low frequency information in a larger spectral interval, but at a condensed strength. The information can be retrieved by coherent demodulation, using the same PR sequence. Thus, the PR sequence more accurately, the seed of the PR sequence generator acts as a secret key between the transmitter and the receiver.

2.9.1 Pseudo Noise (PN) Sequences

Pseudo Noise sequences, also referred to as pseudo random sequences, are sequences that are characteristically generated and thus far possess some properties that one would expect to find in randomly generated sequences. PN sequences applications include signal synchronization, spread-spectrum communications, navigation, radar ranging, random number generation, multipath resolution, signal identification and cryptography in multiple-access communication systems. The correlation between two sequences $\{x(t)\}$ and $\{g(t)\}$ is the complex inner product of the first sequence with a shifted version of the second sequence. The correlation is known to be:

1. an autocorrelation if the two sequences are the same
2. a cross-correlation if they are distinct

3. a periodic correlation if the shift is a cyclic shift
4. an aperiodic association if the shift is not cyclic and,
5. a partial-period correlation if the inner product consists of only a partial segment of the two sequences.

Binary m sequences, are the well-known family of PN sequences. The run-distribution, balance, and autocorrelation properties of these sequences imitate those of random sequences. It is possibly the random-like correlation properties of PN sequences that make them most attractive in a communications system, and it is ordinary to refer to any collection of low-correlation sequences as a family of PN sequences.

Feedback shift register sequences have been extensively used as synchronization codes, masking or scrambling codes, and for white noise signals in communication systems, signal sets in CDMA (code division multiple access) communications, major stream generators in stream cipher cryptosystems and a random number generators in many cryptographic primitive algorithms, and for testing vectors in hardware design. The m codes are of length $2^N - 1$, where N is the number of shift registers in the circuit used to generate the m sequence. The important properties of binary sequences are :

1. Balance Property : All the codes have equal number of zeros and ones.
2. Shift Property : If the code is shifted by any nonzero number of elements, the resultant sequence will have half of its elements, the same as in the original sequence and half of the elements are different from the original sequence.

M sequence generator can be illustrated with the following example:

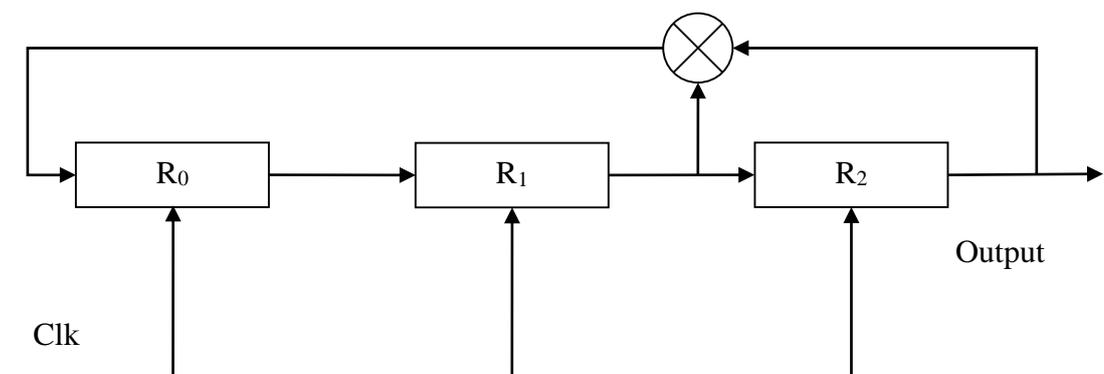


Fig 2.4 Linear Feedback Shift Register (LFSR) for m sequence Generation

One way to design this circuit is to start with a generator polynomial. The feedback taps in the feedback shift register . These are selected to correspond to the coefficients of the primitive polynomial. The LFSR circuit shown above corresponds to the polynomial

$$g(x) = X^3 + X^2 + 1$$

Binary sequences drawn from the alphabet {0,1} are shifted through the shift register in response to clock pulses. After each clock time, the register shifts all its contents to the right. The particular 1s and 0s occupying the shift register stages after a clock pulse are called states.

The illustration of the contents of the shift register along with the clock cycles is illustrated as shown below:

TABLE 2.2

ILLUSTRATION OF THE REGISTER CONTENTS AND THE CIRCUIT STATES WITH THE CLOCK CYCLES

Time	R0	R1	R2
0	1	0	0
1	0	1	0
2	1	0	1
3	1	1	0
4	1	1	1
5	0	1	1
6	0	0	1
7	1	0	0

Thus, the m sequence generated by this LFSR circuit is 0101110

2.10 Video Watermarking

A video is basically a sequence of frames which as passed at such a rate so as to give the impression of a motion picture. some video capable digital cameras and camcorders offer not only the ability to select resolutions of 640x480, 1280x720, or 1920x1080 along with the ability to select from one of a number of frame rates such as from 24 till 30 or even 60 frames per second. The video under consideration can be interlaced or progressive. for example 60i indicates that the video will be 60 frames per second where each field is only half of a

frame. The term interlacing means that each field consists of the odd rows or even rows in the picture. Since each update (very 1/60th of a second) only updates half the frame, and one can never actually see an entire frame. half a picture mixed with the other half taken 1/60th of a second later is actually visible. Of course, this happens so fast that human eyes perceive the data as if it were full frames. On the other note, 30p video known to display one entire frame every 1/30th of a second so an entire frame is displayed at once. 30p resolution video displays a full frame at a time, n complete frame or snapshot of the scene is displayed for 1/30th of a second and then the next frame is displayed for 1/30 of a second and so on. That is why 30p is generally regarded as better than 60i as far as resolution.

Video watermarking has its roots in image watermarking as in most of the cases, individual frames of the video are watermarked. The individual frames are generally images in some format, which can be suitably watermarked using techniques available for image watermarking. Cryptographic information is embedded in Video watermarking derived from frames of digital video into the video itself. Practically, this technology works separately of the video file format or codec since the watermark is part of the video, than part of the file format or DRM system. Difference can never be perceived by the user viewing the video between the original, marked video and the unmarked video, but embedded information can certainly be obtained by watermark extraction application.

The watermarking algorithm proposed in this dissertation optimizes for three distant factors:

1. **Robustness:** It is the ability of the watermark to resist attempts by an invader to destroy it by modifying the quality, size, rotation, or other visual aspects of the video. Robustness means Resistance to non-targeted modifications and common media operations. For instance the Stirmark or Mosaik tools attack the robustness of watermarking algorithms with geometrical distortions. The watermark has to be fragile to detect altered media for manipulation recognition. There are certain major issues which need to be ensured for robustness. They include that the watermark detector must be able to detect it and it must be present in digital media.
2. **Security:** the ability of the watermark to resist attempts by a sophisticated attacker to destroy or remove it through cyptoanalysis, without alteration of the video itself. It states if the embedded watermarking information cannot be removed beyond reliable

detection by targeted attacks considering a thorough understanding of the embedding algorithm and the detector along with the knowledge of at least one watermarked data. The security feature of watermarking implies that the watermark should be tricky to remove or alter without distortion of the host signal. As all watermarking systems search for protection of watermark information. Watermarking security can be considered as the ability to guarantee secrecy and integrity of the watermark information.

3. **Perceptual fidelity:** the perceived visual quality of the marked video compared to the original and unmarked video.

Video watermarking can be categorized based on various characteristics such as:

1. **Perceptibility:** Watermark is inserted in an image as noise and operates the perception masking capabilities of the human eye to make this watermark visible. Watermarks are hidden from user perception. It does not hinder with the image and can only be perceived if the user concentrate on the watermark.
2. **Continuous or Sampled Data:** The information on a cassette is not digitized and does not need any sampling to be played. Documents are stored as either continuous (analog) or sampled (digital) Data. .wav is a file that stores information about the sound to be played back in a discrete and discontinuous manner.

Some of the applications of video watermarking are:

1. **Security:** copyright identification traitor tracing i.e, active fingerprinting authentication copy control
2. **Media Enhancement:** broadcast monitoring device control enrichment (functionalities and/or metadata's with forward compatibility) improve compression performances and error recovery & correction
3. **Copyright protection:** Digital content can be embedded with watermarks depicting metadata identifying the copyright owners. Digital watermarking can be used to identify and protect copyright ownership.
4. **Tamper proofing:** Digital watermarks are delicate in nature, can be used for tamper proofing. Digital content can be entrenched with fragile watermarks that get destroyed

whenever any sort of modification is made to the content. These kinds of watermarks can be used to authenticate the content

5. **Media forensics:** Forensic watermarking is used not only to gather evidence for various criminal cases, but it is also used to enforce contracts between a content owner and the people or companies with which it shares its content. Forensic watermark applications enhance a content owner's ability to detect and respond to misuse of its assets

2.12 Chapter Summery

Watermarking refers to a pattern of bits inserted into a digital image, video or audio file that identifies the file's copyright information such as author, rights, etc. The name comes from the faintly visible watermarks imprinted on stationery that identify the producer of the stationery. The principle behind digital watermarks is to offer copyright protection for intellectual property in digital format. Unlike printed watermarks, which are proposed to be visible up to some extent, digital watermarks can also be completely invisible, and in the case of audio clips, it can be inaudible. Furthermore, the actual bits representing the watermark must be spread throughout the file in such a way that they cannot be recognized and manipulated. And finally, the digital watermark should be robust enough so that it can survive normal changes to the file, such as reduction from lossy compression algorithms. Fullfilling all these requirements is not an easy task, but there are a number of techniques offering challenging results through research. Watermarking is a branch of Information Systems Security and its roots lies in data embedding and information hiding.

CHAPTER 3

PROBLEM STATEMENT

3.1 Proposed Video Watermarking using PN sequences

The current work focuses on video watermarking in frequency domain using Discrete Fourier Transform. A given video is segregated into the frames which are then separately processed. The embedding algorithm works as follows:

Embedding Algorithm:

1. Separate the given video into frames. The frame rate typically depends on the format of the video as well as video quality.
2. Let the frame size be $m \times n$. Each frame is then divided into red, green and blue planes for the purpose of watermark embedding.
3. A binary watermark is modulated with a given PN sequence and embedded in magnitude of the Fourier Coefficients of the RGB planes one by one.

