



**UNIVERSITY OF NAIROBI**  
**SCHOOL OF ENGINEERING**  
**DEPARTMENT OF ELECTRICAL AND INFORMATION**  
**ENGINEERING**

**A ROBUST VIDEO WATERMARKING APPROACH BASED ON**  
**SINGULAR VALUE DECOMPOSITION AND WAVELET**  
**TRANSFORM**

**BY**

**ADUL, VINCENT OTIENO**

**F56/71321/2011**

A Thesis submitted in partial fulfillment of the requirements of the degree  
of Master of Science in Electrical and Electronic Engineering of the  
University of Nairobi

**AUGUST 2021**

# UNIVERSITY OF NAIROBI

## DECLARATION OF ORIGINALITY FORM

**NAME OF STUDENT:** ADUL, VINCENT OTIENO

**REGISTRATION NUMBER:** F56/71321/2011

**FACULTY/ SCHOOL/ INSTITUTE:** Engineering

**DEPARTMENT:** Electrical & Information Engineering

**COURSE NAME:** Master of Science in Electrical & Electronic Engineering

**TITLE OF WORK:** A Robust Video Watermarking Approach based on Singular Value Decomposition and the Wavelet transform

- 1) I understand what plagiarism is and I am aware of the university policy in this regard.
- 2) I declare that this research proposal is my original work and has not been submitted elsewhere for examination, award of a degree or publication. Where other people's work or my own work has been used, this has properly been acknowledged and referenced in accordance with the University of Nairobi's requirements.
- 3) I have not sought or used the services of any professional agencies to produce this work
- 4) I have not allowed, and shall not allow anyone to copy my work with the intention of passing it off as his/her own work
- 5) I understand that any false claim in respect of this work shall result in disciplinary action, in accordance with University anti- plagiarism policy

**SIGNATURE:**



**Date: 26<sup>TH</sup> AUGUST 2021**

## DECLARATION

This thesis is my original work and has not been presented for the award of any degree in any other University

ADUL, VINCENT OTIENO

REGISTRATION NUMBER: F56/71321/2011

SIGNATURE:



Date: 26<sup>TH</sup> AUGUST 2021

This thesis has been submitted for examination with my approval as a University Supervisor.

Prof Elijah Mwangi

Signature.....*Elijah Mwangi*.....  
Date.....*26<sup>th</sup> August 2021*.....

## **ACKNOWLEDGEMENT**

I would like to thank Prof Elijah Mwangi of the Department of Electrical & Electronic Engineering of the University of Nairobi and Mr. Felix Owalla for their invaluable assistance, helpful comments, and ideas in the execution of these works. I would also like to thank late Prof Maurice Mang'oli the guidance he provided, and Prof Vitalis Oduol, Prof Heywood Absalom Ouma and Dr. George Kamucha their constant encouragement during the course work as part of the degree programme.

Finally, I would like to thank my family for their support throughout my studies, and especially my son Henry, whom we had to share a study place as he continued with his undergraduate studies at the School of Built Environment in the University of Nairobi.

## ABSTRACT

The rapid evolution of technologies and the pervasive use of multimedia content has been occasioned by the prevalence of digital videos. The contents contained in these multimedia platforms have penetrated all aspects of people's lives creating opportunities for substantial numbers of pirated videos flooding the Internet, drastically infringing the copyright creators and frustrating the innovative impetus of the Multimedia industry. One of the solutions that has emerged in addressing this challenge among others has been the recourse to use of digital content watermarking techniques. This work was undertaken to demonstrate the robustness of two digital watermarking techniques using test videos in environments that were subjected to various attacks. The process involved carrying out simulations that compared the performance of Single Value Decomposition (SVD) against a hybrid of itself with Discrete Wavelet Transform (SVD/DWT) within the MATLAB/SIMULINK environment. The quality of the extracted watermarked messages was assessed using, PSNR, SSIM and Normalized cross correlations after the video frames were subjected to various attacks during experimentations. . The test video frames sizes ranged from 64x64, to 512x512 pixels (8-bit) of the 'Bus', 'Foreman', 'Carphone' and 'Container', 'Coastguard' and 'Akiyo' and the watermark used was an 8-bit 128x128 IEEE logo. The SVD/DWT hybrid had superior results than those of SVD on Median filtering and Histogram equalization. In other experiments, the SVD/DWT techniques was used as the framework for watermarking where the video frames were subjected to Gaussian, impulse noise amongst others. The results indicated while the Gaussian noise attacked frames reflected deterioration of the watermark with increase in the noise standard deviation, those of the JPEG compression showed that even at 90% degradation the logo was still legible. The impulse noise attack results one the other hand indicated resilience on the part of the message when the noise density was varied tenfold. Validation of the results of the algorithms used SVD/DWT Hybrid techniques against related works, showed superior and or comparable values with recent research efforts. The conclusion was that the results of the research investigations indicate the algorithms presented performed as per the set objectives.

