

**AN EFFICIENT DCT-BASED VIDEO  
WATERMARKING FOR COPYRIGHT CONTROL**

**MAY THARAPHY HTUN**

**M.C.Tech.**

**JUNE, 2022**

**AN EFFICIENT DCT-BASED VIDEO WATERMARKING  
FOR COPYRIGHT CONTROL**

**By  
MAY THARAPHY HTUN  
B.C.Tech.**

**A Dissertation Submitted in Partial Fulfilment of the  
Requirements for the Degree of  
Master of Computer Technology  
(M.C.Tech.)**

**University of Computer Studies, Yangon  
JUNE, 2022**

## ACKNOWLEDGEMENTS

I would first like to thank Prof. Dr. Mie Mie Khin, Rector of the University of Computer Studies, Yangon for kindly giving me an opportunity to work on this thesis.

I would also like to express my gratitude to Dr. Htar Htar Lwin, Pro Rector and Head of the Faculty of Computer Systems and Technologies, University of Computer Studies, Yangon. She always helps me whenever I ran into a trouble spot or had a question about my research or writing.

I would also like to thank Dr. Amy Tun, Professor and Course-coordinator of the Master (thesis) course of the University of Computer Studies, Yangon. Without her advice and support, this thesis could not have been successfully implemented.

I would like to express my special thanks to my supervisor, Dr. Htar Htar Lwin, Pro Rector and Head of the Faculty of Computer Systems and Technologies, University of Computer Studies, Yangon. She consistently steered me in the right direction, and also gave me motivation and guidance whenever she thought I needed it.

I would also like to give my appreciation to Daw Aye Aye Khine, Associate Professor and Head, Department of English, University of Computer Studies, Yangon, for editing my thesis on the language point of view.

I am very grateful to all of my teachers from the University of Computer Studies, Yangon, who had been helping me from beginning to end of my thesis. I really appreciate for their valuable comments, suggestions, helpful hints, and fullest cooperation during the seminars of my thesis.

Finally, I must express my very profound gratitude to my parents and to my friends for providing me with unfailing support and continuous encouragement throughout my years of study and through the process of researching and writing this thesis. This accomplishment would not have been possible without them.

## ABSTRACT

Since the technology is being improved, digital media are facing challenges like copyright infringements. Digital data can be easily created, copied, processed, and distributed freely among unauthorized users. The author or owner of the data does not know that the duplicate files of his work are available on the Internet that can be accessed by anybody. The copyright laws are also not sufficient to deal with the digital data.

With the aim of copyright control in transmission of digital video, this thesis presents a Discrete Cosine Transform (DCT)-based video watermarking method. Its essence is to embed the copyright related information on digital videos in such a way that it can later be extracted in case copyright violation is detected.

Watermark used in copyright protection applications is information that can uniquely identify the owner, such as logo, signature, etc. For copyright protection, the watermark must be carefully embedded in the host video. A good watermarking system should not degrade the visual quality of the host video. It means that watermarks used in the copyright protection applications should not be perceptible by human eyes. It should also resist common signal processing attacks. Only when the copyright violation is detected, it should be able to successfully extract the watermark from the manipulated video to prove the ownership.

In this system, watermark is embedded by only changing the luminance of the video frames. As the human visual system cannot easily detect light intensity changes in the images, the method presented in this thesis well preserves the visual quality of the watermarked videos by keeping the Peak Signal to Noise Ratio (PSNR) of more than 50dB. Moreover, this system solves the frame drop problem by repetitively embedding the watermark in all video frames. Therefore, even if some of the video frames were lost during transmission, other remaining frames can successfully be used for watermark detection.

The system also provides remarkable robustness against compression attack as it is based on the Discrete Cosine Transform (DCT), which is a proven method used in Joint Photographic Experts Group (JPEG) compression.

In addition to compression attacks, the robustness of the system is also tested against various kinds of signal processing attacks: such as compression, rotation, cropping, quantization, filtering, and noise addition. According to the experimental

results, the system shows a great ability to preserve the watermark against those attacks.

# TABLE OF CONTENTS

<b>ACKNOWLEDGEMENTS</b> .....	<b>i</b>
<b>ABSTRACT</b> .....	<b>ii</b>
<b>TABLE OF CONTENTS</b> .....	<b>iv</b>
<b>LIST OF FIGURES</b> .....	<b>vi</b>
<b>LIST OF TABLES</b> .....	<b>vii</b>
<b>LIST OF EQUATIONS</b> .....	<b>viii</b>
<b>CHAPTER 1 INTRODUCTION</b> .....	<b>1</b>
1.1    Data Encryption (or) Cryptography.....	2
1.2    Data Hiding and Watermarking.....	3
1.3    Application Areas of Digital Watermarking Techniques	6
1.4    Objectives of the Thesis .....	6
1.5    Motivation of the Thesis.....	6
1.6    Organization of the Thesis.....	7
<b>CHAPTER 2 BACKGROUND THEORY</b> .....	<b>8</b>
2.1    Physical and Digital Watermarks .....	8
2.2    Classification of Watermarks .....	9
2.2.1 Visibility of Watermark .....	9
2.2.2 Robustness of Watermark.....	9
2.2.3 Blind and Non-Blind Watermark.....	10
2.2.4 Host Media and Watermark Data Types.....	11
2.3    Different Domains of Embedding Watermark .....	11
2.3.1 Digital Video Watermarking in Spatial Domain .....	11
2.3.2 Digital Video Watermarking in Frequency Domain.....	12
2.3.3 Digital Video Watermarking in Discrete Cosine Transform Domain.....	13
2.4    Background on Video .....	14
2.4.1 Structure of a Video .....	15
2.4.2 Color Models .....	16

2.4.3 Video Encoding Format.....	17
<b>CHAPTER 3 DCT-BASED VIDEO WATERMARKING METHOD .....</b>	<b>19</b>
3.1 Selection of System Criteria.....	19
3.2 Watermark Embedding Process .....	21
3.3 Watermark Extraction Process .....	27
<b>CHAPTER 4 EXPERIMENTAL RESULTS .....</b>	<b>29</b>
4.1 Media used for Experiments .....	29
4.2 Evaluation on Watermark Invisibility .....	30
4.2.1 Mean Square Error (MSE).....	30
4.2.2 Peak Signal to Noise Ratio (PSNR) .....	30
4.2.3 Mean Opinion Score (MOS) .....	34
4.3 Evaluation on Watermark Similarity .....	35
4.3.1 Attacks on Watermarking Methods .....	37
<b>CHAPTER 5 CONCLUSION .....</b>	<b>42</b>
5.1 Discussion .....	42
5.1 Further Extension.....	43
<b>REFERENCES.....</b>	<b>45</b>
<b>PUBLICATION .....</b>	<b>50</b>

## LIST OF FIGURES

<b>FIGURE</b>	<b>DESCRIPTION</b>	<b>PAGE</b>
1.1	Types of Steganography	4
2.1	Physical Watermark	8
2.2	Structure of a Video	15
2.3	Primary and Secondary Colors of RGB Model	17
3.1	Flow of the Proposed Watermark Embedding Process	21
3.2	RGB to YCbCr Conversion	22
3.3	Frame Structure after Dividing into 8x8 Blocks	24
3.4	YCbCr to RGB Calculation Result	26
3.5	Flow of the Proposed Watermark Extraction Process	28
4.1	Watermark Images: M logo (left) and Lena (right)	30
4.2	Original (left) and Watermarked (right) Video Frames for Nature1.mp4	31
4.3	Original (left) and Watermarked (right) Video Frames for Cartoon1.mp4	32
4.4	Original (left) and Watermarked (right) Video Frames for Music1.mp4	32
4.5	Extracted Logos after Compression Attack	38
4.6	Extracted Logos after Cropping Attack	38
4.7	Extracted Logos after Rotation Attack	38
4.8	Extracted Logos after Quantization Attack	39
4.9	Extracted Logos after Filtering Attack	39
4.10	Extracted Logos after Adding Noise Attack	40



## LIST OF TABLES

<b>TABLE</b>	<b>DESCRIPTION</b>	<b>PAGE</b>
3.1	Different Characteristics of Watermarking Systems	20
4.1	Videos in MP4 Format Used for Experiments	29
4.2	PSNR after Watermark Embedding with Different Alpha Value	31
4.3	PSNR Results for 30 Different Videos with $\alpha = 5$	33
4.4	Mean Opinion Score (MOS) Ranking	34
4.5	MOS Ranking for 3 Different Video with $\alpha = 5$	35
4.6	Similarity Results for Extracted Watermarks from 30 Different Videos (Before Attack)	36
4.7	Average Similarity Scores for Various Attacks	40

## LIST OF EQUATIONS

<b>EQUATION</b>	<b>DESCRIPTION</b>	<b>PAGE</b>
2.1	2D DCT-II Transformation	14
2.2	Inverse DCT Transformation (IDCT)	14
2.3	YCbCr Conversion from Gamma Corrected Value (Y')	17
2.4	YCbCr Conversion from Gamma Corrected Value (Cb')	17
2.5	YCbCr Conversion from Gamma Corrected Value (Cr')	17
2.6	RGB Conversion from Gamma Corrected Value (R')	17
2.7	RGB Conversion from Gamma Corrected Value (G')	17
2.8	RGB Conversion from Gamma Corrected Value (B')	17
3.1	RGB to YCbCr Conversion (Y)	22
3.2	RGB to YCbCr Conversion (Cb)	22
3.3	RGB to YCbCr Conversion (Cr)	22
3.4	Log-average Luminance	24
3.5	2D DCT-II Transformation	24
3.6	Watermark Embedding	25
3.7	Inverse DCT Transformation (IDCT)	25
3.8	YCbCr to RGB Conversion (R)	26
3.9	YCbCr to RGB Conversion (G)	26
3.10	YCbCr to RGB Conversion (B)	26
3.11	Watermark Extraction	28
4.1	Mean Square Error Calculation	30
4.2	PSNR Calculation	31
4.3	MOS Calculation	34
4.4	Similarity Measurement of Each Pixel	35
4.5	Similarity Measurement of Whole Image	35

# CHAPTER 1

## INTRODUCTION

This thesis is intended to develop a digital watermarking method that can protect the copyright infringement of digital videos. This chapter firstly introduces the challenging problems which the digital media industry is facing in this high-technology world and how they can be solved by using data encryption and data hiding methods. Then, it explains the importance of digital watermarking methods for Digital Rights Management systems along with some related works. Finally, this chapter is concluded with application areas of digital watermarking, and objectives and organization of the thesis.

No need to doubt, the present time is widely known as the digital age. Today society is more familiar with digital devices and technologies rather than analog. There is no one who does not use any digital devices in his/her daily life. Technological improvements have completely changed the lifestyle of the society. Starting from daily life, every sector such as education, health, economy, etc. has completely moved to digital.

Improvement of digital technologies has also impacted the communication and entertainment sectors. For example, digital video is one of the most common forms of multimedia these days. Digital video can be edited, transmitted, copied, and shared faster and easier on the network than analog data. Digital video streaming sites like YouTube, Metacafe, and social media such as Facebook, Twitter, etc. have become extremely popular among the users of any age. Anywhere and anytime, songs, movies, games, live streaming tournaments, documentaries and more on those sites can be watched; and just need Internet connection and digital devices such as mobile phones or computers. Those improvements make our daily life more comfortable. Not only for entertainment, but also learning any interesting field of study on those sites by watching tutorial videos and get knowledge and build the careers more brilliantly.

However, as a downside, those improvements are followed by unauthorized actions such as copyright infringement, stealing the owner's media, unauthorized publication, and so on. Those works have a severe impact on the digital media markets. The digital media industry is inevitably facing economic failures due to those issues. In addition, those illegal publications may be low quality and they may also be carriers of computer viruses, which are an unwanted threat to the safety of

media users. No need to doubt, those illegal issues are very big problems for the fruitful development of digital video markets [65].

Digital Rights Management (DRM) issues such as copyright problems are the motivating factors in developing the data protection techniques such as data encryption, data hiding, and watermarking techniques for digital video. The following sections introduce those techniques.

## **1.1 Data Encryption (or) Cryptography**

Data encryption or cryptography is a commonly used technique for securing the contents of digital media. It basically is the systematic destruction (i.e. allowing perfect reconstruction) of the media contents that are going to transmit or store. The media intended to be protected may be any kind like text, image, video, and so on. The main aim is to make the media contents invisible or unintelligible by unauthorized persons.

General concept of data encryption is as follows: the data to be protected is known as plaintext and transmits it after encryption. Only the authorized user who has the key can decrypt the data at the receiver end. The encrypted data is known as cipher text. Without the true key, the data cannot be intelligibly accessed by anyone. Based on the key usage and operational procedure, there are a variety of different kinds of data encryption algorithms. Some examples are DES [56], RC4 [40], AES [1], RSA [66], and to name just a few. These algorithms can be applied on any kind of media, e.g. image [17,3,4], audio [2, 48], and video [5, 62]. This thesis only emphasizes digital video contents.