4. The color plane is selected for watermark embedding which provides the highest value of PSNR.
5. Inverse Fourier Transform is obtained for the plane to get back the watermarked plane.
6. The planes are combined again to get back the watermarked frame.
7. The process is repeated for all the frames of the video to get the watermarked video.

The process of watermark embedding is illustrated in the figure 3.1.

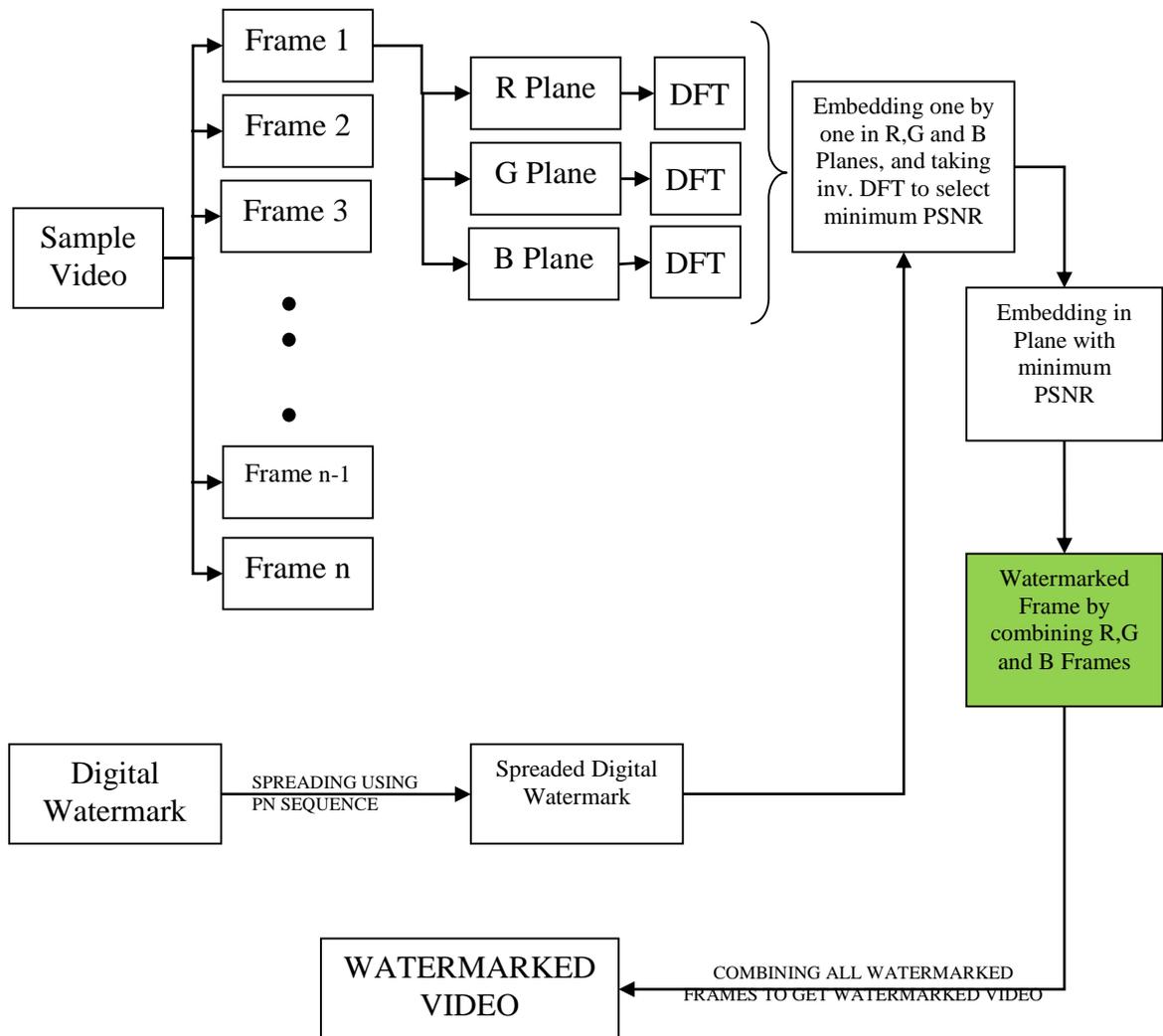


Fig 3.1 Proposed Architecture of Embedding Process

Extraction Algorithm:

1. Separate the given video into frames. Obtain the Red, Green and Blue planes of frame.

2. Obtain the DFT of each of the color plane.
3. Given a PN sequence, perform the correlation analysis with each of the plane, and corresponding de spreading. A value well above threshold gives the watermark bits.
4. Repeat the process for all the frames to ensure the presence of watermark even in the case of frame dropping. As watermark is embedded in all the frames, the scheme provides a robust watermarking technique.

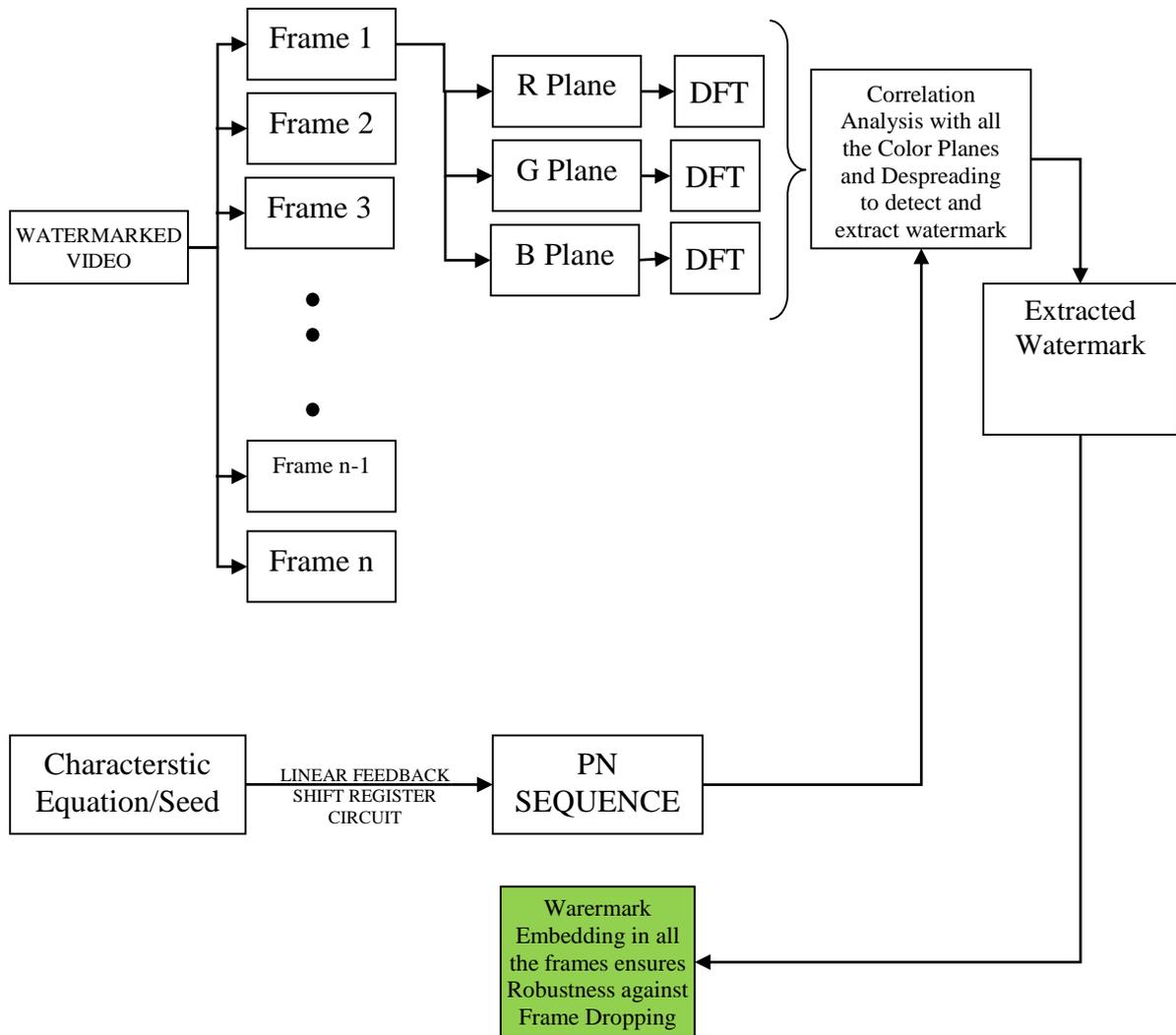


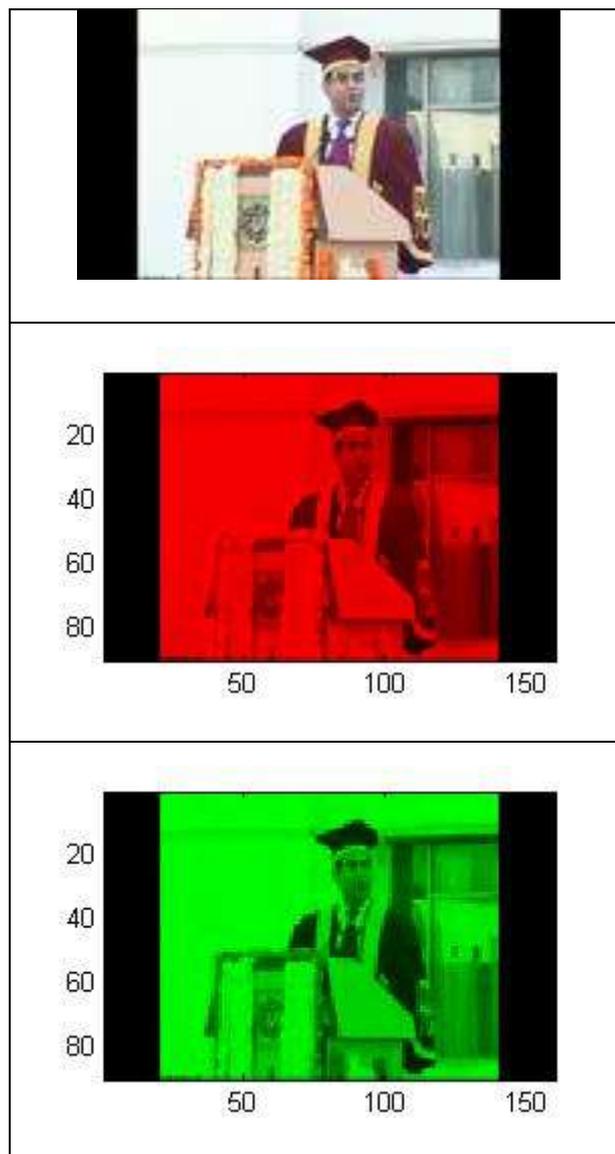
Fig 3.2 Proposed Architecture of Extraction Process

3.2 Watermark Embedding Scheme

In this dissertation, the watermarking scheme is carried out over 24 bit color videos. Consider an arbitrary frame of sample video to be watermarked, as shown in table 3.1. Each frame is specifically an image and can be visualized in some format. Also, each pixel of the image is a

24 bit value comprising of 8 bits each of Red, Blue and Green color. Pixel value separation in RGB color planes is performed to embed the watermark.