# TABLE OF CONTENTS

DECLARATION .....	i
ACKNOWLEDGEMENT .....	ii
ABSTRACT.....	iii
ABBREVIATIONS AND ACRONYMS .....	iv
LIST OF FIGURES .....	viii
LIST OF TABLES.....	ix
CHAPTER 1 .....	1
INTRODUCTION .....	1
1.0 Background to the Study.....	1
1.1 Information Hiding Technologies.....	1
1.2 Digital Watermarking Properties .....	3
1.3 Applications of Digital video watermarking.....	5
1.4 Problem Statement .....	6
1.5 Justification of the Study .....	7
1.6 Objectives .....	7
1.7 Scope of Work .....	7
1.8 Thesis Organization .....	8
1.9 Note on Publications .....	8
CHAPTER 2 .....	9
LITERATURE REVIEW .....	9
2.1 Introduction.....	9
2.2 General Digital Watermarking Techniques .....	9
2.3 Digital Video Watermarking Methods.....	11

2.4 Spatial Domain Watermarking.....	11
2.5 Frequency Domain Watermarking.....	12
2.6 Singular Value Decomposition.....	13
2.7 Discrete Wavelet Transform Watermarking.....	16
2.8 Hybrid Transform-Based Watermarking Algorithms.....	18
2.9 Video watermarking.....	18
2.10 Related works.....	20
2.12 Knowledge Gaps.....	22
CHAPTER 3 .....	24
METHODOLOGY .....	24
3.1 Developed Watermarking Scheme.....	24
3.2 Materials .....	24
3.3 Methodology for Experimental Simulations.....	25
3.4 MATLAB/Simulink SVD Watermarking.....	27
3.6 Hybrid SVD/DWT Watermarking Technique.....	30
3.7 Metrics for Visual Quality .....	33
CHAPTER 4 .....	35
RESULTS AND DISCUSSIONS.....	35
4.1. Introduction to Simulations.....	35
4.2 Experimental Results .....	36
4.3. Signal processing Attacks .....	37
CHAPTER 5 .....	47
CONCLUSION AND RECOMMENDATIONS.....	47
5.1 Evaluation of the Study.....	47
5.2 Scope for Future Work.....	47

REFERENCES .....	49
APPENDIX A: VIDEO DATA USED IN SIMULATIONS.....	55
APPENDIX B: MATLAB PROGRAMS .....	56
APPENDIX C: PUBLICATION .....	80



## ABBREVIATIONS AND ACRONYMS

AVC	Advanced Video Codec
CIE	Commission International d'Eclairage (International Commission on Illumination)
CIF	Common Intermediate Format
DCT	Discrete Cosine Transform
DFT	Discrete Fourier Transform
DRM	Digital Rights Management
DWT	Discrete Wavelet Transform
GHz	Giga Hertz
HVS	Human Visual System
JAWS	Just Another Watermarking System
JPEG	Joint Picture Expert Group
LSB	Least Significant Bit
MB	Mega Byte
MRI	Magnetic Resonance Imaging
MSE	Mean Squared Error
NC	Normalized Correlation
PSNR	Peak Signal Noise Ratio
QCIF	Quarter Common Intermediate Format
RAM	Random Access Memory
RGB	Red, Green and Blue
SD	Spatial Domain
SSIM	Structured Similarity Index Quality Measure
SVD	Singular Value Decomposition
UIQI	Universal Image Quality Index
YCbCr	Luma(Y), Blue difference, Red Difference
YUV	Luma (Y), Chrominance (UV)