To present some example works of digital video encryption, Chadha and Mallik [6] proposed an encryption method for digital video based on Rivest, Shamir, Adleman Algorithm (RSA) and Pseudo Noise (PN) sequence. It was intended to protect sensitive video data transfers. In that system, it separately encrypted the audio and video components of a digital video to get the desired security level. The RSA and PN-based encryption were used to encrypt the video data, whereas the audio data was encrypted by using the PN-based encryption and Discrete Cosine Transform (DCT). The proposed two-level encryption resulted in negligible similarities in visual perception between the original and encrypted video files, which is a sign of strong candidate for applications whose main focus is the security of data. If the application interest was mainly the visual similarity, encryption could be done for only one level

by using RSA. This could also result in quicker processing for encryption.

Chen and Ding [63] also proposed a video encryption method based on moving objects. That method could effectively balance the effectiveness for security and efficiency of real-time video data. The proposed method firstly extracted the moving objects from a video frame. Those objects were then encrypted by using an encryption algorithm based on the entropy encoding. The method had negligible computational overhead and great encryption real-timeliness as per the experiments.

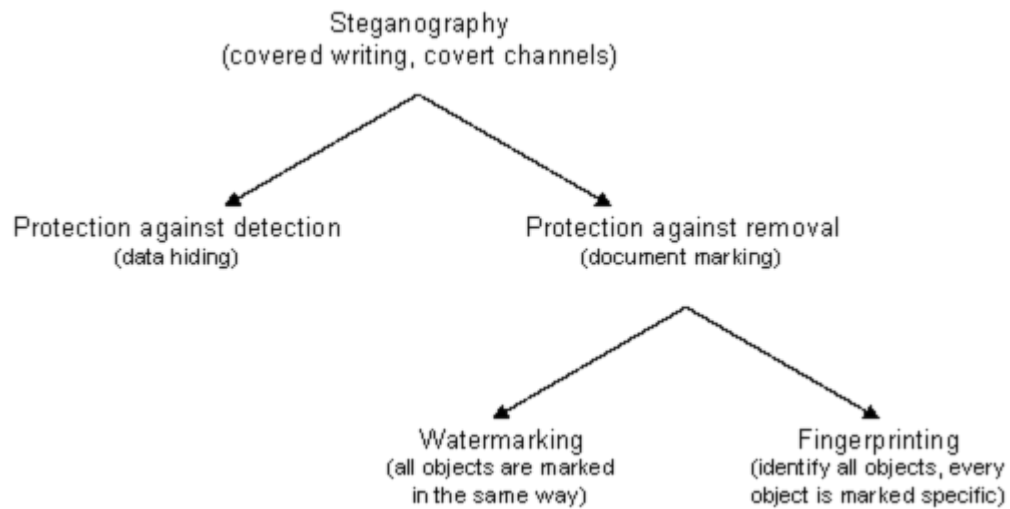
As mentioned, cryptographic techniques can reasonably protect the digital media from unauthorized access. They ensure data confidentiality, authentication, and integrity. However, unfortunately, those techniques only protect the media before decryption; the media becomes unprotected after decryption. It yields that cryptographic techniques cannot be effectively used for solving copyright issues. Then, data hiding and digital watermarking techniques become handier for dealing with copyright issues.

## **1.2 Data Hiding and Watermarking**

The origin of data hiding and watermarking comes from steganography. Steganography is also known as “cover writing”. Its main concept is to hide the message in the digital media instead of encrypting the message. Only the sender and intended receiver can understand that the message is hidden in the transmitted data [31]. In steganography, the transmitted data is termed as the cover or the host and it can be accessible by anyone. However, accessing the hidden message needs the actual key, which is assumed to be known only by the sender and the intended receiver. Thus, the main focus of steganography is for covering the point-to-point communication between two parties.

Stenographic methods can be classified into two specific groups depending on the purpose, namely data hiding and watermarking, as shown in Figure (1.1). In data hiding, steganography is used for hiding the data or information in the cover, which will be retrieved later by a specific end user or party. The main purpose in this scenario is to protect the hidden information from being detected by the other users who do not have the actual key. In watermarking, the message to be inserted is used to protect the host media and its owner [50]. Mostly, the hidden message is something that can uniquely identify the media owner. In this case, trying to remove the hidden message without the actual key will damage the host media; any illegal access to the

host media can be detected by detecting those damages. Thus, in case of DRM issues such as copyright problems, digital watermarking techniques are increasingly popular.



**Figure 1.1 Types of Steganography**

As in cryptography, steganographic techniques can be applied on any kind of media such as text, image, video, etc. In digital image and video watermarking, copyright information is hidden in host image and video frames and must be difficult to extract or distort by a variety of attacks. In this digital age, digital video watermarking is widely applied in applications like copyright protection, broadcasting media, authentication of video transmission, fingerprinting, and more. A lot of digital video watermarking methods used in copyright control applications have been proposed in the literature. Some of them are discussed below.

Kumar and Shukla [43] implemented a DCT based watermarking technique for AVI videos using Matlab Simulink. In that paper, the watermark image was firstly segmented and then embedded in separate frames. It means that, for identifying the copyright owner, watermark must be extracted from multiple frames and reconstructed. In the case of frame loss, watermark detection cannot be successful and copyright protection fails.

Yadav and Anand [57] also proposed a DCT based digital video watermarking technique using Matlab Simulink. The main aim of that method was to detect modifications on the host video by using visible watermarks.

Kalra [21] also proposed a DCT and thresholding based digital video watermarking method that repetitively embedded the same watermark in every frame of the video. Even if some frames were lost, the watermark could be successfully

detected in other frames.

Ambadekar, Jain, and Khanapuri [54] proposed a digital image watermarking method which could be used in data authentication and copyright protection applications. The proposed image watermark embedding and extraction methods were developed based on the Discrete Wavelet Transform (DWT) coefficients, distance measurement, and encryption. The multi-resolution property of the DWT could provide the simplest structure for both watermarks embedding and extraction processes. The system achieved a Peak Signal to Noise Ratio (PSNR) of greater than 50 dB and also showed considerable resistance for compression, noise, and geometric transformations.

Rajkumar and Malemath [22] used the concept of video steganography, where the data was hidden behind the frames of videos. This paper provides two levels of security to the data using the power of steganography and cryptography. First the data was encrypted using the AES algorithm and then the encrypted data was embedded into frames of videos. The technique used to embed the data was Least Significant Bit (LSB) coding. It is the most common technique which can hide large amounts of data in the simplest and most efficient way. Watermarking algorithms may have different requirements based on the intended applications. Generally, the performance of a video watermarking algorithm can be considered efficient if the algorithm is imperceptible, undeletable, undetectable, and robust against compression and intended and unintended signal processing operations [27]. The embedded watermark must be imperceptible so that it cannot degrade the host video quality. It must also be undeletable and undetectable so that it sticks with the host video and proves the copyright owner at any necessary time. Robustness can be said to be the main requirement of copyright control applications as the hidden copyright information must reside in the host video no matter how severe illegal intended manipulation occurs on the host video.

In this thesis, digital video watermarking technology is applied to prevent the copyright infringement of the digital video owner. As it is intended to detect the unauthorized access of the media, a watermark logo that can clearly identify the copyrighted owner is inserted to the video and managed to be extracted successfully in case it is needed to identify the owner. The proposed system is implemented in the DCT domain to achieve good robustness, which is the main requirement of copyright protection applications.

### **1.3 Application Areas of Digital Watermarking Techniques**

For all kinds of digital media including digital video which require security and owner identification, watermarking methods can be effectively applied. A few most common applications are listed hereby.

- 1. Owner Identification**

It can be used to identify the owner of any media.

- 2. Unauthorized Copy Protection**

It can be used to prevent illegal copying.

- 3. Broadcast Monitoring**

It can be used to monitor the broadcasting of TV programs and news.

- 4. Theater Identification**

It can be used to detect the illegal act on the films played in theaters. Some theaters embed theater identification watermarks on their movies; owners can act if this identification is detected from a copy.

### **1.4 Objectives of the Thesis**

The objectives of the proposed system in this thesis are described below.

- To solve the security issues of digital information distribution
- To develop a system that prevents copyright infringement of the digital video
- To study video watermarking techniques
- To study how to apply digital signal processing and video signal processing techniques in real world applications

### **1.5 Motivation of the Thesis**

From another viewpoint, violation of digital media could be considered as the motivation of developing good digital watermarking techniques. Digital watermark has become the preferred choice for copyright protection and more advanced watermark embedding and detection algorithms are proposing constantly.

There are different types of digital watermark such as fragile watermark, blind watermark, etc. Each has its significant purpose for intended applications. These watermarks can be applied on any type of digital media like digital documents, image, video, and more. This thesis emphasizes the robust features of digital video watermarking. By enhancing the robustness of the embedded watermark to any intended video manipulation attacks, the proposed system is intended to solve the



copyright violation issues and to be able to help the healthy development of the digital video industry.

## **1.6 Organization of the Thesis**

This thesis will be described in five chapters. Chapter 1 firstly introduces the most commonly used techniques for solving the DRM issues such as data security and copyright protection problem. Next, Chapter 2 presents a thorough explanation on the background theories of digital watermarking and other inclusive components and methods of the thesis. Then, Chapter 3 discusses the proposed watermark embedding and extraction processes in detail. Finally, Chapter 4 and 5 elaborates the experimental results and conclusion of the thesis, respectively.

## CHAPTER 2

### BACKGROUND THEORY

This chapter discuss the theoretical background of the digital image watermarking, which is the basis of digital video watermarking. Digital image watermarking is considered as a method which embeds the watermark into a host image in an unobtrusive way. A watermark is a more or less transparent data that has been inserted to the host image, to protect the original data or to make it more difficult to copy the item, e.g. money watermarks or stamp watermarks. Digital image watermarking techniques can be categorized in various ways. Classifications of the watermarking such as visibility, robustness, blind and non-blind watermarks, host media and watermark data types, and different domains of embedding watermark are discussed in the following subsections.

#### 2.1 Physical and Digital Watermarks

Watermarking was firstly at Bologna, Italy in 1282. At first, it was used in paper mills as a paper mark of the company [41]. Then it was common in practice up to the 20th century. Also, in Myanmar, watermarks are used in the postage stamps and currency notes as shown in Figure (2.1).

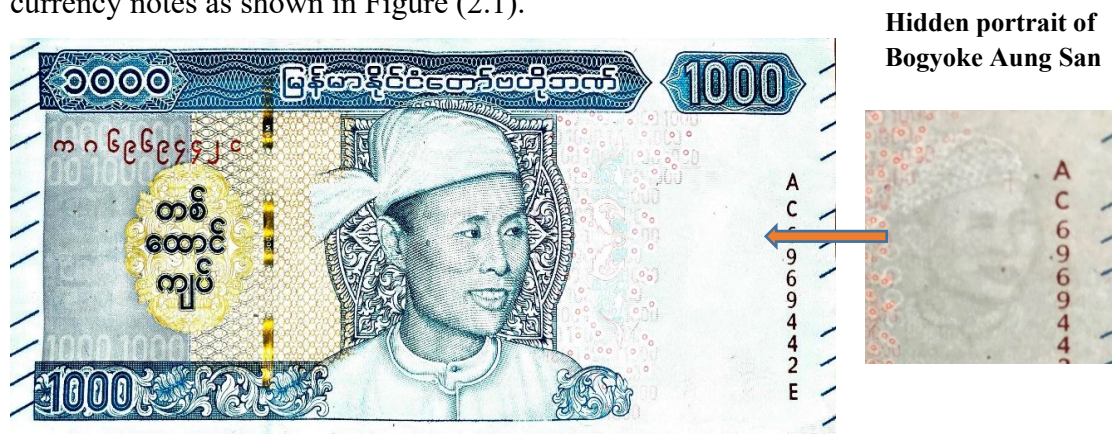


Figure 2.1 Physical Watermark

In the analog world, a painting is signed by the artist to attest the copyright, an identity card is stamped by the steel seal to avoid forgery, and the paper money is identified by the embossed portrait [13]. These types of signatures, seals or watermarks are physical watermarks and have been used from ancient times as a way to recognize the source, creator of a document or images or pictures.

In digital image watermarking, the watermark must be digitally embedded into

the host digital image. As an example, digital watermarks can be seen on some TV programs, which identify the copyright owner of the program. They can also be seen on online printed material like PDF files. The most common purpose of digital watermarking is to identify the copyright owner of the media. Video watermarking is also derived from image watermarking to hide copyright information into video sequence to prevent copyright infringement [53].

## **2.2 Classification of Watermarks**

Based on the operational procedure and application requirements, digital image watermarks can be grouped into various ways. Common classifications of watermarks are discussed in following subsections.

### **2.2.1 Visibility of Watermark**

Watermarks can be divided into two main groups – visible and invisible watermarks. A visible watermark is a visible semi-transparent text or image overlaid on the original image [12]. It allows the original image to be viewed, but it still provides copyright protection by marking the image as its owner’s property. Visible watermarks are robust against image transformation, especially when you use a semi-transparent watermark placed over the whole image. Thus, they are preferable for strong copyright protection of intellectual property that’s in digital format.