Figure 3.3 is a sample video frame to be watermarked (frame number 191 in the frames displayed in chapter 4). Figure 3.2, 3.3 and 3.4 shows the Red, Green and Blue planes of the frame. The proposed method takes the Fourier Transform of each of these planes and embed the watermark modulated with PN sequence to each one of the R, G and B plane. PSNR value of each of these frames is computed and finally the one with highest value of PSNR (or least value of Mean Square Error) is selected. The rest two bit planes are kept intact. All the color planes are these merged to get the watermarked frame. All such frames are then combined to get the watermarked video.



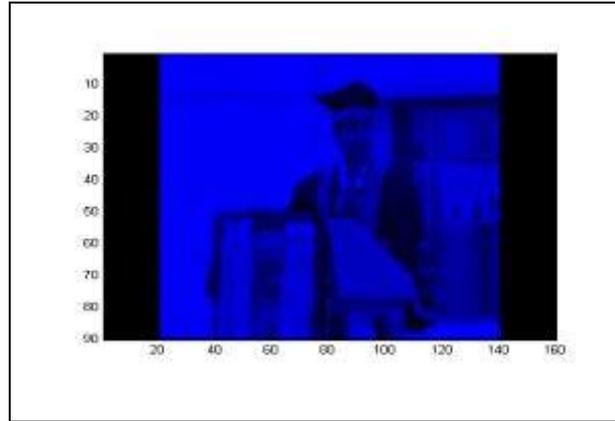


Fig 3.3 Original Color Frame and Decomposition into Red, Green and Blue color Planes

3.3 Watermark Detection

At the receiver end, the watermarked video is again partitioned into frames. Each frame is then further separated into Red, Green and Blue planes. The Fourier Coefficients of each of the color planes are computed and the presence of watermark is ensured by a correlation test with the same PN sequence. If the value of correlation between PN sequence and the extracted PN sequence is greater than a certain threshold, then one can ensure the presence of watermark in the given image plane. However, proper embedding of the modulated PN sequence in the Fourier coefficients is a tricky issue and can be done in several ways. The two dimensional Discrete Fourier Transform of a set of n numbers gives n complex number of the form $a+ib$, where $a, b \in \mathbb{R}$, and $i = \sqrt{-1}$. In this dissertation, Binary phase only filter (BPOF) signature is used to embed watermark in the Fourier Coefficients.

3.4 Watermark Embedding in Fourier Coefficients

Consider an imaginary pixel matrix of dimension 8×8 as illustrated in figure 3.7.

11820331	6494913	10415645	12449088	12095072	8289687	1192490	9428863
12169468	6509693	7991089	5169284	4785451	12229114	1571845	2496992
9167064	3292591	14391728	1609133	659523	1655136	14452314	4201615
9041585	2030055	6468675	12742307	9000882	14376745	11324348	5311066
15459006	1376940	14296363	4963269	12989979	9841832	6723410	2274452
2209273	4487618	6910443	4115055	11229072	2794613	8091716	15981385
8949244	3091192	4421592	4159465	3213446	702114	8429983	12942187
13127260	4490681	11180514	6083491	4301314	14695071	16068162	494159

All the above values lie in the range $[0,16777215]$. The above table can be converted into binary values of length 24 bit in which the first 8 bits represent Red, the next 8 bits represents

Green and the last 8 bits represents the Blue color. The corresponding RGB bit planes are as shown in figure 3.8, 3.9 and 3.10.

10110100	01100011	10011110	10111101	10111000	01111110	00010010	10001111
10111001	01100011	01111001	01001110	01001001	10111010	00010111	00100110
10001011	00110010	11011011	00011000	00001010	00011001	11011100	01000000
10001001	00011110	01100010	11000010	10001001	11011011	10101100	01010001
11101011	00010101	11011010	01001011	11000110	10010110	01100110	00100010
00100001	01000100	01101001	00111110	10101011	00101010	01111011	11110011
10001000	00101111	01000011	00111111	00110001	00001010	10000000	11000101
11001000	01000100	10101010	01011100	01000001	11100000	11110101	00000111

Red Plane

01011101	00011010	11101110	11110101	10001110	01111101	00110010	11011111
10110000	01010100	11101111	11100000	00000101	10011001	11111100	00011001
11100000	00111101	10011001	10001101	00010000	01000001	10000110	00011100
11110110	11111001	10110100	01101110	01010111	01011111	11001011	00001010
11100010	00000010	00100101	10111011	00110110	00101100	10010111	10110100
10110101	01111001	01110001	11001010	01010111	10100100	01111000	11011011
10001101	00101010	01110111	01110111	00001000	10110110	10100001	01111011
01001110	10000101	10011001	11010011	10100010	00111010	00101110	10001010

Green Plane

00101011	11000001	00011101	01000000	01100000	10010111	00101010	01111111
11111100	01111101	00110001	10000100	00101011	11111010	00000101	11100000
11011000	10101111	10110000	10101101	01000011	01100000	01011010	10001111
10110001	11100111	01000011	10100011	10110010	00101001	10111100	01011010
10111110	10101100	00101011	11000101	00011011	10101000	01010010	10010100
11111001	11000010	11101011	01101111	10010000	01110101	01000100	01001001
11111100	11111000	11011000	11101001	10000110	10100010	10011111	01101011
01011100	10111001	11100010	10100011	00000010	10011111	01000010	01001111

Blue Plane

The corresponding decimal values of each of the planes are:

180	99	158	189	184	126	18	143
185	99	121	78	73	186	23	38
139	50	219	24	10	25	220	64
137	30	98	194	137	219	172	81
235	21	218	75	198	150	102	34
33	68	105	62	171	42	123	243

136	47	67	63	49	10	128	197
200	68	170	92	65	224	245	7

Red Plane

93	26	238	245	142	125	50	223
176	84	239	224	5	153	252	25
224	61	153	141	16	65	134	28
246	249	180	110	87	95	203	10
226	2	37	187	54	44	151	180
181	121	113	202	87	164	120	219
141	42	119	119	8	182	161	123
78	133	153	211	162	58	46	138

Green Plane

43	193	29	64	96	151	42	127
252	125	49	132	43	250	5	224
216	175	176	173	67	96	90	143
177	231	67	163	178	41	188	90
190	172	43	197	27	168	82	148
249	194	235	111	144	117	68	73
252	248	216	233	134	162	159	107
92	185	226	163	2	159	66	79

Blue plane

The two dimensional FFT of the Blue plane can be obtained as shown in the figure given below:

TABLE 3.1

FFT COEFFICENTS FOR BLUE PLANE OF JPG IMAGE OF FRAME

1.4710 + 0.0000i	1.5230 + 0.0000i	1.0410 + 0.0000i	1.2360 + 0.0000i	0.6910 + 0.0000i	1.1440 + 0.0000i	0.7000 + 0.0000i	0.9910 + 0.0000i
-0.2050 - 0.0262i	-0.0603 + 0.0893i	-0.0331 + 0.2840i	-0.1182 + 0.0452i	-0.1269 + 0.0140i	0.1605 + 0.0554i	-0.1708 + 0.0273i	0.0780 - 0.1506i
-0.2350 - 0.2320i	-0.0580 + 0.0970i	-0.3200 + 0.0090i	-0.1450 + 0.0830i	-0.0780 - 0.0070i	0.0610 - 0.1670i	-0.1250 + 0.1810i	0.0250 - 0.1280i
-0.0890 - 0.0982i	0.1023 - 0.0567i	0.0051 + 0.2040i	-0.1478 - 0.0748i	0.2649 - 0.1200i	-0.1945 - 0.0766i	0.0908 - 0.1107i	-0.1200 - 0.0786i
-0.0690 + 0.0000i	0.0530 + 0.0000i	-0.1130 + 0.0000i	0.0980 + 0.0000i	-0.0430 + 0.0000i	0.0100 + 0.0000i	0.0460 + 0.0000i	0.0590 + 0.0000i
-0.0890 + 0.0982i	0.1023 + 0.0567i	0.0051 - 0.2040i	-0.1478 + 0.0748i	0.2649 + 0.1200i	-0.1945 + 0.0766i	0.0908 + 0.1107i	-0.1200 + 0.0786i
-0.2350 + 0.2320i	-0.0580 - 0.0970i	-0.3200 - 0.0090i	-0.1450 - 0.0830i	-0.0780 + 0.0070i	0.0610 + 0.1670i	-0.1250 - 0.1810i	0.0250 + 0.1280i
-0.2050 + 0.0262i	-0.0603 - 0.0893i	-0.0331 - 0.2840i	-0.1182 - 0.0452i	-0.1269 - 0.0140i	0.1605 - 0.0554i	-0.1708 - 0.0273i	0.0780 + 0.1506i

The magnitude of these Fourier coefficients are shown in figure 3.14 below:

TABLE 3.2
MAGNITUDE OF FOURIER COEFFICIENTS

1.471	1.523	1.041	1.236	0.691	1.144	0.7	
0.206667462	0.107752401	0.285922385	0.12654754	0.127669926	0.169792255	0.172968003	0
0.33022568	0.113017698	0.320126537	0.167074834	0.078313473	0.177792013	0.21996818	0
0.132530148	0.116962302	0.20406374	0.165649872	0.290812672	0.209040211	0.143175172	
0.069	0.053	0.113	0.098	0.043	0.01	0.046	
0.132530148	0.116962302	0.20406374	0.165649872	0.290812672	0.209040211	0.143175172	
0.33022568	0.113017698	0.320126537	0.167074834	0.078313473	0.177792013	0.21996818	0
0.206667462	0.107752401	0.285922385	0.12654754	0.127669926	0.169792255	0.172968003	0

The phase of the coefficients are shown in the table 3.15 given below:

TABLE 3.3
PHASE OF FOURIER COEFFICIENTS

0	0	0	0	0	0	0	0
0.00223062	-	-	-	-	-	-	-
0.01723219	-0.02919742	0.000490874	0.009990838	0.001566322	0.047818362	0.025277749	-0.08959948
0.019259834	0.009673828	0.839099631	0.008833154	0.007906526	0.006873744	-0.02128162	0.011432405
0	0	0	0	0	0	0	0
0.019259834	0.009673828	0.839099631	0.008833154	0.007906526	0.006873744	0.02128162	0.011432405
-0.01723219	0.02919742	0.000490874	0.009990838	0.001566322	0.047818362	0.025277749	0.08959948
-0.00223062	0.025852839	0.150879829	0.006674285	0.001925504	0.006024449	0.002789673	0.033711042

Using the modulus and the phase of the complex numbers, each of these can be written in the Euler form

$$a + ib = Ae^{i\theta} = A(\cos\theta + i * \sin\theta)$$

The Binary Phase Only Filter for the Fourier coefficients can be obtained by setting up the value 0 if tangent of angle of phase is equal to 0, -1 if it is negative and +1 if it is positive.

Figure 3.5 shows the BPOF filter values.

TABLE 3.4
MAGNITUDE OF FOURIER COEFFICIENTS

0	0	0	0	0	0	0	0
1	-1	-1	-1	-1	1	-1	-1
1	-1	-1	-1	1	-1	-1	-1
1	-1	1	1	-1	1	-1	1
0	0	0	0	0	0	0	0
-1	1	-1	-1	1	-1	1	-1
-1	1	1	1	-1	1	1	1
-1	1	1	1	1	-1	1	1

3.2 PN sequence Generator

Consider the following circuit to generate a PN sequence:

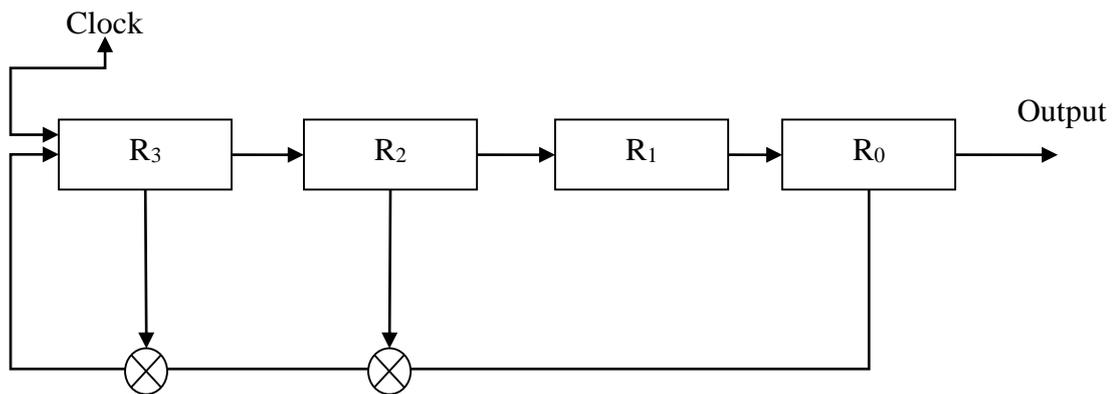


Fig 3.4 LFSR Circuit for generation of PN sequence

R_1 , R_2 , R_3 and R_4 are the registers that hold 1 bit of information. The contents of the R_0 and R_2 are ex-ored and the result is again ex-ored with R_3 and the result is again fed to the R_3 register. The above circuit is an example of Linear Feedback Shift Register (LFSR) Circuit. Let the initial contents of the registers be (0,1,1,0), then the output can be generated in the following way.

TABLE 3.5

LINEAR FEEDBACK SHIFT REGISTER O/P FOR M SEQUENCES

Time	LFSR States	Output
0	0,1,1,0	-
1	1,1,0,1	0
2	1,0,1,0	1
3	0,1,0,0	1
4	1,0,0,0	0
5	0,0,0,1	1
6	0,0,1,1	0

7	0,1,1,0	0
---	---------	---

The last row is the same as the first one and therefore indicates that the same pattern would repeat thereafter. Thus, the period of the LFSR consisting of 3 registers is $2^3-1 = 7$.

Denoting 0 with -1 to keep the values in polar format, the PN sequence obtained is -1,1,1,-1,1,-1,-1.

Consider the watermark to be bit sequence 101. Converting it into polar form, the watermark can be represented as 1 -1 1. The speeded sequence obtained to embed in the host signal is:

TABLE 3.6
EMBEDDING OF BITS CORRESPONDING TO PSNR VALUES

Watermark Bit	PN Sequence	Spread Sequence
1	-1	-1
	1	1
	1	1
	-1	-1
	1	1
	-1	-1
	-1	-1
-1	-1	1
	1	-1
	1	-1
	-1	1
	1	-1
	-1	1
	-1	1
1	-1	-1
	1	1
	1	1
	-1	-1
	1	1
	-1	-1
	-1	-1

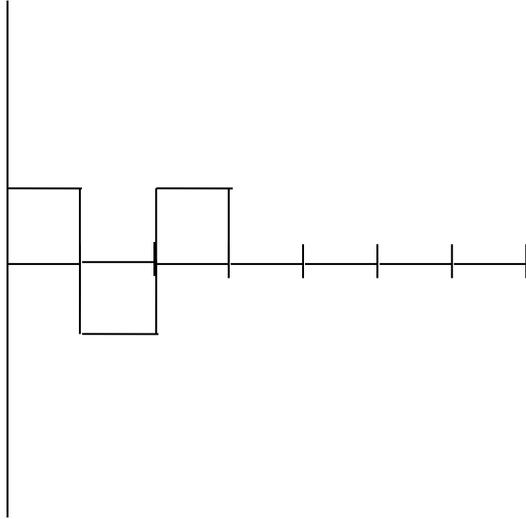


Fig 3.5 The watermark to be embedded (represented as a signal). Horizontal axes shows the time and the vertical axes shows the amplitude.

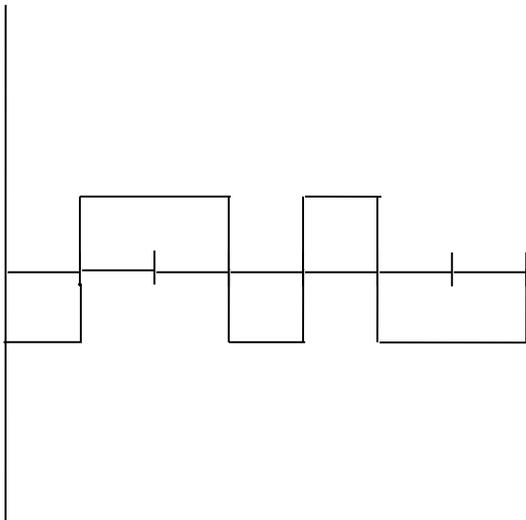


Fig 3.5 The spreading corresponding to a one bit data. Horizontal axes shows the time and the vertical axes shows the amplitude.

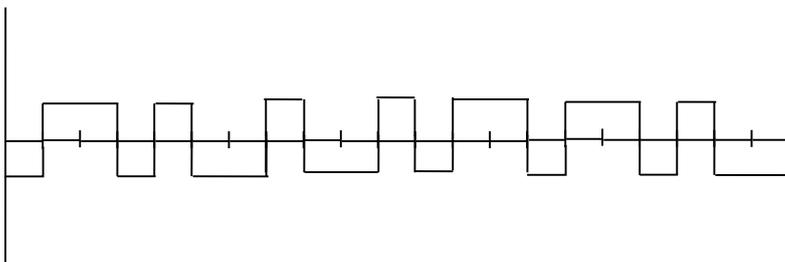


Fig 3.7 The spreading corresponding to a sequence 101. Horizontal axes shows the time and the vertical axes shows the amplitude.

Thus, the sequence to be embed is -1,1,1,-1,1,-1,-1,1,-1,-1,1,-1,1,1,-1,1,-1,-1. These watermark bits are to be added in such a way so as to maintain the symmetry of the Fourier transform.

3.3 Proposed Algorithm for watermark embedding

The data to be embedded is used to modulate the magnitude of the Fourier coefficients in such a way so as to maintain the symmetry of the magnitudes. In the example shown above, the hypothetical frame consists of 64 pixels (8X8 matrix) and the watermark is to be embedded in all the bits of the frame, such that the symmetry is preserved. It can be easily observed that the first row of the transform consists of all the real values, whereas the other seven values of each row are symmetrically separated. Thus, keeping the two rows, the first one and the middle, one can have a total of 48 pixels in which the values are to be modified. Also, as the symmetry is to be preserved, a total of 24 bits can be embedded in the magnitude of the coefficients. The spreaded code to be embed is to be appended with padding bits to make it 24 bits long.

Let α be the masking threshold value above which denotes a 1 and below which denotes a 0. This value is suitably chosen as per the magnitude of the Fourier coefficients. Consider the table for magnitude of Fourier coefficients repeated here for ready reference.

1.471	1.523	1.041	1.236	0.691	1.144	0.7	
0.206667462	0.107752401	0.285922385	0.12654754	0.127669926	0.169792255	0.172968003	0.
0.33022568	0.113017698	0.320126537	0.167074834	0.078313473	0.177792013	0.21996818	0.
0.132530148	0.116962302	0.20406374	0.165649872	0.290812672	0.209040211	0.143175172	
0.069	0.053	0.113	0.098	0.043	0.01	0.046	
0.132530148	0.116962302	0.20406374	0.165649872	0.290812672	0.209040211	0.143175172	
0.33022568	0.113017698	0.320126537	0.167074834	0.078313473	0.177792013	0.21996818	0.
0.206667462	0.107752401	0.285922385	0.12654754	0.127669926	0.169792255	0.172968003	0.