An invisible watermark is an embedded image which cannot be perceived with human’s eyes [11]. Only electronic devices (or specialized software) can extract the hidden information to identify the copyright owner. Invisible watermarks are used to mark a specialized digital content (text, images or even audio content) to prove its authenticity [20].

### **2.2.2 Robustness of Watermark**

One of the most commonly measured properties of digital watermarking systems is that the watermark signals must be reasonably resilient to various attacks and common signal processing operations. Based on how robust the watermark is to the attacks, watermarking methods can be categorized as robust, semi-fragile, and fragile watermarking.

Once some watermark signal is inserted in the original content, distortions may be applied to the signal unavoidably when the signal is encoded, decoded, and

distributed across the Internet. These distortions may be designed to apply the expected distortion to the watermarked signals or compress it before transmission, and they may or may not significantly disrupt the watermarked signals. In robust watermarking, digital watermark should be difficult or impossible to remove or separate from the original image. It is impossible for a watermarking system to be robust against all signal processing operations whereas the requirement is application dependent [24]. Robust watermarks are usually designed to resist arbitrary malicious attacks such as image scaling, bending, cropping, compression, etc., and they are commonly used for copyright protection of the host media [18].

In case of digital video watermarking, a robust watermarking method is likely to resist noise addition, JPEG compression, filtering, cropping, and geometrical transformations such as scaling, translation, and rotation as well [51]. Application of this type of watermark is copyright protection such as inserting the author name, work serial number, and other relative information into the host video frames, i.e. images.

Semi-fragile watermarking algorithms for image authentication focus on the ability of detection, location, and recovery from tamper attacks carried out on images [14]. Because the semi-fragile watermark has a certain degree of fragility, when the host image suffers attacks, the watermarking information will make a corresponding change as well. The algorithm can then realize the image authentication by detecting if the watermark is damaged or not. However, the semi-fragile watermark should have a certain degree of robustness for normal image-processing operations so that normal and malicious tampering operations can be distinguished [59].

Likewise, fragile watermark is a type of watermark that inserts information to the host unreceptive. It is easy to destroy. Fragile watermarks are adopted and designed to detect any unauthorized modification, even for the slightest changes to the data. When the watermarked image has been changed, the watermark will be changed correspondingly so that the owner can know the host media has been tampered [58].

### **2.2.3 Blind and Non-Blind Watermark**

This property is related to the watermark extraction or detection process. In blind watermarking, the hidden watermark information can be successfully detected/extracted without any knowledge of the original, un-watermarked content. This is in contrast to informed or non-blind detection, which needs some knowledge of the original un-watermarked work. This knowledge can take the form of the

original work itself or some function of the original work [33].

#### **2.2.4 Host Media and Watermark Data Types**

The watermark can be embedded into any kind of host/cover media, e.g. image, video, audio, etc. The watermark itself can be any kind as well. Before selecting the watermarking methods, type of the host media and the watermark needs to be determined. The watermark can just be treated as a pure noise which is added into the host media [10]. Sometimes the watermark can be added as a noise but with other side information such as the information of the owner can just be considered as a second message that must be transmitted along with the watermarked message. All of this means that watermarking is known as a communication-based model. The watermark embedder wishes to communicate with the watermark receiver by using a watermark as a message [38].

In applications of copyright protection, the content of the watermark must uniquely identify the owner and it can be such as

1. Signature,
2. Copyright logos / messages,
3. Ownership identifiers marks, and
4. Other digital information formats.

The size of the watermark is also an important factor. Watermarks are often used as owner identification or security confirmation of the host signal when the data is transmitted. It is important that the size of the watermark should be minimal because it will increase the size of the data to be transmitted [9]. However, the small-size watermark is considered to contain less weight of message information than the larger size.

### **2.3 Different Domains of Embedding Watermark**

Digital image watermarking can be performed in both spatial and transform domains.

#### **2.3.1 Digital Video Watermarking in Spatial domain**

Spatial domain-based watermarking methods embed the watermark bits directly to the pixels of the cover image. Spatial domain methods make use of the human visual system and can be easily modeled and analyzed mathematically.

However, the embedded watermark can be easily destroyed or removed by signal processing attacks such as filtering [39].

The least significant bit (LSB) method is an example of a spatial domain method where the watermark is embedded into the least significant bits of the cover image [23]. The LSBs are highly sensitive to noise and thus the watermark can easily be removed by image manipulations such as rotation and cropping [44]. The correlation-based method is another example of spatial domain technique in which the watermark is converted into a pseudo noise sequence which is then weighted and added to the cover image bits.

The spatial domain methods are less complex compared to transform domain methods, however weak to different image attacks. However, data hiding capacity of spatial domain techniques is higher than that of transform domain methods.

### **2.3.2 Digital Video Watermarking in Frequency Domain**

The robustness and imperceptibility of the watermarked image/video can be improved by performing watermarking in frequency domain. In that domain, watermark embedding is done by modifying the image coefficients using image transforms. Masking techniques based on the transform domain are more robust than the LSB method with respect to cropping, compression, and image processing. Many of the transform coefficients are small; hence even though they are discarded during the process of compression, the effect is negligible [49].

Many transforms had been considered for digital image watermarking. Some examples are Discrete Fourier Transform (DFT), Fast Fourier Transform (FFT), Fourier-Mellin Transform, Fractal transform, and Discrete Wavelet Transform (DWT).

In wavelet domain-based methods [55], both the image and watermark are transformed into the DWT domain by using the multi-resolution wavelet decomposition. Those methods deploy the human visual system. When the image is decomposing by wavelet transformation, its components are separated into bands of scale, much like the retina of the eye splits an image into several components [52]. It then allows the independent processing of the resulting components much like the human eyes. The low-frequency components must be modified in order to embed the information in a reliable and robust way. The DWT is highly robust to JPEG compression, additive noise, and linear filtering.

### 2.3.3 Digital Video Watermarking in Discrete Cosine Transform Domain

Discrete Cosine Transform (DCT) is also widely used in digital image/video watermarking. It expresses a finite sequence of data points in terms of a sum of cosine functions oscillating at different frequencies. The use of cosine rather than sine functions is critical for compression, since it turns out that fewer cosine functions are needed to approximate a typical signal [30].

In particular, the DCT is a Fourier-related transform similar to the DFT except using only real numbers. It converts the spatial domain (pixel based) to the frequency domain. It is based on orthogonal transform, which is the most commonly used linear transform in digital signal processing [37]. It is widely used in digital signal compression such as image compression and other fields. JPEG compression is a standard established on the basis of the DCT transform. Watermarking in the DCT domain has enhanced the ability to resist compression on watermark.

There are different types of DCT transformations as stated below.

#### (a) One-Dimensional DCT

- DCT-I
- DCT-II
- DCT-III
- DCT-IV
- DCT-V to VIII

#### (b) Multidimensional DCT

- 2D DCT-II
- 3D DCT-II
- MD DCT-IV

#### (c) Inverse DCT Transform

Among the mentioned transforms, two dimensional DCT (2D DCT) is used to transform an image from the spatial domain to frequency domain. It represents an image as an addition of sinusoids of varying magnitudes and frequencies. It has the property that, for a typical image, most of the visually important information about the image is concentrated in just small coefficients of the DCT [32]. For this reason, the DCT is frequently used in image compression applications.

A 2D DCT-II of an image is simply the 1D DCT-II performed along the rows and then along the columns (or vice versa). The equation of a 2D DCT-II for an  $N \times N$

image is given in eq. (2.1).

$$D_{u,v} = \frac{1}{\sqrt{2N}} C_u C_v \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} Y_{x,y} \text{Cos} \left[ \frac{(2y+1)v\pi}{2N} \right] \text{Cos} \left[ \frac{(2x+1)u\pi}{2N} \right], \quad (2.1)$$

where  $Y_{x,y}$  is the pixel at row  $x$  and column  $y$ ,  $D_{u,v}$  is the DCT coefficient in row  $u$  and column  $v$ , and

$$C_u = \begin{cases} \frac{1}{\sqrt{2}}, & \text{for } u = 0 \\ 1, & \text{otherwise} \end{cases}, \quad C_v = \begin{cases} \frac{1}{\sqrt{2}}, & \text{for } v = 0 \\ 1, & \text{otherwise} \end{cases}.$$

Inverse DCT transform (IDCT) is needed for reconstruction of an image. The equation of IDCT for an  $N \times N$  image is given in eq. (2.2).

$$Y'_{x,y} = \frac{1}{\sqrt{2N}} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} C_u C_v D'_{u,v} \text{Cos} \left[ \frac{(2y+1)v\pi}{2N} \right] \text{Cos} \left[ \frac{(2x+1)u\pi}{2N} \right], \quad (2.2)$$

where  $D'_{u,v}$  is the DCT coefficient in row  $u$  and column  $v$  and  $Y'_{x,y}$  is the reconstructed pixel at row  $x$  and column  $y$ .

Among the mentioned transform domains, DCT is more widely used for image and video watermarking. It is because the DCT has been proved as a good choice for compressing the images; it is used in the JPEG image compression standard [7]. As a video is a sequence of image frames, DCT also works well for videos. Watermarking methods developed in the DCT domain are mostly robust against compression attacks. This thesis also implements a digital video watermarking method in the DCT domain aiming to achieve high robustness to signal processing attacks including compression.

The following section presents some background theory regarding digital video.

## 2.4 Background on Video

Video is an electronic media used for recording, playback, broadcasting, and display of moving visual data [61]. When digital techniques are used in creating a video, the result is called a digital video. In the early 1970s, the very first digital video was created with the DCT coding which is a lossy compression process [42, 47]. In the late 1980s, the DCT coding was adapted into motion-compensated DCT video compression, e.g. H.261 [25]. Compared to the earlier analog technology, digital video was capable of higher quality and much lower cost.



### 2.4.1 Structure of a Video

A digital video is made up of a series of orthogonal still digital images displayed in rapid succession at a specified rate. These images are called frames in the context of video. The rate at which the frames are displayed is measured in frames per second (FPS) or frame rate. The minimum frame rate to achieve a reasonable illusion of a moving image is about 16 FPS. As each frame is an orthogonal still digital image, it comprises a raster of pixels. If a frame has a width of  $w$  pixels and a height of  $h$  pixels, then the frame size is  $w \times h$ .

A pixel has only one property, its color. The number of distinct colors a pixel can represent depends on the color depth of a video measured in the number of bits per pixel. The more the bits per pixel, the subtler variations of colors can be presented. However, the human eye is not sensitive to details in color as much as brightness. Thus, if we want to reduce the number of bits required to encode a digital video, a common way is to use chroma subsampling. In that scheme, the luminance values of all pixels are kept as they are, while the chrominance values are averaged for a number of pixels in a block and that same value is used for all of them. This scheme has no impact on the number of possible color values that can be displayed; however, it reduces the number of distinct points at which the color changes. The generalized structure of a video is depicted in Figure (2.2).

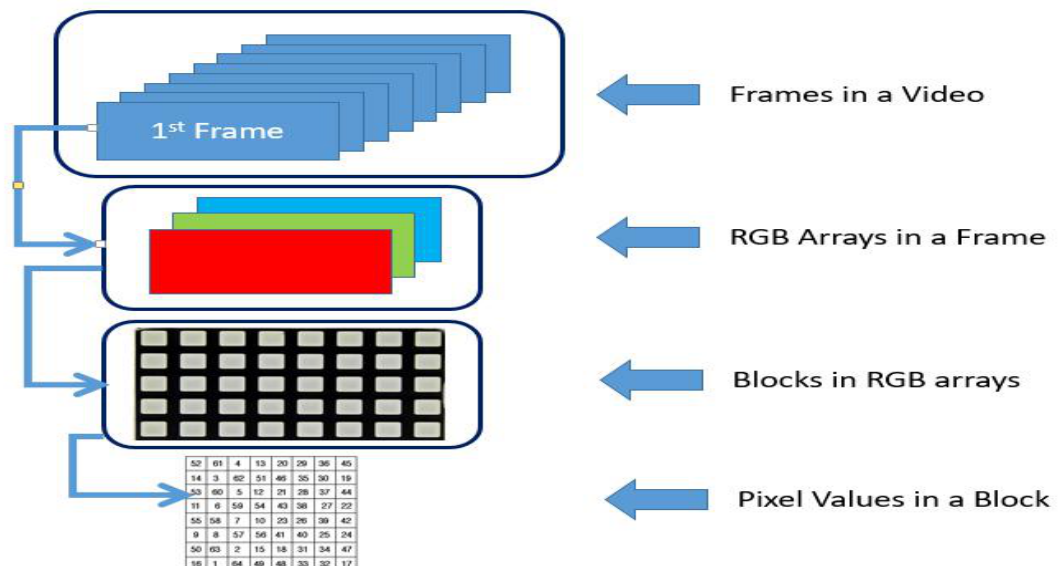


Figure 2.2 Structure of a Video

### **(a) Frames in a Video**

Consider a video that has 10 seconds of duration with frame rate 25 FPS. It means 25 frames are being displayed in one second and that video comprises a total of 250 frames. Each frame is a separate and different image, and they are combined to form a video.