The two rows which are highlighted are to be kept intact and the embedding is performed in half of the remaining bits, as the other half is to be modified to maintain the symmetry of Fourier Transform.

Let $\alpha = .2$, which indicates that a value equal to or below 0.2 denotes a -1 and a value above it denotes a +1. An integer multiple of a step size β is to be added (or subtracted) to (from) the magnitude to change it to denote the value it should represent.

Consider the data stream -1,1,1,-1,1, -1, -1, 1,-1,-1,1,-1,1,1,-1,1,1,-1,1,-1,-1,±1,±1,±1. The last three are padding bits which can either be +1 or -1.

One possible embedding with $\beta= 0.1$ is

TABLE 3.7

EMBEDDING IN MAGNITUDE VALUES OF FFT COEFFICIENTS (CHANGED VALUES ARE SHOWN IN RED)

1.471	1.523	1.041	1.236	0.691	1.144	0.7
0.106667462	0.207752401	0.285922385	0.12654754	0.227669926	0.169792255	0.172968003
0.13022568	0.113017698	0.320126537	0.167074834	0.278313473	0.277792013	0.11996818
0.232530148	0.116962302	0.20406374	0.165649872	0.190812672	0.209040211	0.143175172
0.069	0.053	0.113	0.098	0.043	0.01	0.046
0.232530148	0.116962302	0.20406374	0.165649872	0.190812672	0.209040211	0.143175172
0.13022568	0.113017698	0.320126537	0.167074834	0.278313473	0.277792013	0.11996818
0.106667462	0.207752401	0.285922385	0.12654754	0.227669926	0.169792255	0.172968003

The Fourier coefficients can be constructed from these modified magnitudes and the original phase of the frame, using the Euler Formula.

TABLE 3.8

RECONSTRUCTION OF FOURIER COEFFICIENTS FROM MODIFIED MAGNITUDE AND PHASE VALUES

1.471	1.523	1.041	1.236	0.691	1.144	0.7	0.991
-	-	-	-	-	-	-	-
0.105806833 160219- 0.013522629 4087695i	0.11626162 963945+0.1 7217518286 572i	0.03310000 00494138+ 0.28400000 0423974i	0.1181999 99557336 +0.045199 99983072 4i	0.22629694 0233946+0. 0249657774 883786i	0.160499999604 685+0.0553999 998635486i	0.170800000176 429+0.0273000 000281998i	0.123990406324152- 0.239396861441247i
-	-	-	-	-	-	-	-
0.092673091 815026- 0.091490031 0684512i	0.05800000 0138993+0. 0970000002 324537i	0.31999999 9518525+0. 008999999 8645852i	0.1450000 00405504 +0.083000 00023211 63i	0.27719944 1638467- 0.02487687 29675547i	0.095309752539 9101- 0.260929978265 i	0.068173599167 5316+0.098715 3715945857i	0.0441690512392877- 0.226145542345153i
-	-	-	-	-	-	-	-
0.156154532 083823- 0.172296348 883499i	0.10230000 033329- 0.05670000 01847268i	0.00509999 999894598 +0.2039999 99957839i	0.1478000 00252124- 0.0748000 00127597 3i	0.17381043 4475517- 0.07873632 36582182i	0.194499999588 112- 0.076599999837 7861i	0.090799999728 0969- 0.110699999668 506i	-0.119999999650228- 0.0785999997708992i
-0.069	0.053	-0.113	0.098	-0.043	0.01	0.046	0.059
-	-	-	-	-	-	-	-
0.156154532 083823+0.17 22963488834 99i	0.10230000 033329+0.0 5670000018 47268i	0.00509999 999894598- 0.20399999 99957839i	0.1478000 00252124 +0.074800 00012759 73i	0.17381043 4475517+0. 0787363236 582182i	0.194499999588 112+0.0765999 998377861i	0.090799999728 0969+0.110699 999668506i	- 0.119999999650228+ 0.0785999997708992i
-	-	-	-	-	-	-	-
0.092673091 815026+0.09 14900310684 512i	0.05800000 0138993- 0.09700000 02324537i	0.31999999 9518525- 0.00899999 998645852i	0.1450000 00405504- 0.0830000 00232116 3i	0.27719944 1638467+0. 0248768729 675547i	0.095309752539 9101+0.260929 978265i	0.068173599167 5316- 0.098715371594 5857i	0.0441690512392877 +0.226145542345153i

- 0.105806833 160219+0.01 35226294087 695i	- 0.11626162 963945- 0.17217518 286572i	- 0.03310000 00494138- 0.28400000 0423974i	- 0.1181999 99557336- 0.0451999 99830724i	- 0.22629694 0233946- 0.02496577 74883786i	0.160499999604 685- 0.055399999863 5486i	- 0.170800000176 429- 0.027300000028 1998i	0.123990406324152+ 0.239396861441247i
--	---	--	---	--	---	--	--

which on inverse Fourier Transform gives

TABLE 3.9
RECONSTRUCTION OF PIXEL VALUES FROM MODIFIED FOURIER
COEFFICIENTS

86.5914	179.0096	29.0000	64.0000	100.4215	159.5774	56.2066	143.2899
257.1213	100.4501	48.9801	131.9652	36.7449	273.4859	25.5695	272.3826
158.7248	154.2812	176.0000	173.0000	124.3743	87.4226	75.7934	160.4070
193.5757	226.2233	66.9858	163.0000	165.7658	17.5094	167.4168	73.0462
217.5721	185.9904	43.0000	197.0000	24.8218	176.5774	96.2066	141.2947
173.6237	218.5499	235.0199	111.0348	159.1935	140.4791	88.5728	73.6901
238.1117	268.7188	216.0000	233.0000	176.2254	153.4226	144.7934	80.0085
145.6793	189.7767	226.0142	163.0000	5.2957	135.5256	45.4409	46.8810

The original blue plane pixel values are shown in the table 3.10.

TABLE 3.10
ORIGINAL PIXEL VALUE OF THE BLUE PLANE OF IMAGE

43	193	29	64	96	151	42	127
252	125	49	132	43	250	5	224
216	175	176	173	67	96	90	143
177	231	67	163	178	41	188	90
190	172	43	197	27	168	82	148
249	194	235	111	144	117	68	73
252	248	216	233	134	162	159	107
92	185	226	163	2	159	66	79

The MSE for the above hypothetical video frame is 620 giving PSNR value 20.20db.

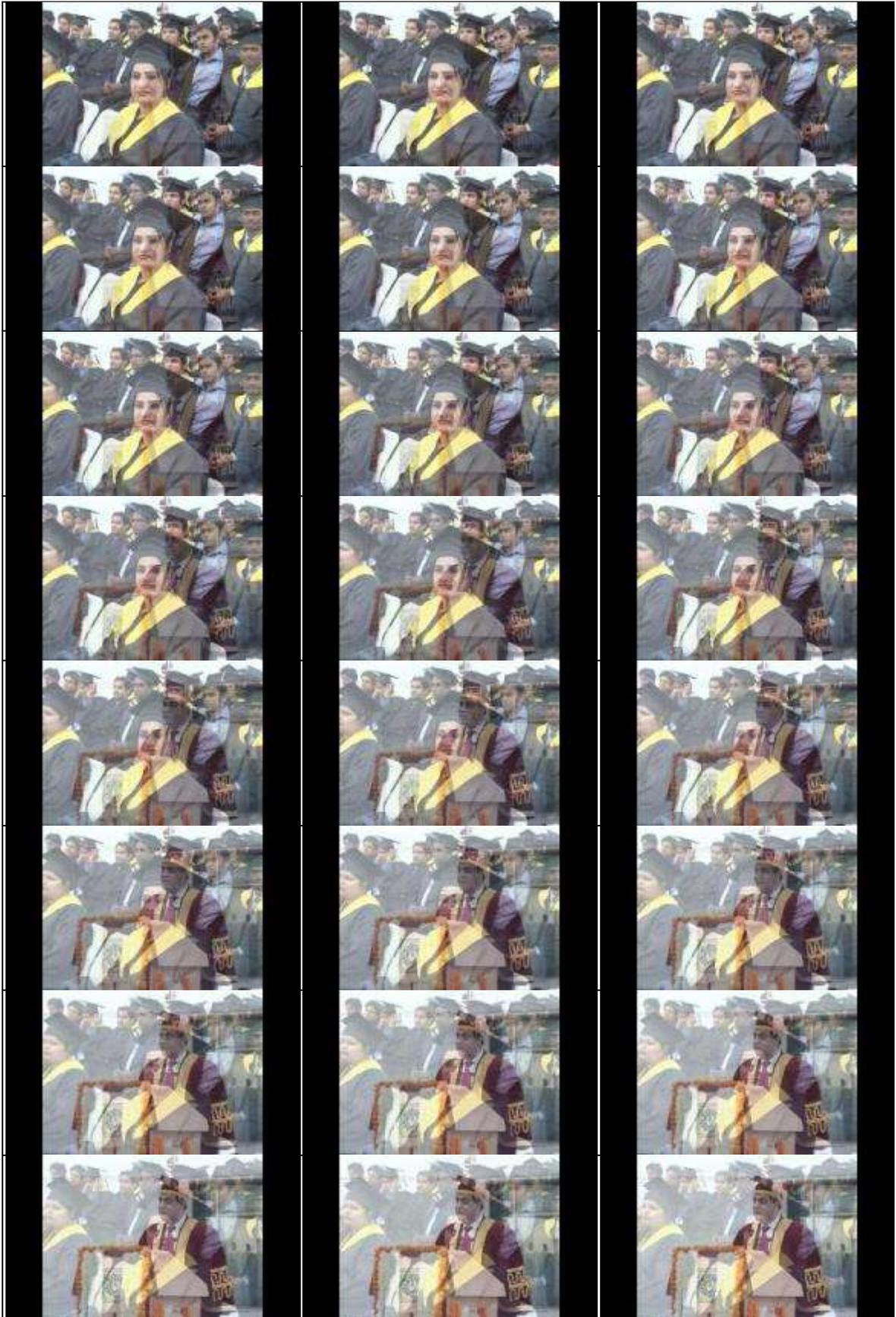
The PSNR value for Suresh GyanVihar University convocation video (also provided in CD ROM enclosed) is computed in Chapter 4.

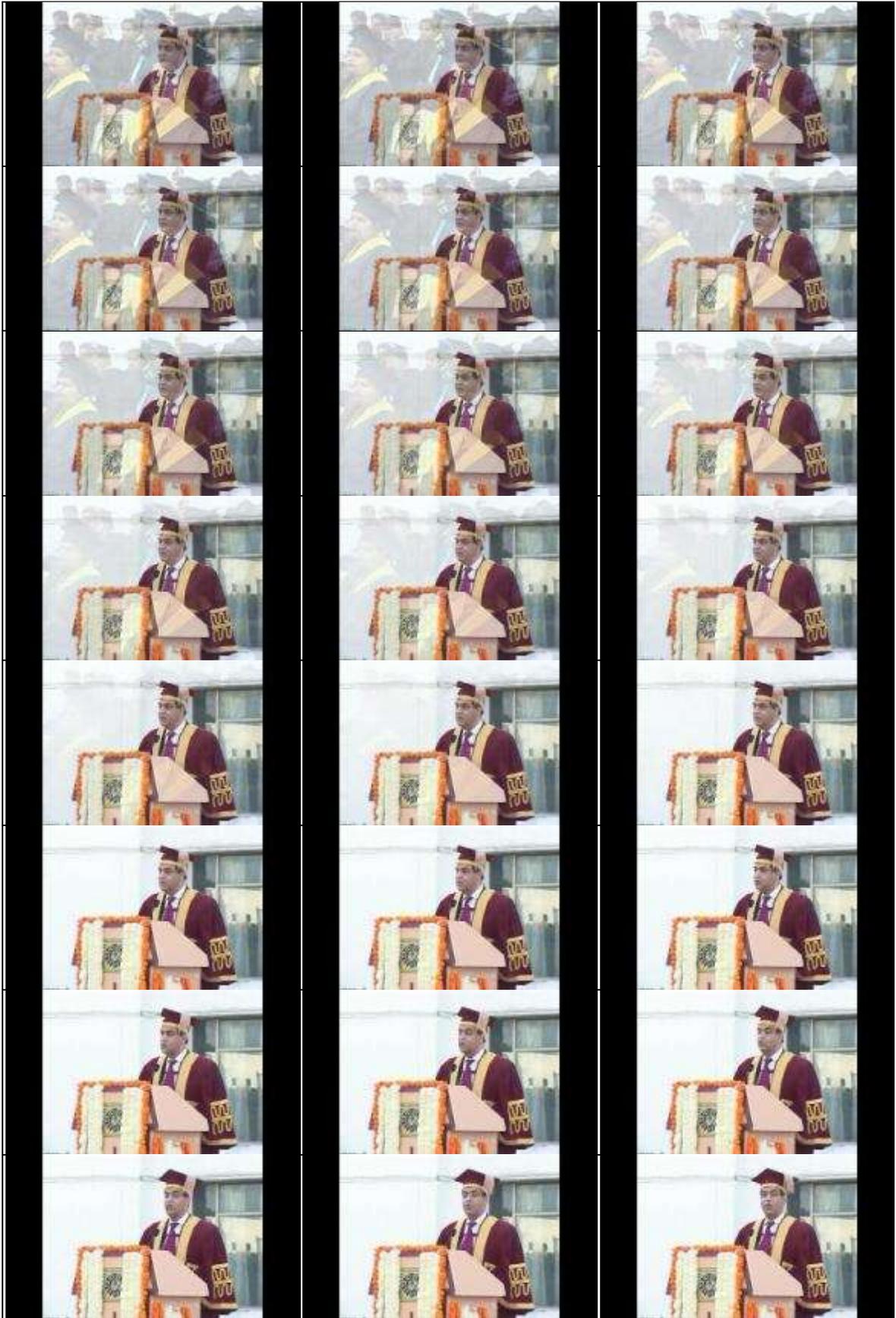
CHAPTER 4

ANALYSIS OF PROPOSED WORK

4.1 Video Watermarking in FFT domain

The sample video of convocation at Suresh GyanVihar University campus is partitioned into frames using MATLAB. The total length of file is 10 seconds with a frame rate of 28 fps giving a total of 280 frames illustrated as shown:





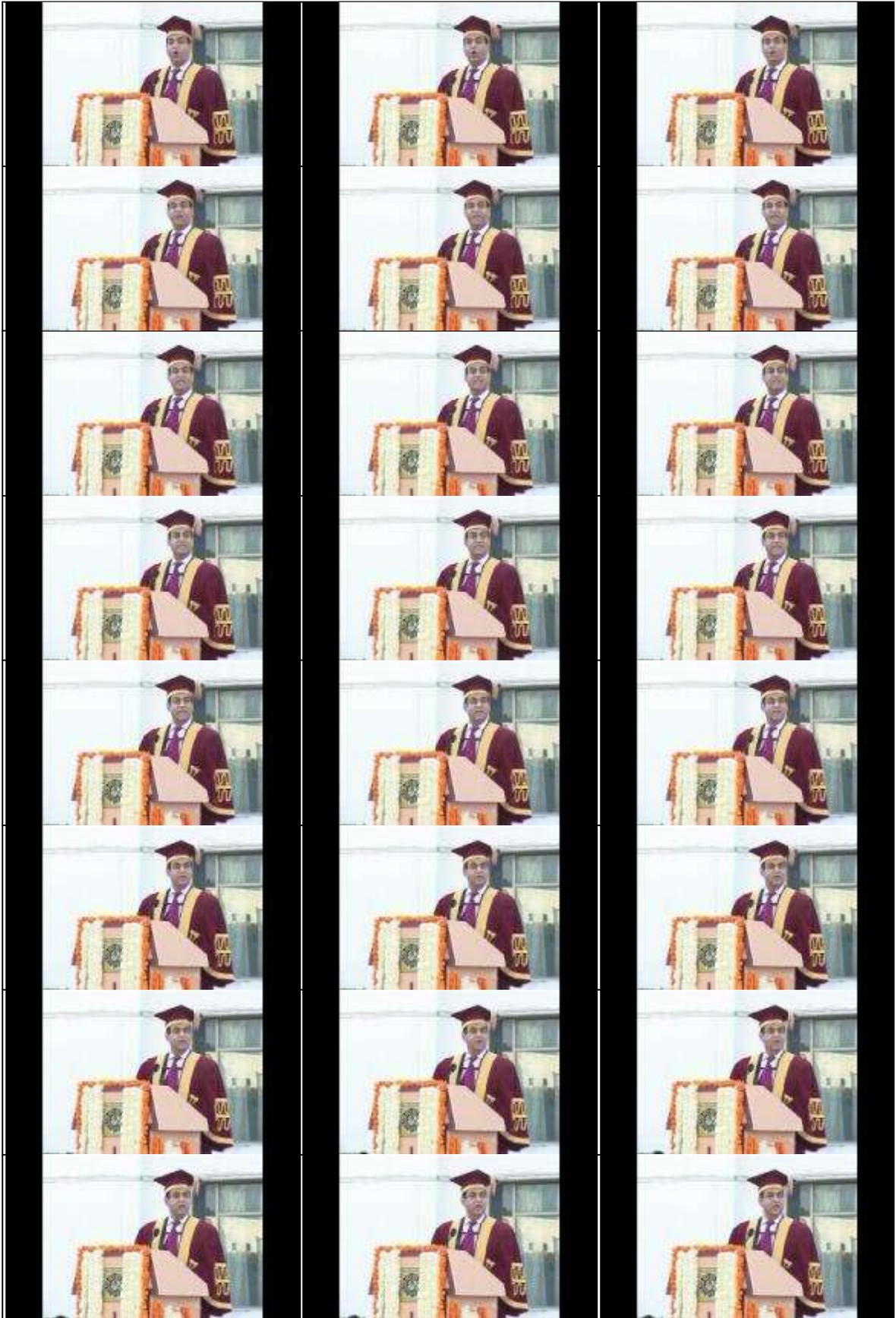




Figure 4.1 Illustration of Frames of Convocation Video of Suresh GyanVihar Universe [Few Frames are shown as the size of data is large]

Consider the digital watermark as the text "Hello World".

The text is to be converted into binary ASCII code before embedding into the video frame. Moreover, the ASCII code is then modulated with PN sequence to obtain the modulated code which is then embedded into the magnitude of the Fourier coefficients. This process is illustrated as shown:

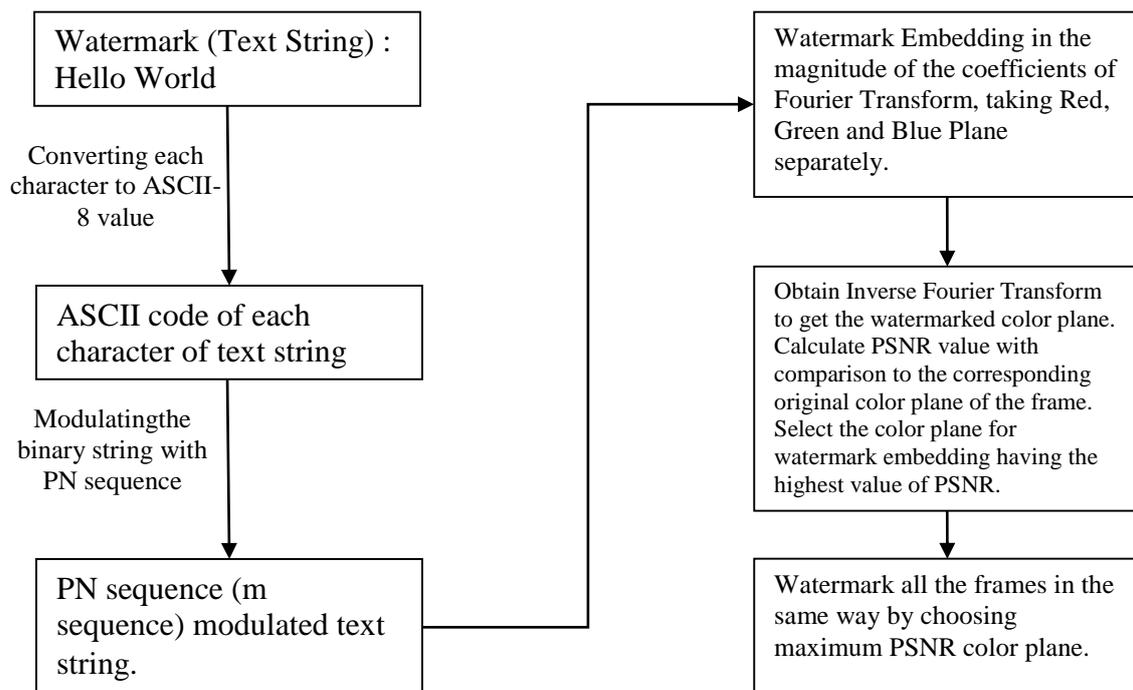


Fig. 4.2 Watermark Embedding Process

For all the 280 frames of the convocation video, and watermarking with the text string Hello World, the following table illustrates the procedure.