### **(b) RGB Arrays in a Frame**

In the case of RGB images, each frame consists of three layers of color arrays. Each represents Red, Green, and Blue color, respectively. Each color layer value is mostly represented by 8 bits. Thus, for RGB frames, each pixel value is encoded in 24 bits, and there will be a total of  $256 \times 256 \times 256$  (~16 millions) colors available.

### **(c) Blocks in RGB Arrays**

Considering the size of a video frame is  $640 \times 480$ , there will be  $640 \times 480 = 307,200$ -pixel values in total in each layer of the color array. Instead of processing those large numbers as a whole, most video processing techniques work on separate blocks. Most common block sizes are  $8 \times 8$  or  $16 \times 16$ .

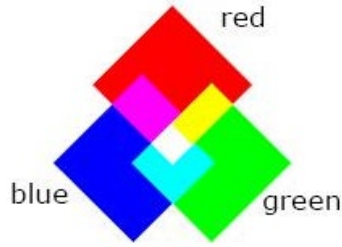
### **(d) Pixel Values in a Block**

Each block is composed of pixels. In the case of an  $8 \times 8$  block, it has width of 8 pixels and height of 8 pixels, so a total of 64 pixels. Each pixel is mostly encoded in 24 bits to define color.

## **2.4.2 Color Models**

The aim of a color model is to facilitate the specification of colors in some standard way. As an example, the most commonly used specifications are the RGB color model typically used in computer graphics and YCbCr model used for digital video. Transforming color information from one model to another requires transformation from one set of values to another.

In the RGB model, each color represents a combination of red, green, and blue, which are primary colors. This model is known as additive. That is, the primary colors can be added to produce any other secondary colors of light. Figure (2.3) shows the RGB model's primary and secondary colors – magenta (red + blue), cyan (green + blue), and yellow (red + green). The white color is assumed as a combination of red, green, and blue at full intensities [15].



**Figure 2.3 Primary and Secondary Colors of RGB Model**

The YCbCr color space which is mainly used for component digital video was developed as part of the ITU-R BT.601 Recommendation. Y is the luminance and Cb and Cr are the blue-difference and red-difference chroma components. The Intel IPP functions use the basic equations shown in Equation (2.3) – (2.8) to convert between R'G'B' in the range 0-255 and Y'Cb'Cr' (prime (') notation means that all components are derived from gamma-corrected RGB) [16].

$$Y' = 0.257 \times R' + 0.504 \times G' + 0.098 \times B' + 16 \quad (2.3)$$

$$Cb' = -0.148 \times R' - 0.291 \times G' + 0.439 \times B' + 128 \quad (2.4)$$

$$Cr' = 0.439 \times R' - 0.368 \times G' - 0.071 \times B' + 128 \quad (2.5)$$

$$R' = 1.164 \times (Y' - 16) + 1.596 \times (Cr' - 128) \quad (2.6)$$

$$G' = 1.164 \times (Y' - 16) - 0.813 \times (Cr' - 128) - 0.392 \times (Cb' - 128) \quad (2.7)$$

$$B' = 1.164 \times (Y' - 16) + 2.017 \times (Cb' - 128) \quad (2.8)$$

### 2.4.3 Video Encoding Format

The content representation format for storage or transmission of digital video content (such as bitstream) is known as video encoding format or sometimes video compression format. Most commonly, the video coding format uses a standardized video compression algorithm based on DCT coding and motion compensation. Some examples of video coding formats are MPEG-4 Part 2 [45], H.264 (MPEG-4 Part 10) [8], HEVC (H.265) [26] and to name just a few.

A specific software or hardware implementation that is capable of compression and decompression to and from a specific video coding format is known as a video codec. Video contents encoded in a particular video encoding format are normally bundled with an audio stream which is encoded using an audio coding format. They are then packaged inside a multimedia container format such as AVI and MP4 [60]. That is, the user normally has a .mp4 video file instead of having a H.264 file, which really is an MP4 container containing the H.264-encoded video. A

number of different video encoding formats can be bundled in a multimedia container format; e.g. either the MPEG-2 Part 2 or the H.264 video coding format can be bundled in the MP4 container format.

In this thesis, the proposed system can accept any kind of multimedia container format, e.g. AVI or MP4, as input. Then, by deploying the fact that the human visual system is not sensitive to light intensity changes and compression resistance capability of DCT, it embeds the copyright control information in the DCT transformed luminance values of each video frame. That is why the proposed system is intended to maintain the visual quality of the watermarked video as high as possible and also robust to most common signal processing attacks including compression.

## CHAPTER 3

### DCT-BASED VIDEO WATERMARKING METHOD

Digital video watermarking is a technology in which watermark information is embedded in a host video, with the purpose of copyright protection. This chapter discusses the proposed DCT-based video watermarking method in detail. It gives a thorough explanation of the basic processes of a digital video watermarking system such as watermark generation, embedding, and detection or extraction.

#### 3.1 Selection of System Criteria

The digital video watermarking systems can be applied in many different applications, e.g. owner identification, unauthorized copy protection, broadcast monitoring, theater identification, and so on. Based on the application purpose, the requirements and characteristics of a watermarking system may be different. In this system, the main aim is to identify the owner of the media, so we should set the characteristics of the watermarking system to meet that objective. Table (3.1) shows the different characteristics of a watermarking system; the choice of this proposed system is highlighted in “bold”.

Firstly, in this system the watermark type to be embedded in the host video is chosen as logo (image) of “.jpg” format. It should clearly identify the owner in the case of copyright violation. That logo image is converted into a binary image before embedding in the host.

The robustness characteristic of the proposed system should be “robust”. No matter how severe unauthorized and intended manipulations are subjected to the host video, the embedded logo image should be completely recoverable. In this way, the owner of the media can be proved.

As for the watermarking domain, the frequency domain is chosen because watermarking in that domain can improve the robustness and imperceptibility of the watermarked video. Among the possible frequency domains, the DCT domain is chosen for its robustness to compression.

In case of “perceptivity”, invisible watermark type is used because there should be perceptual similarity between the original and the watermarked version of the video. Invisible watermarks can also prevent users from extracting the logo easily by the unauthorized user.

**Table 3.1 Different Characteristics of Watermarking Systems**

<b>No.</b>	<b>Criteria</b>	<b>Classification</b>
1.	Watermark Type	1. Noise: pseudo noise, Gaussian, random, and chaotic sequences 2. <b>Image</b> : logo, stamp, signature, etc.
2.	Robustness	1. Fragile: easily manipulated 2. Semi-fragile: resist from some type of attacks 3. <b>Robust</b> : not affected from attack
3.	Domain	1. Spatial: e.g. LSB 2. <b>Frequency</b> : e.g. DWT, <b>DCT</b> , DFT
4.	Perceptivity	1. Visible watermarking: e.g. channel logo 2. <b>Invisible watermarking</b> : like steganography
5.	Host Media	1. Image 2. Text 3. Audio 4. <b>Video</b>
6.	Data Extraction	1. Blind 2. Semi-blind 3. <b>Non-blind</b>

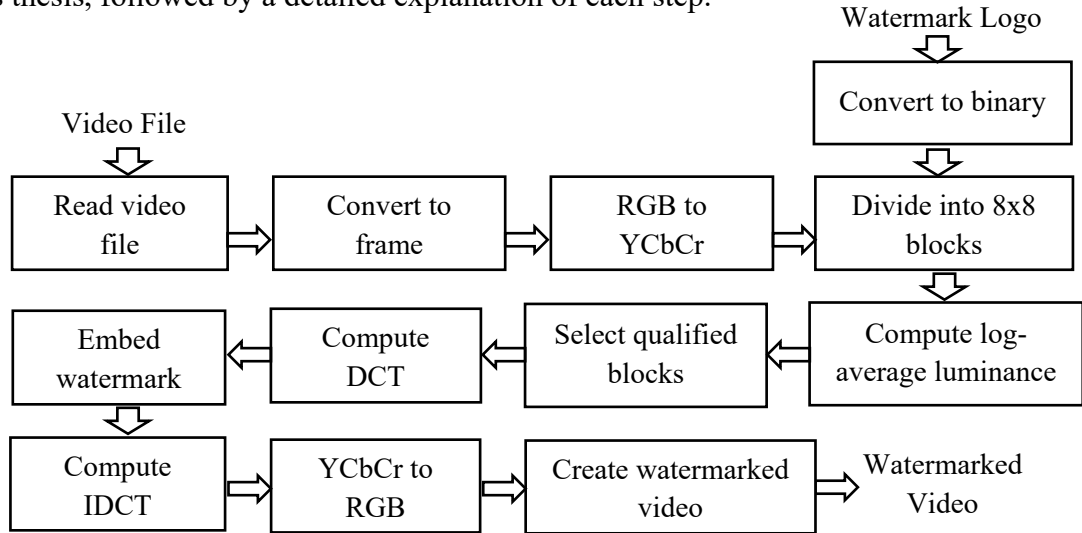
As for the “host media”, the proposed system is a video watermarking system; a video file in “.AVI” format is chosen as the host. File size of the host video can be any size; smaller file size (assuming fewer frames) helps the watermarking algorithm run faster but the watermark embedding capacity may be low. Bigger file size will take long to embed the watermark but with higher embedding capacity. The size of each video frame is also important for perceptibility of the video. The larger the frame size, the better the perceptual quality of the embedded video frame. All experimental videos used in this system are 15-16 seconds long and frame size is 640x480.

As for the “watermark extraction” process, this system uses a “non-blind” method, which needs the original data to extract the watermark. The system is aimed to protect the identification of the owner, thus using the “non-blind” method ensures that the owner who has the original data can successfully extract the watermark.

In the following sections, the main two processes of the proposed system, watermark embedding and extraction processes, are discussed in detail.

### 3.2 Watermark Embedding Process

Figure (3.1) depicts the flow of the watermark embedding process proposed in this thesis, followed by a detailed explanation of each step.



**Figure 3.1 Flow of the Proposed Watermark Embedding Process**

To start the embedding process, the host video that will carry the watermark is read as an input. Although the proposed method can be applied on any video file type, e.g. AVI, MP4, etc., and any frame size as well, the following criteria for host video selection should be considered for efficient processing. Using a bigger frame size can cause more complexity, take more space while processing, more blocks to process, and will result in higher quality after embedding. Using a smaller frame size can cause less complexity, take less space while processing, fewer blocks to process, and will result in lower resolution after embedding. For the experiments demonstrated in the next chapter of this thesis, MP4 video files will be used with frame size of 640x480 as the host.

As the next step, the host video is converted into image frames. The number of frames will be different depending on the frame rate. As an example, a 16-sec long video with a frame rate of 30 fps consists of 480 frames. The proposed method has no restriction on the frame rate and size. However, it is noted that the more the frames, the more the time and space complexity as every single frame needs to be processed in this system.

Next, the color format of each frame is changed to Luminance, Chroma Blue, Chroma Red color format (YCbCr) because this system embeds the watermark in the luminance value ( $Y$ ) of the pixel. The  $Y$  values are chosen for watermark embedding because the human visual system is not sensitive to the light intensity changes, i.e. luminance value changes. Embedding the watermark in  $Y$  component will keep the invisibility of the watermarks and thus preserve the video quality as high as possible. Equations (3.1) - (3.3) show how to convert RGB colors to YCbCr format.

$$Y = 16 + \frac{65.74}{256} \cdot R + \frac{129.06}{256} \cdot G + \frac{25.06}{256} \cdot B, \quad (3.1)$$

$$Cb = 128 - \frac{37.95}{256} \cdot R - \frac{74.49}{256} \cdot G + \frac{112.44}{256} \cdot B, \quad (3.2)$$

$$Cr = 128 + \frac{112.44}{256} \cdot R - \frac{94.15}{256} \cdot G - \frac{18.29}{256} \cdot B, \quad (3.3)$$

where  $R$ ,  $G$ , and  $B$  are red, green and blue components of RGB color space respectively.

For clearer understanding, Figure (3.2) shows how MATLAB command “`rgb2ycbcr`” is used to convert RGB to YCbCr color space as an example. The results are exactly the same as manual calculation depicted in Example (3.1).