TABLE 4.1

ILLUSTRATION OF CONVERSION INTO ASCII CODE OF THE WATERMARK TEXT

Watermark	ASCII-8	Binary (8 Bits)
H	104	01101000
E	101	01100101
L	108	01101100
L	108	01101100
O	111	01101111
[space]	32	00100000
w	119	01110111
o	111	01101111
r	114	01110010
l	108	01101100
d	100	01100100

All these binary values are concatenated to get the watermark sting in binary form. Each 1 is replaced by -1 and 0 is replace by 1 to get the string in the polar variables. To modulate the string using PN sequence, each 1 is replaced by the entire PN sequence and each -1 is replaced by negation of the sequence (sequence with sign of each element of the string reversed.)

The modulated string thus obtained is embedded in the magnitude of Fourier coefficients in the following way.

- Number of characters in the Watermark String : 11
- Number of characters in Binary Watermark string : $11 * 8 = 88$ (ASCII 8 bits)
- Length of PN (m sequence) code : 7 ($=2^3 - 1$)
- Length of modulated sequence : $88 * 7 = 616$ bits
- Frame dimensions (in pixels) : $100 * 100 = 10,000$

Complex Coefficients : 10000-200 = 9800

*Coefficients available for embedding : 9800/2 = 4900

The last point indicates that only half of the Fourier coefficients are used for watermark bit embedding as the other half is to be modified accordingly in view as to keep the symmetry of the Fourier Transform.

The rest of the places are filled with padding bits or can be left unchanged but will provide garbage on extraction.

For a $n \times n$ dimension frame video, the number of characters that can be embedded in the frame is;

$$\# \text{ characters} = \frac{(n * n - 2n)}{(2^{k-1}) * 8 * 2}$$

Later, at the detection and extraction end, the watermark is extracted using correlation analysis, with the given m sequence.

TABLE 4.2

PLANES SELECTED FOR WATERMARK EMBEDDING BASED ON PSNR VALUES

# frame	plane	MSE	PSNR
1	RED	660	19.96936
2	GREEN	671	19.89757
3	RED	667	19.92354
4	RED	689	19.78261
5	GREEN	673	19.88465
6	BLUE	610	20.3115
7	BLUE	645	20.0692
8	RED	644	20.07594
9	BLUE	682	19.82696
10	RED	610	20.3115
11	GREEN	641	20.09622
12	BLUE	647	20.05576
13	RED	684	19.81424
14	BLUE	678	19.8525

15	BLUE	638	20.11659
16	RED	650	20.03567
17	GREEN	638	20.11659
18	BLUE	621	20.23388
19	RED	646	20.06247
20	RED	679	19.8461
21	BLUE	678	19.8525

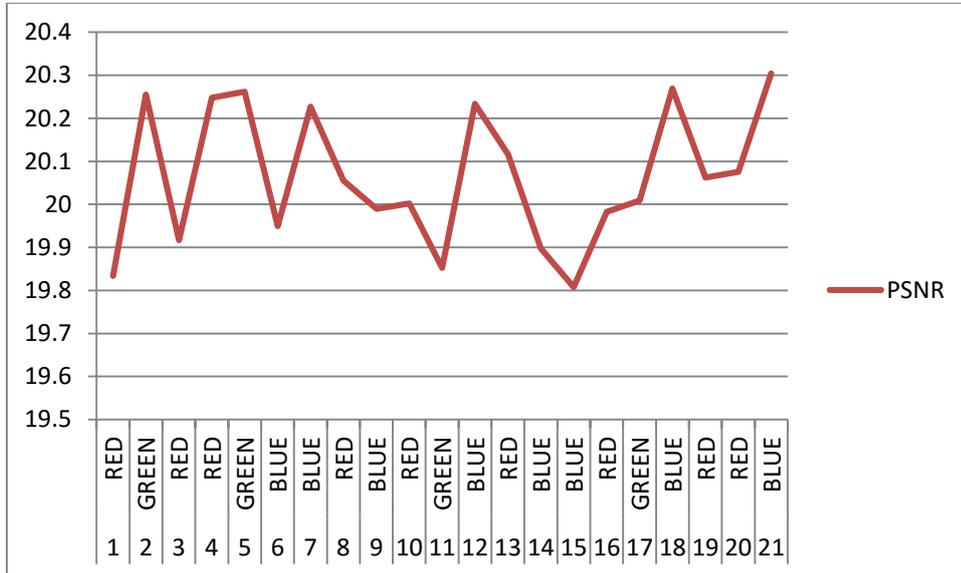


Fig 4.3 PSNR values corresponding to different Frames.

TABLE 4.3

PSNR VALUES AS A FUNCTION OF LENGTH OF WATERMARK MESSAGE

length of Characters in Message	MSE	PSNR
1	164	86.31155
2	214	82.4724
3	265	79.38858
4	326	76.39979
5	373	74.45675
6	401	73.41248
7	475	70.96923
8	501	70.2004
9	590	67.84136
10	625	67.00994
11	669	66.02844

12	741	64.55377
13	750	64.3796
14	817	63.14515
15	876	62.1392
16	949	60.98443
17	964	60.75818
18	1017	59.98603
19	1060	59.38858
20	1134	58.41502

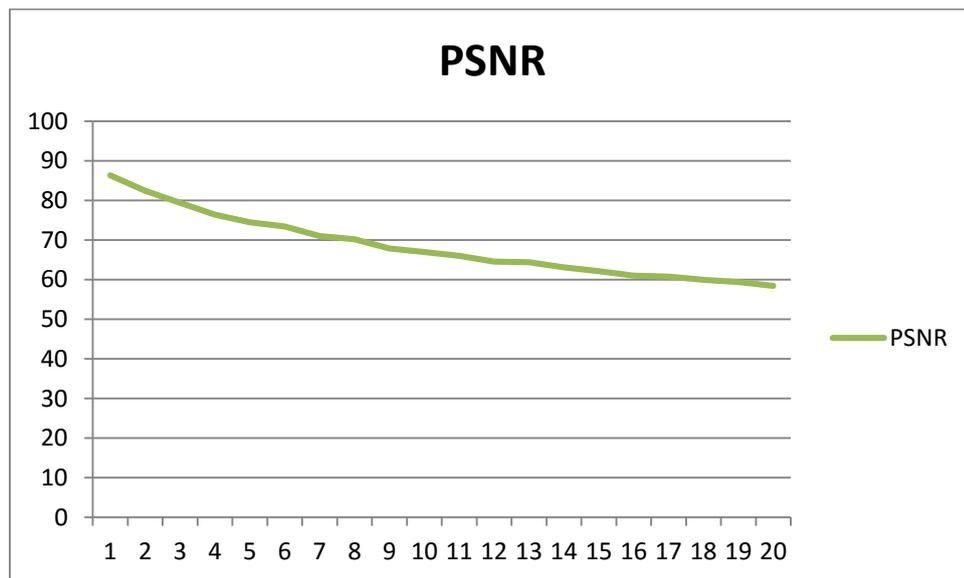


Fig 4.4 Variation in PSNR (db) with number of symbols

The above chart clearly indicates that as the length of message increases, the Peak Signal to Noise Ratio decreases. This is accordance with the fact that Mean Square Error increases as the number of bits to be embedded increases which in turn, lowers the value of PSNR.

4.2 PSNR values for Normalized Comparison

The PSNR values as function of length of message is computed in section 4.3. The length of the spreaded message to be embedded in the video frame depends upon the length of the spreading code. The length of the m sequence is of the order of $2^n - 1$ where $n \in \mathbb{I}$. Thus, the length of the m sequence can be 3,7,15,31 etc. for n being 2,3,4,5 etc. correspondingly. The larger the length of the spreaded signal, the more is the Mean Square Error and the less is the PSNR.

However, m sequence of large length gives a greater robustness against distortion attacks as compared to the code of small length.

The values of PSNR as function of code length is derived for sample string "Hello World" and GyanVihar University Convocation video and is illustrated as shown:

TABLE 4.4

Length of m sequence and PSNR values for mp4 format video

PSNR	30.7149993	28.6314616	26.4651083	26.4651083	21.4866398	18.2315835	13.8742517
R	217719	235304	185912	185912	524944	901287	551385
M Sequence Length	1	3	7	15	31	63	127

The plot corresponding to Table 4.4 is shown in figure 4.5.

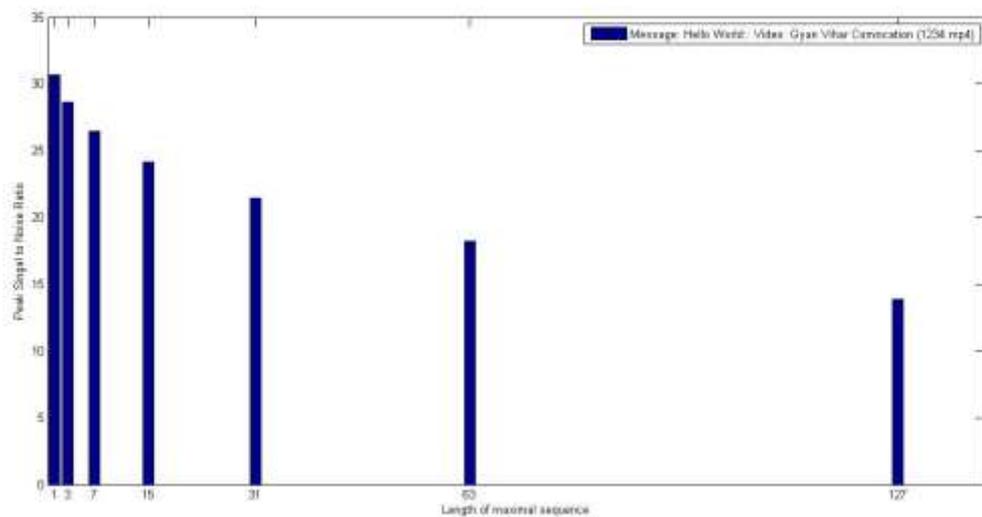


Fig. 4.5 PSNR values as function of length of m sequences

TABLE 4.5

Length of m sequence and PSNR values for wmv format video

PSNR	30.7149993 217719	26.6314616 235304	22.4651083 185912	18.1352516 159957	13.4866398 524944	8.23158359 012865	1.87425175 513852
M Sequence Length	1	3	7	15	31	63	127

The plot corresponding to Table 4.5 is shown in figure 4.6.

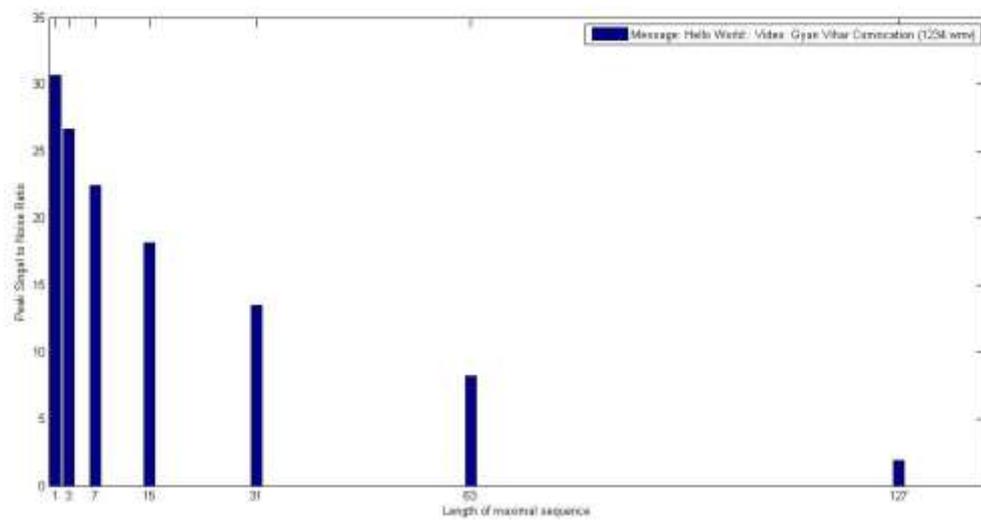


Fig. 4.6 PSNR values as function of length of m sequences

TABLE 4.6

Length of m sequence and PSNR values for avi format video

PSNR	30.71499 93217719	27.63146 16235304	24.46510 83185912	21.13525 16159957	17.48663 98524944	13.23158 35901287	7.874251 75513852
M Sequence Length	1	3	7	15	31	63	127

The plot corresponding to Table 4.6 is shown in figure 4.7.

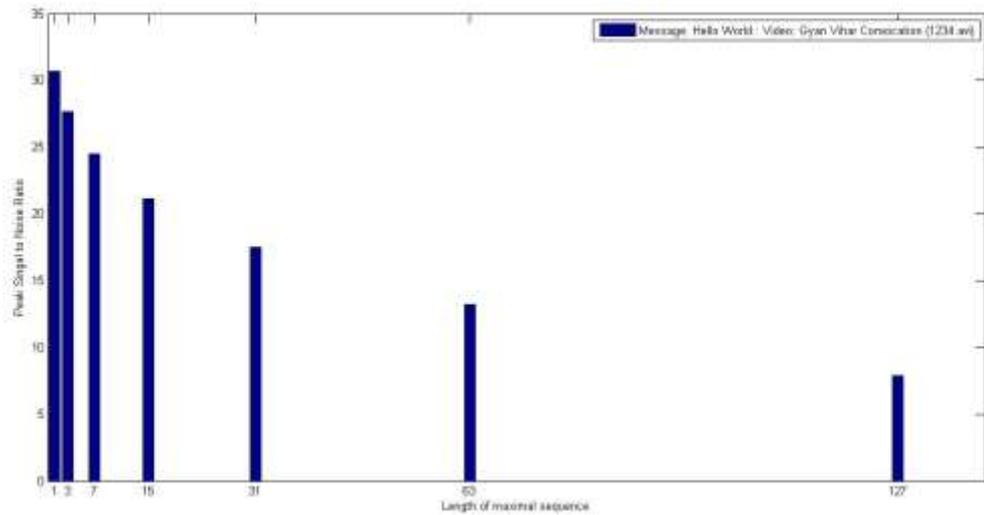


Fig. 4.7 PSNR values as function of length of m sequences

The comparative analysis for all the three video formats is shown in figure 4.8.

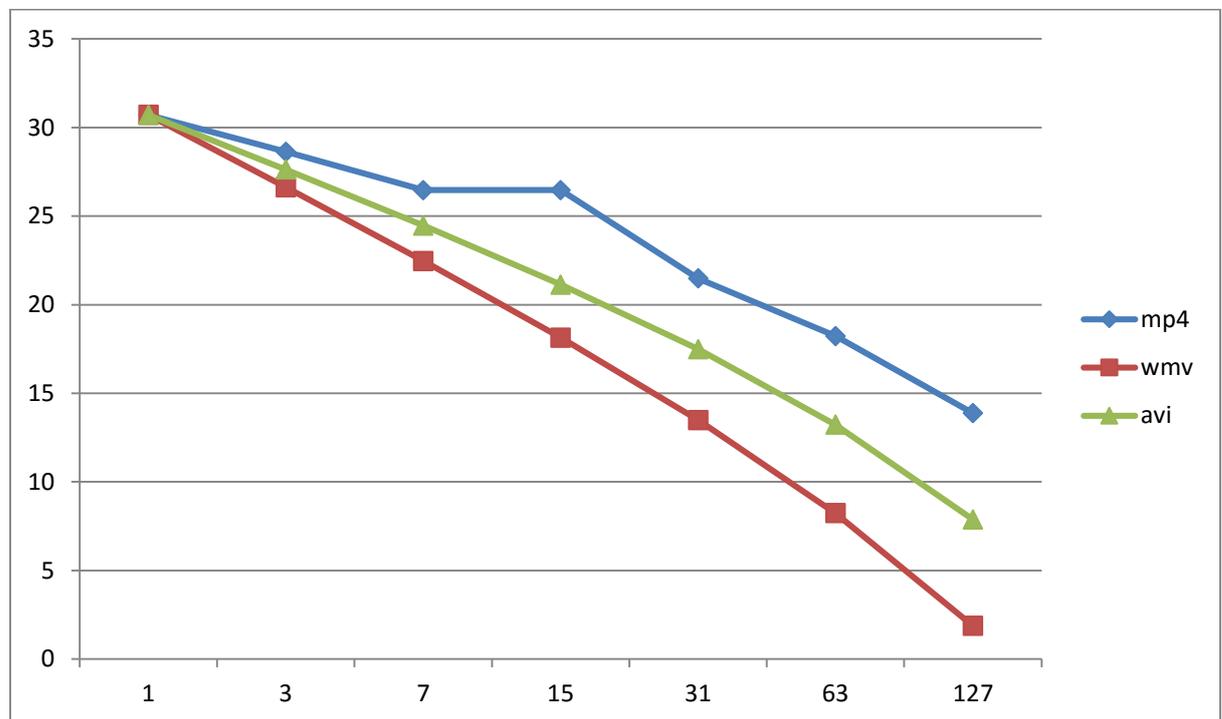


Fig.4.8. Comparative analysis of PSNR for all the three video formats: mp4, wmv and avi.

The above plot clearly indicates that a higher values of PSNR can be achieved if a maximal sequence of small length is used. However, for a small length m sequence, correlation values cannot be computed efficiently in case of attacks and thus gives poor robustness.

CONCLUSION AND FUTURE SCOPE**5.1 Conclusion**

This dissertation proposes a technique of watermark embedding in video in frequency domain in which the video is segregated into frames. Each frame is then converted to image format and then separated into Red, Green and Blue Planes. The pixel matrix of each of the color planes is then subjected to two dimensional Fourier Transform. The text string to be used as watermark is first converted into ASCII code, and then modulated using PN sequence. The Fourier Coefficients are then converted into Euler Form by decomposing into magnitude and phase. This modulated string is then embedded into the magnitude of the Fourier Coefficients without affecting the phase.

To keep the fidelity of the watermark video as high as possible, the proposed technique embed the watermark in only one of the three planes, viz red, green and blue, which gives the highest value of PSNR.

Later, at the time of detection, the extracted string is to be demodulated with the same PN sequence to get back the watermark, or the presence of watermark is ensured. It turns out that the PSNR value is a function of length of the watermark. PSNR values are inversely proportional to the length of the watermark message.

The proposed technique for video watermarking presents a robust technique against frame dropping attacks. As the different frames of the video, consists of watermark in any of red, green or blue planes, the watermark presence is unaltered against frame dropping. However, an additional computation cost is inculcated as the detector has to decompose the frame into RGB planes and each one is separately operated for the presence of watermark.

5.2 Future Scope

The proposed future work on this technique of video watermarking embeds the watermark selectively in the transitional frames of the video. Transitional frames are the frames in which the location of a sequence of frames changes. These are the frames in which some content of the new scene is superimposed on some content of the previous scene. However, in such a technique, there is a serious drawback of robustness of the embedded watermark. However, embedding selectively in the transitional frames and some intermediate frames will serve the purpose of robustness and at the same time, optimizing the fidelity of the watermarked video.