<pre>&gt;&gt; Img 64x64x3 uint8 array Img(:,:,1) = Columns 1 through 18 21 19 16 10 4 0 0 5</pre>	<pre>&gt;&gt; Y=rgb2ycbcr(Img); &gt;&gt; Y 64x64x3 uint8 array Y(:,:,1) = Columns 1 through 18 21 21 20 19 17 16 19 24</pre>
<pre>Img(:,:,2) = Columns 1 through 18 0 0 0 0 0 0 0 5</pre>	<pre>Y(:,:,2) = Columns 1 through 18 125 125 126 127 127 128 127 126</pre>
<pre>Img(:,:,3) = Columns 1 through 18 0 0 0 0 0 0 0 1</pre>	<pre>Y(:,:,3) = Columns 1 through 18 137 136 135 132 130 128 126 125</pre>

Figure 3.2 RGB to YCbCr Conversion



### Example 3.1: YCbCr Calculation Example

For the pixel at row = 1 and column = 8,

$$R = 5, G = 12, B = 4,$$

$$\begin{aligned} Y &= 16 + \frac{65.74}{256} \cdot R + \frac{129.06}{256} \cdot G + \frac{25.06}{256} \cdot B \\ &= 16 + \frac{65.74}{256} \cdot 5 + \frac{129.06}{256} \cdot 12 + \frac{25.06}{256} \cdot 4 = 23.7 \approx 24 \end{aligned}$$

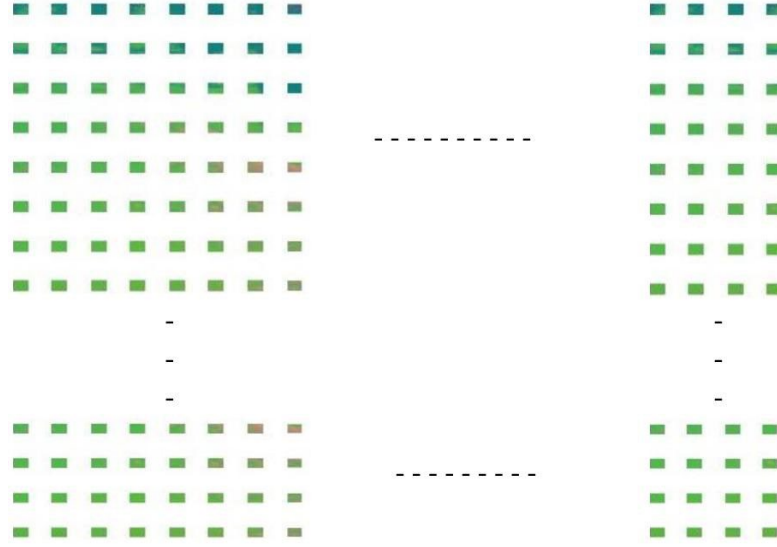
$$\begin{aligned} Cb &= 128 - \frac{37.95}{256} \cdot R - \frac{74.49}{256} \cdot G + \frac{112.44}{256} \cdot B \\ &= 128 - \frac{37.95}{256} \cdot 5 - \frac{74.49}{256} \cdot 12 + \frac{112.44}{256} \cdot 4 = 125.5 \approx 126 \end{aligned}$$

$$\begin{aligned} Cr &= 128 + \frac{112.44}{256} \cdot R - \frac{94.15}{256} \cdot G - \frac{18.29}{256} \cdot B \\ &= 128 + \frac{112.44}{256} \cdot 5 - \frac{94.15}{256} \cdot 12 - \frac{18.29}{256} \cdot 4 = 125.4 \approx 125 \end{aligned}$$

As the next step, the  $Y$  component of each frame is then divided into  $8 \times 8$  blocks. In order to keep the best possible video quality, the watermark will not be embedded into the whole frame. Instead, each frame is divided into blocks and only the selected blocks will be used for watermark embedding. In this way, the distortion introduced by watermark embedding can be controlled to some extent. For the frame size of  $640 \times 480$ , there are a total of  $(640 \times 480) / (8 \times 8) = 4800$  blocks per frame. Figure (3.3) shows the logical frame structure after dividing into  $8 \times 8$  blocks.

Not only the host video frame, the watermark logo is first converted to binary and also divided into  $8 \times 8$  blocks as each watermark block is embedded into each host frame block. For a watermark size of  $64 \times 64$ ,  $(64 \times 64) / (8 \times 8) = 64$  blocks are produced.

The next step is to select the image blocks for watermark embedding. When a  $640 \times 480$  image frame is divided into  $8 \times 8$  blocks, 4800 blocks are produced. Since the  $64 \times 64$  watermark logo has 64  $8 \times 8$  blocks, only 64 host image blocks are required to accomplish the watermark embedding process. Those blocks are selected based on the log-average luminance of the entire image and of each block. The log-average luminance is calculated as described in Equation (3.4).



**Figure 3.3 Frame Structure after Dividing into 8x8 Blocks**

$$L_{avg} = \exp(\sum \text{Log}(\delta + Y_{x,y}) / N), \quad (3.4)$$

where  $L_{avg}$  is the log-average luminance,  $Y_{x,y}$  is the luminance  $Y$  of the pixel at location  $[x,y]$  of the host image/block,  $\delta$  is a small value (0.1 in this system) used to avoid taking the log of a black pixel, and  $N$  is the number of pixels in the image/block.

The block selection criterion is as follows:

Among the 4800 blocks, the 64 blocks whose log-average luminance is the closest to that of the entire image are selected. The log-average luminance of each selected block should be in the range  $[L_{avg} - \beta, L_{avg} + \beta]$  where  $L_{avg}$  is the log average luminance of the host image and  $\beta$  is the minimum floating-point value that is enough to determine an adequate number of blocks.

After block selection, the next step is to apply the DCT on the  $Y$  component of each selected block using the forward DCT transform shown in Equation (3.5).

$$D_{u,v} = \frac{1}{\sqrt{2N}} C_u C_v \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} Y_{x,y} \text{Cos}\left[\frac{(2y+1)v\pi}{2N}\right] \text{Cos}\left[\frac{(2x+1)u\pi}{2N}\right], \quad (3.5)$$

where  $N$  is 8 for the block size of  $8 \times 8$ ,  $Y_{x,y}$  is the luminance  $Y$  of the pixel at location  $[x,y]$  of a block,  $D_{u,v}$  is the DCT coefficient in row  $u$  and column  $v$ , and

$$C_u = \begin{cases} \frac{1}{\sqrt{2}}, & \text{for } u = 0 \\ 1, & \text{otherwise} \end{cases}, \text{ and } C_v = \begin{cases} \frac{1}{\sqrt{2}}, & \text{for } v = 0 \\ 1, & \text{otherwise} \end{cases}.$$

The next step is the watermark embedding process. The pixels in each  $8 \times 8$  block of the watermark image (already converted to binary) are embedded in the DCT coefficients of each selected block as defined in Equation (3.6). If the watermark's pixel is white (i.e. 1) then an additional factor  $\alpha$  is added to the DCT coefficient. If the pixel is black (i.e. 0) then  $\alpha$  is subtracted from the DCT coefficient.

$$D'_{u,v} = \begin{cases} D_{u,v} + \alpha, & \text{where } W_{x,y} = 1 \\ D_{u,v} - \alpha, & \text{where } W_{x,y} = 0 \end{cases} \quad (3.6)$$

Where  $D_{u,v}$  and  $D'_{u,v}$  are the selected DCT coefficient before and after embedding, respectively,  $\alpha$  is the addition factor, and  $W_{x,y}$  is the value of the watermark's pixel at location  $[x,y]$ .

As for watermark embedding, there are two commonly used options: some systems embed watermark blocks separately in different image frames and some embed all blocks in each and every single frame. As for the former case, the purpose is to detect tampering on the frames. It can be used to protect the integrity of the media. If a frame is tampered (i.e. intentionally or unintentionally modified) or dropped, the watermark portion from that frame cannot be successfully detected. This means that the watermark extraction fails, which is an indicator of tampering. In this system, our main aim is to be able to extract the watermark logo in any cases so that it can identify the owner. Thus, in this system, all watermark blocks are embedded in each and every single frame. Even if some frames are tampered or dropped with the aim of destroying copyright information, we will still be able to detect the logo in some other frames. It is not possible to tamper all frames in a real situation as it will destroy the host video quality as well.

After the embedding process, the next step is to convert the watermarked  $Y$  components to the spatial domain by using the IDCT defined in Equation (3.7).

$$Y'_{x,y} = \frac{1}{\sqrt{2N}} \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} C_u C_v D'_{u,v} \cos\left[\frac{(2y+1)v\pi}{2N}\right] \cos\left[\frac{(2x+1)u\pi}{2N}\right], \quad (3.7)$$

where  $N$  is 8 for the block size of  $8 \times 8$ ,  $Y'_{x,y}$  is the luminance  $Y$  of the watermarked pixel at location  $[x,y]$  of the block,  $D'_{u,v}$  is the watermarked DCT coefficient in row  $u$  and column  $v$ ,

$$C_u = \begin{cases} \frac{1}{\sqrt{2}}, & \text{for } u = 0 \\ 1, & \text{otherwise} \end{cases}, \quad C_v = \begin{cases} \frac{1}{\sqrt{2}}, & \text{for } v = 0 \\ 1, & \text{otherwise} \end{cases}.$$

As the next step, the watermarked frame is converted back to the RGB color space by using Equations (3.8) - (3.10). Only  $Y$  components of each frame are used for watermark embedding. However, after embedding, all color arrays are necessary to be combined to get full color watermarked frame.

$$R = \frac{298.08}{256}.Y + \frac{408.58}{256}.C_r - 222.92, \quad (3.8)$$

$$G = \frac{298.08}{256}.Y - \frac{100.29}{256}.C_b - \frac{208.12}{256}.C_r - 135.57, \quad (3.9)$$

$$B = \frac{298.08}{256}.Y - \frac{516.41}{256}.C_b - 276.84, \quad (3.10)$$

where  $Y$ ,  $C_b$ , and  $C_r$  are components of the YCbCr color space respectively.

Figure (3.4) shows the YCbCr to RGB conversion result obtained by using the MATLAB command “ycbcr2rgb”. The results are exactly the same as the manual calculation shown in Example (3.2).

```

>> YCbCr(:,:,1)
ans =
64x64 uint8 matrix
Columns 1 through 18
234 232 232 231 232 231 225 230
>> YCbCr(:,:,2)
ans =
64x64 uint8 matrix
Columns 1 through 18
127 128 130 130 130 130 134 131
>> YCbCr(:,:,3)
ans =
64x64 uint8 matrix
Columns 1 through 18
129 130 129 128 126 125 128 129
>> RGB(:,:,1)
ans =
64x64 uint8 matrix
Columns 1 through 18
255 255 253 250 248 246 243 251
>> RGB(:,:,2)
ans =
64x64 uint8 matrix
Columns 1 through 18
253 250 250 250 252 252 241 247
>> RGB(:,:,3)
ans =
64x64 uint8 matrix
Columns 1 through 18
252 252 255 254 255 254 255 255

```

Figure 3.4 YCbCr to RGB Calculation Result

### Example 3.2: RGB Calculation Example

For the pixel at row = 1 and column = 8,

$Y = 230$ ,  $C_b = 131$ ,  $C_r = 129$ ,

$$\begin{aligned} R &= \frac{298.08}{256} \cdot Y + \frac{408.58}{256} \cdot C_r - 222.92 \\ &= \frac{298.08}{256} \cdot 230 + \frac{408.58}{256} \cdot 129 - 222.92 \\ &= 250.77 \approx 251 \end{aligned}$$

$$\begin{aligned} G &= \frac{298.08}{256} \cdot Y - \frac{100.29}{256} \cdot C_b - \frac{208.12}{256} \cdot C_r - 135.57 \\ &= \frac{298.08}{256} \cdot 230 - \frac{100.29}{256} \cdot 131 - \frac{208.12}{256} \cdot 129 - 135.57 \\ &= 247.18 \approx 247 \end{aligned}$$

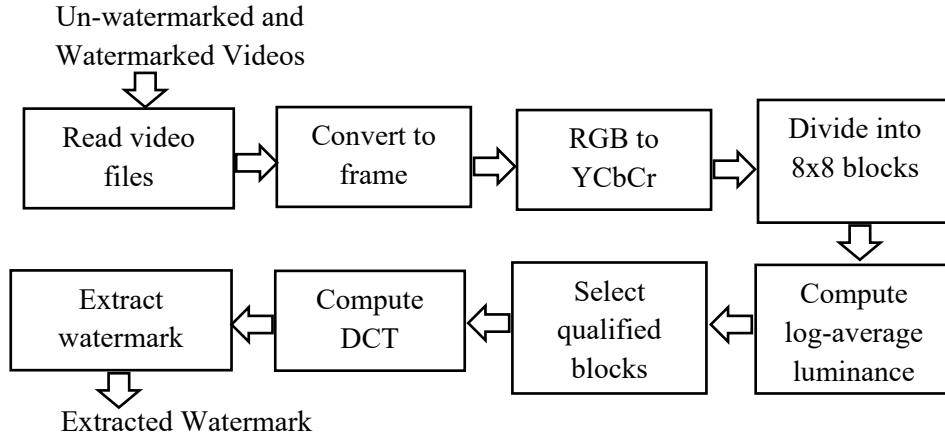
$$\begin{aligned} B &= \frac{298.08}{256} \cdot Y - \frac{516.41}{256} \cdot C_b - 276.84 \\ &= \frac{298.08}{256} \cdot 230 - \frac{516.41}{256} \cdot 131 - 276.84 \\ &= 255.22 \approx 255 \end{aligned}$$

All the above steps must be repeated for all frames. The aim of the proposed system is to identify the owner of the media in case copyright violation is detected. Thus, the watermark logo is repetitively embedded in all frames so that some frames can still be used for owner identification even if some are manipulated by malicious users. Finally, the watermarked frames are ordered to get the watermarked video. Visibility of the watermark can be checked once the watermarked video is obtained.

### 3.3 Watermark Extraction Process

Figure (3.5) shows the flow of the proposed watermark extraction process. All the first 7 steps are exactly the same as the embedding process. The watermark has to be extracted from the same blocks that have been used in the embedding process.

The last step of Figure (3.5) is the watermark extraction step. The proposed method needs the original un-watermarked video for the extraction process. To extract a watermark bit, the difference between the watermarked DCT coefficient and the un-watermarked coefficient is calculated, as stated in Equation (3.11). If the result is greater than or equal to zero, then the watermark color is assumed as white; otherwise, it is assumed as black. Equation (3.11) shows the extraction of a watermark bit from each pixel.



**Figure 3.5 Flow of the Proposed Watermark Extraction Process**

$$W'_{x,y} = \begin{cases} 1, & \text{where } D'_{u,v} - D_{u,v} \geq 0 \\ 0, & \text{where } D'_{u,v} - D_{u,v} < 0 \end{cases} \quad (3.11)$$

where  $W'_{x,y}$  is the extracted watermark pixel,  $D_{u,v}$  and  $D'_{u,v}$  are the DCT coefficients before and after embedding, respectively. After extracting the watermark bits from each pixel of all selected blocks, the  $64 \times 64$  watermark logo image can then be obtained.

In this system, the above-mentioned watermark extraction process is performed for all frames. Then, all the extracted watermarks are compared with the original watermark, and the one with the highest similarity is chosen as the final result to identify the owner.

Otherwise, in order to reduce the computational complexity, a predefined similarity threshold can be set. If the similarity between the extracted watermark from the current frame and the original watermark is greater than the threshold, this is assumed that the current watermark is satisfactory to identify the owner. Then, it is not necessary to perform the watermark extraction process for the next frames.

This chapter discussed the watermark embedding and extraction processes of the proposed system in detail. In the next chapter, will implement those processes in MATLAB and evaluate the performance of the proposed system by conducting experiments on example videos.

## CHAPTER 4

### EXPERIMENTAL RESULTS

This chapter evaluates the performance of the proposed luminance modification-based video watermarking system. The requirements of a good watermarking system depend on the application purpose. This system is intended to solve the copyright infringement issue of the digital video industry. Thus, the proposed watermarking system needs to be robust against common signal processing attacks, while maintaining acceptable visual quality. This chapter evaluates the robustness of the proposed system to compression, geometric operations, and common signal processing attacks by measuring the similarity between the original and detected watermarks from attacked video files. In addition, the visual quality of the watermarked video is measured in terms of peak signal-to-noise ratio (PSNR).

#### 4.1 Media Used for Experiments

The proposed watermarking system is simulated in Matlab R2017b and applied on a total of 30 videos (.mp4), mentioned in Table (4.1). Among them, 10 are “Nature” videos with views of the nature backgrounds, 10 are “Cartoon” videos with the movement of people, and 10 are “Music” videos with dance and singers.

**Table 4.1 Videos in MP4 Format Used for Experiments**

Name	Length	Size	Name	Length	Size	Name	Length	Size
Music1	16s	5.77MB	Nature1	16s	6.05MB	Cartoon1	5s	1.73MB
Music2	14s	2.14MB	Nature2	17s	6.47MB	Cartoon2	5s	1.15MB
Music3	10s	2.33MB	Nature3	17s	6.40MB	Cartoon3	5s	1.92MB
Music4	13s	5.07MB	Nature4	14s	3.42MB	Cartoon4	5s	2.10MB
Music5	10s	3.96MB	Nature5	16s	5.93MB	Cartoon5	5s	2.13MB
Music6	18s	6.90MB	Nature6	16s	6.17MB	Cartoon6	5s	2.08MB
Music7	17s	1.70MB	Nature7	13s	5.19MB	Cartoon7	5s	622 KB
Music8	13s	5.06MB	Nature8	18s	6.80MB	Cartoon8	5s	1.76MB
Music9	19s	6.12MB	Nature9	17s	6.60MB	Cartoon9	5s	2.08MB
Music10	15s	5.85MB	Nature10	20s	7.41MB	Cartoon10	5s	2.07MB

Figure (4.1) shows an image that is used as the copyright logo to be embedded as the watermark image. It can be any format of image. For the experiments discussed below, the “JPEG” format with 64×64 size is used.



Figure 4.1 Watermark Images: M Logo (Left), Lena (Right)

## 4.2 Evaluation on Watermark Invisibility

Watermark invisibility means the visual quality of the watermarked video. A good watermarking system should maintain the quality of the host video as high as possible. In this system, invisibility of the embedded watermark is analyzed in terms of the Mean Square Error (MSE), the Peak Signal to Noise Ratio (PSNR) and the Mean Opinion Score (MOS).

### 4.2.1 Mean Square Error (MSE)

Mean Squared Error (MSE) defined in Equation (4.1) and it is one of the simplest methods to measure the distortion. The lower the value, the better the result and 0 means the model is perfect.

$$MSE = \frac{1}{N \times M} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} (I(x,y) - I_w(x,y))^2, \quad (4.1)$$

where  $N \times M$  is the image size,  $I(x,y)$  and  $I_w(x,y)$  are the pixel values at location  $(x,y)$  of the original and watermarked images, respectively.

### 4.2.2 Peak Signal to Noise Ratio (PSNR)

The PSNR is widely used to measure the invisibility of the watermark. Typical PSNR values for lossy image and video compression is 30-50 dB, where the higher the better [46]. Optimal values for JPEG image transmission over wireless channels are considered to be about 20-25 dB [19]. In this system, it is assumed that the watermark invisibility is acceptable if the PSNR of the watermarked video is about 50 dB.



The PSNR is calculated as defined in Equation (4.2).

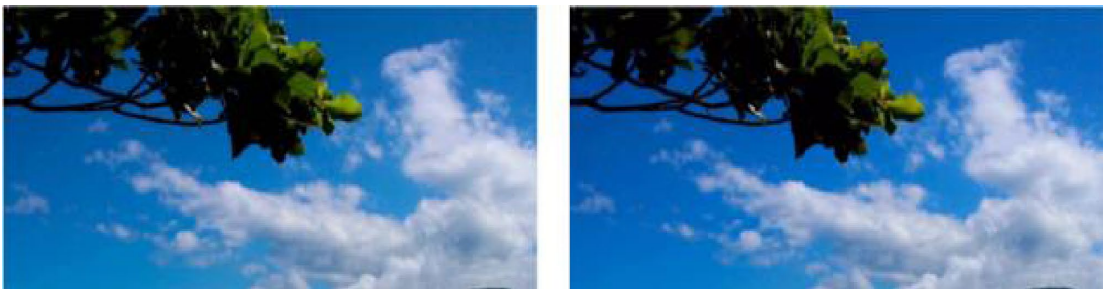
$$PSNR(dB) = 10 \log_{10} \left( \frac{(MAX_i)^2}{MSE} \right), \quad (4.2)$$

where  $MAX_i$  is the maximum possible pixel value of the image (video frame),  $MSE$  is the mean square error.

**Table 4.2 PSNR after Watermark Embedding with Different  $\alpha$  Value**

Video	<i>PSNR (dB) in Different <math>\alpha</math></i>				
	$\alpha=10$	$\alpha=5$	$\alpha=3$	$\alpha=1$	$\alpha=0.1$
Nature1.mp4	54.85	58.04	60.41	66.28	69.83
Music1.mp4	54.65	57.89	60.36	67.15	73.13
Cartoon1.mp4	54.62	56.98	58.15	66.35	69.75

In this system, visual quality of the watermarked video depends on the additional factor  $\alpha$  used in the embedding process. Table (4.2) shows the PSNR results after watermark embedding with different  $\alpha$  values for each video type. It can be seen that the smaller the  $\alpha$ , the higher the PSNR which means the better the visual quality of the watermarked video. Unfortunately, there is a tradeoff for the choice of  $\alpha$  for good robustness and good invisibility. The larger the  $\alpha$ , the more robust to malicious manipulation. In this system,  $\alpha = 5$  is chosen for keeping a balance of the acceptable robustness and invisibility. As an example, for checking the visibility of the watermark, Figure (4.2) - (4.4) comparatively shows the original and watermarked video frames extracted from the experiments.



**Figure 4.2 Original (Left) and Watermarked (Right) Video Frames for Nature1.mp4**



**Figure 4.3 Original (Left) and Watermarked (Right) Video Frames for Cartoon1.mp4**



**Figure 4.4 Original (Left) and Watermarked (Right) Video Frames for Music1.mp4**

Table (4.3) shows the MSE and PSNR results for all the 30 videos of Table (4.1) after embedding the watermark with  $\alpha = 5$ . It can be concluded from Table (4.3) that the proposed watermarking system achieves good watermark invisibility as the PSNR for all videos are greater than 50 dB. Table (4.3) also shows the time needed for watermark embedding. As the proposed system embeds the watermark in every single frame, the embedding time varies based on the frame size and number of frames consisting of the host video.

**Table 4.3 PSNR Results for 30 Different Videos with  $\alpha = 5$** 

<b>Name</b>	<b>MSE</b>	<b>PSNR (dB)</b>	<b>Embedding Time</b>
Music 1	0.137	57.336	1.09 mins
Music 2	0.164	56.155	0.74 mins
Music 3	0.224	54.734	1.21 mins
Music 4	0.221	54.838	1.54 mins
Music 5	0.203	55.291	1.24 mins
Music 6	0.232	54.595	2.07 mins
Music 7	0.043	63.742	0.73 mins
Music 8	0.237	54.499	1.38 mins
Music 9	0.179	55.872	1.95 mins
Music 10	0.250	54.323	1.67 mins
Nature 1	0.136	57.83	1.55 mins
Nature 2	0.137	57.075	1.67 mins
Nature 3	0.171	56.297	1.69 mins
Nature 4	0.191	55.618	1.39 mins
Nature 5	0.232	54.698	1.63 mins
Nature 6	0.160	56.246	1.76 mins
Nature 7	0.191	55.534	1.31 mins
Nature 8	0.167	56.028	1.8 mins
Nature 9	0.228	54.769	1.82 mins
Nature 10	0.219	54.833	2.76 mins
Cartoon 1	0.249	54.336	0.62 mins
Cartoon 2	0.223	54.75	0.68 mins
Cartoon 3	0.222	54.810	0.56 mins
Cartoon 4	0.193	55.447	0.52 mins
Cartoon 5	0.225	54.807	0.52 mins
Cartoon 6	0.228	54.625	0.79 mins
Cartoon 7	0.103	69.873	0.59 mins
Cartoon 8	0.184	55.656	0.59 mins
Cartoon 9	0.241	54.417	0.56 mins
Cartoon 10	0.185	55.589	0.55 mins

### 4.2.3 Mean Opinion Score (MOS)

A Mean Opinion Score (MOS) is a numerical measure of the human-judged overall quality of an image or video. The Mean Opinion Score is a ranking of the quality of voice and video in telecommunication system.

MOS is judged on a range of 1 (bad) to 5 (excellent), Mean Opinion Scores are the average of a number of random human-scored individual parameters. Nevertheless, Mean Opinion Scores were originally derived from comments of professional observers, today a MOS is often decided by an Objective Measurement Method approximating a human ranking. The levels of the Ranking are described in Table (4.4).

**Table 4.4 Mean Opinion Score (MOS) Ranking**

<b>MOS</b>	<b>Quality</b>	<b>Impairment</b>
5	Excellent	Imperceptible
4	Good	Perceptible but not annoying
3	Fair	Slightly annoying
2	Poor	Annoying
1	Bad	Very Annoying

The MOS is calculated as the arithmetic mean over single ratings performed by human subjects for a given stimulus in a subjective quality evaluation test as described in Equation (4.3).

$$MOS = \frac{\sum_{n=1}^N R_n}{N} \quad (4.3)$$

where  $R$  are the individual ratings for a given stimulus by  $N$  subjects.

Due to the human tendency to avoid perfect ratings (now reflected in the objective approximations), somewhere around 4.3 - 4.5 is considered an excellent quality target. On the low end, call or video quality becomes unacceptable below a MOS of roughly 3.5. With the choice of  $\alpha$  value 5, the evaluated result from 10 different observers are listed in Table (4.5). The results of MOS scores for 3 videos are above 4.5, so the watermarked videos can be considered as imperceptible.

**Table 4.5 MOS Ranking for 3 Different Video with  $\alpha = 5$**

	<b>Nature1.mp4</b>	<b>Music1.mp4</b>	<b>Cartoon1.mp4</b>
Observer 1	5	5	5
Observer 2	5	4	4
Observer 3	5	5	5
Observer 4	4	4	4
Observer 5	5	5	5
Observer 6	5	5	5
Observer 7	4	4	5
Observer 8	5	5	5
Observer 9	5	4	5
Observer 10	5	5	5
Avg MOS Score	4.8	4.6	4.8

### 4.3 Evaluation on Watermark Similarity

The main aim of the proposed system is to identify the media owner in case copyright violation is detected. Thus, it should be able to extract the embedded copyright logo no matter how severe the host video is manipulated by malicious users. This section analyzes the robustness of the proposed system to malicious attacks.

Robustness is measured by calculating the similarity between the embedded and extracted watermarks [34], as mentioned in Equation (4.4).

$$S_{x,y} = \begin{cases} 1, & \text{if } W'_{x,y} = W_{x,y} \\ 0, & \text{if } W'_{x,y} \neq W_{x,y} \end{cases} \quad (4.4)$$

where  $S_{x,y}$  is the similarity of each pixel,  $W'_{x,y}$  and  $W_{x,y}$  are the extracted and embedded watermark pixels, respectively. Then, similarity for the whole image ( $\sigma$ ) is calculated as shown in Equation (4.5). The 100% is the maximum similarity.

$$\sigma(\%) = 100 \times \frac{1}{N \times M} \sum_{x=0}^{N-1} \sum_{y=0}^{M-1} S_{x,y}, \quad (4.5)$$

where  $N \times M$  is the size of the watermark image (64×64).

Table (4.6) shows the similarity results calculated by comparing the original and extracted watermarks from the videos with no attack. Similarity may be changed based on the value of  $\alpha$  used. As for  $\alpha = 5$ , we can see from Table (4.6) that the similarity is 100% for all videos when no attack occurs. Table (4.6) also shows the time needed for watermark extraction.

**Table 4.6 Similarity Results for Extracted Watermarks from 30 Different Videos (Before Attack)**

<b>Original Video</b>	<b>Watermarked Video</b>	<b>Extraction Time</b>	<b>Similarity</b>
Music1	WMMusic1	0.66 mins	100%
Music2	WMMusic2	0.47 mins	100%
Music3	WMMusic3	0.85 mins	100%
Music4	WMMusic4	1.51 mins	100%
Music5	WMMusic5	0.56 mins	100%
Music6	WMMusic6	1.66 mins	100%
Music7	WMMusic7	0.56 mins	100%
Music8	WMMusic8	0.55 mins	100%
Music9	WMMusic9	1.61 mins	100%
Music10	WMMusic10	1.51 mins	100%
Nature1	WMNature1	0.56 mins	100%
Nature2	WMNature2	0.68 mins	100%
Nature3	WMNature3	0.61 mins	100%
Nature4	WMNature4	1.14 mins	100%
Nature5	WMNature5	1.54 mins	100%
Nature6	WMNature6	1.24 mins	100%
Nature7	WMNature7	0.59 mins	100%
Nature8	WMNature8	0.56 mins	100%
Nature9	WMNature9	0.52 mins	100%
Nature10	WMNature10	1.13 mins	100%
Cartoon1	WMCartoon1	1.06 mins	100%
Cartoon2	WMCartoon2	0.74 mins	100%
Cartoon3	WMCartoon3	1.21 mins	100%
Cartoon4	WMCartoon4	0.79 mins	100%

Cartoon5	WMCartoon5	0.59 mins	100%
Cartoon6	WMCartoon6	0.62 mins	100%
Cartoon7	WMCartoon7	1.09 mins	100%
Cartoon8	WMCartoon8	0.73 mins	100%
Cartoon9	WMCartoon9	0.59 mins	100%
Cartoon10	WMCartoon10	0.56 mins	100%
<b>Average</b>		<b>0.883 mins</b>	<b>100%</b>

In this system, the watermarks are extracted from every single frame. Then, the watermark with the highest similarity is chosen to prove the copyright. Thus, the watermark extraction time is a bit longer compared with the duration of the host video. However, in order to improve the watermark extraction time, a similarity threshold can be set. Once a frame with the similarity of the extracted watermark greater than threshold is detected, no more frames are needed for watermark extraction. In this way, the extraction time can be reduced to some extent.

The following subsections discuss the signal processing attacks that are applied on the watermarked video for testing the robustness of the proposed system.

#### **4.3.1 Attacks on Watermarking Methods**

Watermarked videos may face a variety of unintended or intended attacks trying to destroy the copyright information. A good watermarking method should resist those attacks. In this system, the following attacks are simulated in Matlab for robustness testing. For all those attacks, parameters are chosen by guessing from the attacker perspective.

1. Compression schemes
2. Geometric operations such as cropping and rotation
3. Signal processing operations such as quantization, filtering, and noise addition

##### **4.3.1.1 Compression Attack**

Compression attack may occur when users intentionally compress to distort the watermark or unintentionally compress to reduce the video file size.

The proposed luminance-based watermarking system is implemented on the DCT, which is a basis of the JPEG image compression, and thus it well resists the

compression attack. For simulation of this attack, the parameter “compression ratio” is defined as the storage size of the original image divided by the storage size of the compressed image. For example, if the video is originally 1000 bytes and needed to be 500 bytes then the ratio would be 2 [64]. Figure (4.5) shows the extracted watermark after compressing the “WMNature1.mp4” with compression ratio of 2. Both achieve the similarity result of 100%.



**Figure 4.5 Extracted Logos after Compression Attack**

#### **4.3.1.2 Cropping Attack**

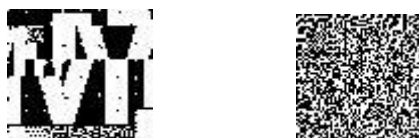
Cropping attack occurs when the video is unintentionally or intentionally cropped to remove some portion. The attack parameter is the desired frame size. As an example, Figure (4.6) shows the extracted watermark after cropping the 640x480 frames of the “WMNature1.mp4” to 620x460. It achieves 62.01% and 67.29% similarity respectively.



**Figure 4.6 Extracted Logos after Cropping Attack**

#### **4.3.1.3 Rotation Attack**

Rotation attack rarely occurs when the users rotate the video for some reason like video editing. Attack parameters can be any angle value such as 90° and 180°; positive value for counterclockwise and negative value for clockwise directions. Figure (4.7) shows the extracted watermark from the 180° clockwise rotated video of “WMNature1.mp4”, which is upside down after the attack. It achieves 62.11% and 52.37% similarity respectively.



**Figure 4.7 Extracted Logos after Rotation Attack**



#### 4.3.1.4 Quantization Attack

Quantization attack occurs when reducing the number of colors required to represent a digital image to reduce its file size. Attack parameter is the quantization level specified in an  $N$  element vector. The quantized image contains  $N + 1$  discrete integer values in the range 1 to  $N + 1$  [36]. For example, to get 4 discrete levels,  $N$  needs to be 3. Figure (4.8) shows the extracted result from a 4-level quantized video of “WMNature1.mp4”, which achieves 79.43% similarity and 64.92% respectively. The original video was 256-level quantized.



Figure 4.8 Extracted Logos after Quantization Attack

#### 4.3.1.5 Filtering Attack

Filtering attack occurs when users perform video editing for quality enhancement or intentionally filter the video to distort the watermark. Motion filter is used in this attack, which results the blurring artifacts in the filtered video [35]. Two attack parameters, *lens* and *theta*, specify the length of the motion and the angle of motion in degrees in a counterclockwise direction. The default *lens* is 9 and the default *theta* is 0, which corresponds to a horizontal motion of pixels [28]. Figure (4.9) shows the result from the filtered video “WMNature1.mp4” with attack parameters of 10 for *lens* and 10 for *theta*. The similarity result are 59.06% and 67.29% respectively.



Figure 4.9 Extracted Logos after Filtering Attack

### 4.3.1.6 Adding Noise

Noise addition attack is one of the most commonly occurring attacks in digital watermarking as the communication channel is noisy in nature. It can also occur when the users perform video editing. Additive white Gaussian noise (AWGN) is a basic noise model used in information systems [29]. The noise density is an important attack parameter. Figure (4.10) shows the result after the AWGN attack on “WMNature1.mp4” with noise density of 0.01. The similarity result are 69.65% and 60.38% respectively.















**Figure 4.10 Extracted Logos after Adding Noise Attack**

Table (4.7) summarizes the average similarity scores of all 30 videos for all kinds of attack.

**Table 4.7 Average Similarity Scores for Various Attacks**

Attack Type	Attack Parameter	PSNR (dB)	Average Similarity	Extracted Logo	Average Similarity	Extracted Logo
No attack	-	57.01	100%		100%	
Compression	2	56.18	100%		100%	
Compression	4	56.18	100%		100%	
Cropping	620 x 340	31.10	62.01%		67.29%	
Cropping	600 x 300	30.79	58.57%		58.11%	
Rotation	90	26.41	58.84%		52.17%	
Rotation	180	27.46	62.11%		52.37%	

Quantization	3	34.50	79.43%		64.92%	
Quantization	7	39.45	80.59%		64.18%	
Filtering	5,5	38.98	59.08%		67.29%	
Filtering	10,10	36.62	59.06%		67.29%	
Noise	0.01	30.44	69.65%		60.38%	
Noise	0.02	31.14	69.46%		60.38%	

Each attack is simulated with two attack parameters and the PSNR values show how severe the attacks are. As it can be seen in Table (4.7), the similarity is 100% when there is no attack and drops to 58% as the lowest after attacks. As the nature of attacks are different, it can also be seen that the extracted logo from “rotation” attack with 62.11% similarity is worse to see than the logo from “cropping” attack with 58.57% similarity. In general, it can be seen that 58% similarity is acceptable and extracted logos are still recognizable with human eyes, except for rotation attack. By the extracted watermarks result, it can also conclude that watermark logo or information should be clear and easily recognizable. Thus, it can be concluded that the proposed method achieves good robustness to most signal processing attacks to some acceptable extent.

## CHAPTER 5

### CONCLUSION

With the advancement of technology these days, owners of digital media are facing copyright infringement problems. In order to prevent copyright infringements, efficient owner identification systems must be developed and the watermarking methods are becoming handy. In watermarking systems, the owner embeds copyright related information or owner identification marks to the host media before distribution.

In such kinds of watermarking systems deployed for copyright protection, the robustness of the embedded watermark is very important. It should resist against intended or unintended signal processing attacks. Once the copyright violation is detected, the owner should be able to extract the copyright information from the attacked/manipulated videos and prove his/her ownership.

This thesis presented a digital video watermarking system in the DCT domain based on the luminance modification process. The proposed system is intended to solve the copyright infringement issues facing the digital video industry. A binary logo image was embedded as copyright information in the host video, which could be any type such as “AVI” or “MP4”. The proposed embedding process only changed the luminance value of the host video frames. The human visual system is not sensitive to the light intensity (luminance) changes. Hence, the proposed system could well preserve good video quality. In addition, as the proposed method was developed in the DCT domain, it achieved good robustness to a certain extent for both intended and unintended attacks. Therefore, it can be concluded that the proposed system is applicable in copyright protection applications.

#### 5.1 Discussion

As per the experimental results,  $\alpha$  value decide the quality of the watermarked video that needs to be chosen in the embedding process. After the experiments with different  $\alpha$  value, result can be considered as the smaller the  $\alpha$ , the higher the PSNR. Higher PSNR means better quality of the watermarked video. As a tradeoff for the choice of  $\alpha$ , system also needs to robust against malicious attack. The larger the  $\alpha$ , the more robust to severe attacks. So,  $\alpha$  value needs to be chosen carefully to keep the

balance between robustness and invisibility. After embedding the watermark with  $\alpha = 5$ , PSNR results for all tested videos are greater than 50 dB, and the results of MOS scores are above 4.5. So, the system can be considered as imperceptible.

The watermarks are embedded in every frame in order to prevent frame loss during transmission. As a tradeoff, watermark needs to be extracted from every single frame. To prove the copyright issue, only the highest similarity watermark needs to be chosen. In order to improve the extraction time, a similarity threshold can be set. If the threshold is obtained, no more frames are needed to be extracted.

After extracting the watermark from watermarked video before attack, the watermark similarity is 100%. The similarity drops to 58% as lowest after geometric attacks like Cropping and Rotation. Even though facing the severe attacks, the system can still prove the ownership of video because the similarity above 50% is considered as acceptable and extracted logos are still recognizable with human eyes. Experimental results also stated that it is important to choose the watermark logo that is clear and easily recognizable. The system proved remarkable robustness against Digital Signal Processing attacks like Quantization, Filtering and Adding Noise as it is implemented in frequency domain. After signal processing attacks, extracted logos are easily recognizable even before the similarity measurement. And the use of DCT domain to embed the watermark, the system proved that watermark logo can resist compression attack as its similarity result is 100%. The system has proved that it is efficient enough to protect the copyright infringement and can be used as commercial product.

## **5.2 Further Extension**

This thesis intends to develop an effective video watermarking system in a security environment. The robustness and watermark invisibility of the method are efficient to use in copyright protection in real world digital distribution. However, the time needed for watermark embedding and extraction processes take long depending on the frame size and video length, and somewhat degrades the effectiveness of the system. As further extension, it can figure out and try different ways of improving the execution time of the system. In addition, current system is implemented for 2D video based on 2D DCT calculation, so the research can be extended for 3D video watermarking by using the 3D DCT method. Also, the system can be extended by

changing the embedding style like embedding different images in each frame, embedding only some pixels in each frame. And many further extensions can be added for better performance and high security.

## REFERENCES

- [1] A. Abdullah, "Advanced Encryption Standard (AES) Algorithm to encrypt and decrypt data," Jun 2017.
- [2] A. Agarwal, P. Raj, and S. Katiyar, "Secured audio encryption using AES algorithm," *International Journal of Computer Applications*, vol. 178, pp. 29-33, 2019.
- [3] A. Arab, M. J. Rostami, and B. Ghavami, "An image encryption method based on chaos system and AES algorithm," *The Journal of Supercomputing* 75, pp. 6663–6682, 2019.
- [4] A. Bashir, A. S. Hasan, and H. Almangush, "A new image encryption approach using the integration of a shifting technique and the AES algorithm," *International Journal of Computer Applications*, vol. 42, no. 9, pp. 38-45, March 2012.
- [5] A. Chadha, S. Mallik, A. Chadha, "Dual-Layer Video Encryption using RSA Algorithm," *International Journal of Computer Applications*, Vol.1, Apr 2015.
- [6] A. Chadha, S. Mallik, A. Chadha, "Dual-Layer Video Encryption using RSA Algorithm," *International Journal of Computer Applications*, Vol.1, Apr 2015.
- [7] A. Watson, "Image compression using the Discrete Cosine Transform," *NASA Ames Research Center, Mathematica Journal*, pg. 81-88, Jan 1994.
- [8] Advanced Video Coding," retrieved at:  
[https://en.wikipedia.org/wiki/Advanced\\_Video\\_Coding](https://en.wikipedia.org/wiki/Advanced_Video_Coding)
- [9] A. Singh, M. Dave and A. Mohan, "Multilevel Encrypted Text Watermarking on Medical Images Using Spread-Spectrum in DWT Domain," *in Wireless Personal Communications*, Aug 2015.
- [10] A. Tefas, I. Pitas, "Image Watermarking," *The Essential Guide to Image Processing*, Jan 2009.
- [11] B. Reddy, A. Jadhav, "Visible and Invisible Image Watermarking," *International Journal of Engineering Research & Technology*, APR 2018.
- [12] C. Phin, N. Rahman, N. Pa, "A Digital Image Watermarking System: An Application of Dual Layer Watermarking Technique," *International Journal on Informatics Visualization*, NOV 2017.
- [13] C. Hsu, and J. Wu, "DCT-BASED WATERMARKING FOR VIDEO," *IEEE Transactions on Consumer Electronics*, Vol. 44, No. 1, FEB 1998.
- [14] C. Cruz, J. Mendoza, "Semi-Fragile Watermarking Based Image Authentication with Recovery Capability," *Conference: Information Engineering and Computer Science*, Jan 2010.

- [15] Color models," retrieved at:  
[https://scc.ustc.edu.cn/zlsc/sugon/intel/ipp/ipp\\_manual/IPPI/ippi\\_ch6/ch6\\_color\\_models.htm](https://scc.ustc.edu.cn/zlsc/sugon/intel/ipp/ipp_manual/IPPI/ippi_ch6/ch6_color_models.htm)
- [16] Color models," retrieved at:  
[https://scc.ustc.edu.cn/zlsc/sugon/intel/ipp/ipp\\_manual/IPPI/ippi\\_ch6/ch6\\_color\\_models.htm](https://scc.ustc.edu.cn/zlsc/sugon/intel/ipp/ipp_manual/IPPI/ippi_ch6/ch6_color_models.htm)
- [17] D. I. G. Amalarethinam and J. S. Geetha, "Image encryption and decryption in public key cryptography based on MR," 2015 International Conference on Computing and Communications Technologies (ICCCT), pp. 133-138, 2015.
- [18] Dolley Shukla, "Watermarking Schemes for Copy Protection: A Survey," International Journal of Computer Science & Engineering Survey, Feb 2012.
- [19] E. Mohsen, Abouelazm, and Prof. Hassan, "JPEG image transmission over mobile network with an efficient channel coding and interleaving," Taylor & Francis Group, 2012.
- [20] Er. Sonia, Er. Naresh Kumar Garg, Er. Gurvinder Singh, "A Survey on Digital Image watermarking," International Journal of Advanced Research in Computer Engineering & Technology (IJARCET), Vol. 3, Issue 6, Jun 2014.
- [21] G. Kalra, "DCT and thresholding based digital video watermarking," in Proc. IEEE International Conference on Image Processing, May 2012.
- [22] G. Rajkumar, V. Malemath, "Video steganography: secure data hiding technique," International Journal of Computer Network and Information Security, Sep 2017.
- [23] G. Rajkumar, V. Malemath, "Video steganography: secure data hiding technique," International Journal of Computer Network and Information Security, Sep 2017.
- [24] H.261: video codec for audiovisual services at p x 64 kbit/s," retrieved at:  
<https://www.itu.int/rec/T-REC-H.261-199303-I/en>
- [25] Hai Tao, Li Chong Min, "Robust Image Watermarking Theories and Techniques: A Review," Journal of Applied Research and Technology, Volume 12, Issue 1, Feb 2014.
- [26] High Efficiency Video Coding," retrieved at:  
[https://en.wikipedia.org/wiki/High\\_Efficiency\\_Video\\_Coding](https://en.wikipedia.org/wiki/High_Efficiency_Video_Coding)
- [27] Hsu, Chiou-Ting, Wu, Ja-Ling, "Image Watermarking by Wavelet Decomposition", in Proc. IEEE International Conference on Image Processing, Jan 2000.
- [28] <http://matlab.izmiran.ru/help/toolbox/images/fspecial.html>



- [29] [https://en.wikipedia.org/wiki/Additive\\_white\\_Gaussian\\_noise](https://en.wikipedia.org/wiki/Additive_white_Gaussian_noise)
- [30] [https://en.wikipedia.org/wiki/Discrete\\_cosine\\_transform](https://en.wikipedia.org/wiki/Discrete_cosine_transform)
- [31]  
<https://www.cs.bham.ac.uk/~mdr/teaching/modules03/security/students/SS5/Steganography.html>
- [32] H. Huang, "An adaptive video watermarking technique based on DCT domain," in the Multidisciplinary Digital Publishing Institute Scientific Conference, Jun 2014.
- [33] H. Nyeem, W. Boles and C. Boyd, "Digital image watermarking: its formal model, fundamental properties and possible attacks," EURASIP Journal on Advances in Signal Processing, Aug 2014.
- [34] I. Amer, T. Sheha, W. Badawy, and G. Jullien, "A Tool for Robustness Evaluation of Image Watermarking Algorithms," In Advanced Techniques in Computing Sciences and Software Engineering. Springer, 2010.
- [35] J. Lee, "Analysis of Attacks on Common Watermarking Techniques," IEEE Electrical and Computer Engineering Department University of British Columbia.
- [36] J. Woods, in Multidimensional Signal, Image, and Video Processing and Coding (Second Edition), 2012.
- [37] Kavitha KJ, B Priestly Shan, "Video Watermarking Using DCT and DWT: A Comparison", European Journal of Advances in Engineering and Technology Conference, 2(6): 83-87, May 2015.
- [38] Kenneth L. Levy, "Watermark payload encryption for media including multiple watermarks," Publication of US8127137B2, Feb 2018.
- [39] Khaled Mahmoud, S. Datta and James Flint, "Frequency Domain Watermarking: An Overview," ResearchGate Publication\_220413592, Jan 2005.
- [40] L. Stošić, M. Bogdanović, "RC4 stream cipher and possible attacks on WEP," International Journal of Advanced Computer Science and Applications, vol. 3, no. 3, Jan 2012.
- [41] Lalit Kumar Saini, Vishal Shrivastava, "A Survey of Digital Watermarking Techniques and its Applications," International Journal of Computer Science Trends and Technology (IJCTST), Jun 2014.
- [42] M. Ghanbari, "Standard codecs: image compression to advanced video coding," Institution of Engineering and Technology, pp. 1–2. ISBN 9780852967102, 2003.
- [43] M. Kumar and D. Shukla, "DCT domain video watermarking technique for AVI video," vol. 3, Apr 2015.

- [44] Mahbuba Begum, Mohammad Shorif Uddin, "Digital Image Watermarking Techniques: A Review," <https://doi.org/10.3390/info11020110>, Feb 2020.
- [45] MPEG-4 Part 2," retrieved at: [https://en.wikipedia.org/wiki/MPEG-4\\_Part\\_2](https://en.wikipedia.org/wiki/MPEG-4_Part_2)
- [46] M. Sharma, A. Tiwari, "A Survey of Transform Domain based Semi-fragile Watermarking Schemes for Image Authentication," 2012.
- [47] N. Ahmed, T. Natarajan, and K. R. Rao, "Discrete Cosine Transform," in IEEE Transactions on Computers, vol. C-23, no. 1, pp. 90-93, Jan. 1974.
- [48] P. Sharma, A. Shrivastava, and K. Chaturvedi, "High capacity audio steganography with RSA encryption," International Journal of Scientific Progress and Research, vol. 62, no. 1, August 2019.
- [49] Paul Nii Tackie Ammah, Ebenezer Owusu, "Robust medical image compression based on wavelet transform and vector quantization," ResearchGate Publication\_332451766, Apr 2019.
- [50] R. Popa, *An Analysis of Steganographic Techniques*, [http://ad.informatik.uni-freiburg.de/mitarbeiter/will/dlib\\_bookmarks/digital-watermarking/popa/popa.pdf](http://ad.informatik.uni-freiburg.de/mitarbeiter/will/dlib_bookmarks/digital-watermarking/popa/popa.pdf), 1998.
- [51] Ratnakirti, Tauheed Ahmed, "Watermarking through Image Geometry Change Tracking," Journal of Applied Research and Technology, Volume 2, Issue 2, Jun 2018.
- [52] Reza Safabakhsh, Shiva Zaboli, Arash Tabibiazar, "Digital watermarking on still images using wavelet transform," Conference: Information Technology: Coding and Computing, May 2004.
- [53] Rua-Huan Tsaih, Tzu-Shian Han, "Managing Innovation and Cultural Management in the Digital Era: The case of the National Place Museum," ISBN 9781138885141 (Hardbook), ISBN 9781315715643 (ebook), 2016.
- [54] S. Ambadekar, J. Jain, and J. Khanapuri, "Digital image watermarking through encryption and DWT for copyright protection," ISBN: 978-981-10-8862-9, Jan 2017.
- [55] S. Ambadekar, J. Jain, and J. Khanapuri, "Digital image watermarking through encryption and DWT for copyright protection," ISBN: 978-981-10-8862-9, Jan 2017.
- [56] S. Singh, S. Maakar, and S. Kumar, "Enhancing the security of DES algorithm using transposition cryptography techniques," International Journal of Advanced Research in Computer Science and Software Engineering, ISSN: 2277 128X, vol. 3, issue 6, Jun 2013.
- [57] S. Yadav and P. Anand, "DCT based digital video watermarking using MATLAB/Simulink," in Proc. IEEE International Conference on Image Processing, vol.3, Feb 2015.

- [58] S. Mousavi, A. Naghsh and A. Bakar, "Watermarking Techniques used in Medical Images: A Survey," *Journal of Digital Imaging*, May 2014.
- [59] T. Ono, "Technique of embedding and detecting digital watermark files," in *Proc. IEEE International Conference on Image Processing*, US7123741 B2, Oct 17, 2006.
- [60] Video coding format," retrieved at:  
[https://en.wikipedia.org/wiki/Video\\_coding\\_format#cite\\_note-1](https://en.wikipedia.org/wiki/Video_coding_format#cite_note-1)
- [61] Video," retrieved at: [https://en.wikipedia.org/wiki/Video#cite\\_note-1](https://en.wikipedia.org/wiki/Video#cite_note-1)
- [62] W. Chen and C. Ding, "Research on real-time video encryption algorithm based on moving objects," *The Open Cybernetics & Systemics Journal*, pp. 768-772, 2014.
- [63] W. Chen and C. Ding, "Research on real-time video encryption algorithm based on moving objects," *The Open Cybernetics & Systemics Journal*, pp. 768-772, 2014.
- [64] X. Wang, A. Matin, "Video encryption/compression using compressive coded rotating mirror camera," 2021.
- [65] X. Yu, C. Wang and X. Zhou, "A Survey on Robust Video Watermarking Algorithms for Copyright Protection," in *Multidisciplinary Digital Publishing Institute Scientific Conference*, Vol. 3, applsci-08-01891, 11 Oct 2018.
- [66] Z. Putra, K. Jassim, and A. Nsaif, "Hybrid cryptography and steganography method to embed encrypted text message within image," *International Conference on Computer Science and Engineering*, Conference Series 1339, Jan 2019.

## **PUBLICATION**

[1] May Tharaphy Htun and Twe Ta Oo, “An Efficient DCT-Based Video Watermarking Method for Copyright Control,” University of Computer Studies, Yangon, Myanmar, 2021.