## LIST OF FIGURES

Figure 1.1 Data Hiding Technologies	2
Figure 1.2: A generic Steganography System	3
Figure 1.3: Cryptography System	3
Figure 2.1: Simple watermarking scheme	10
Figure 2.2: Digital video watermarking Techniques	11
Figure 2.3 Basic block diagrams of watermarking in the frequency domain	12
Figure 2.4: The Haar wavelet basis function	17
Figure 2.5: Examples of various Wavelets	18
Figure 2.6: Different attacks on watermark	22
Figure: 3.1: A 256x256 IEEE trademark	25
Figure 3.2: Watermark embedding model in Simulink	26
Figure 3.3: Watermark extraction model in Simulink	26
Figure 3.4: Illustration for the SVD process	29
Figure 3.5 Block diagram of SVD/DWT Watermark embedding process	32
Figure. 3.6: Block diagram SVD/DWT Watermark extracting process	33
Figure 4.1: Results of Histogram Equalization Attack	37
Figure 4.2: Results of Median Filtering Attack	38
Figure 4.3 Results of Gaussian Noise Attack	40
Figure 4.4: Results of extracted Message after JPEG compression	44

## LIST OF TABLES

Table 4.1: First set of results: SVD and SVD/DWT Extracted Logo without attacks	36
Table 4.2: Second set of results: SVD and SVD/DWT Extracted Logo without attacks	36
Table 4.3: Results for Histogram equalization Attack	38
Table 4.4: Results Median Filtering Attack	39
Table 4.5: Results for extracted watermark after Gaussian Noise Attack	39
Table 4.6 Results of Impulse Noise Attacks on hybrid SVD/DWT Technique	41
Table 4.7: First set of results of extracted logo after JPEG Compression attack	42
Table 4.8: Second set of results of extracted logo after JPEG Compression attack	43
Table 4.9 Results of extracted watermark after JPEG Compression Attacks	45
Table 4.10: Comparison of Thesis Results with three other related Works	46



# CHAPTER 1

## INTRODUCTION

### 1.0 Background to the Study

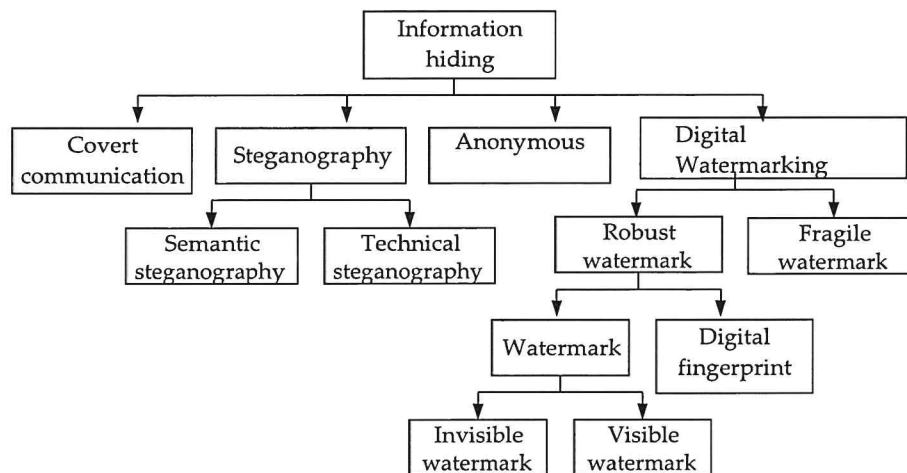
The advent of rapid technological advancements in electronic communications networks continues to improve the processing mechanisms by which transmission and distribution of multimedia content digital content present many explicit leverage in contrast to the analogue versions. However, the structure of digital content allows for simple means of unlimited unauthorized replication without loss of fidelity creating near identical results to the original thereby threatening the enforcement of intellectual property rights. These challenges have provided the impetus control over the ways in which material used in digital formats can be used commonly referred to as DRM an information hiding technologies involving the introduction of encryption to mitigate them. Although this method does is capable of delivering general security, the content is not safe from facile control, and dissemination in the Internet ecosystem. It is for this reason that various transform techniques have emerged as key means of addressing authentication of genuine content [1].

### 1.1 Information Hiding Technologies

Information hiding technologies provide the means by which crucial intelligence can be concealed in another digital product to facilitate its transmission over open carrier systems. These techniques allow for the effective realization and secure conveyance of confidential intelligence between networks [2,3].

Some of the data hiding technologies that technologists have advanced include *inter alia* steganography [4], digital watermarking [5], and covert communication [6], as shown in Figure 1.1. However, the various watermarking technique have been developed for preserving Intellectual Property rights infringements especially copy protection for various converged media content. Application of digital watermarking techniques have continued to expand with the technological changes. These have included but not limited to video Authentication,

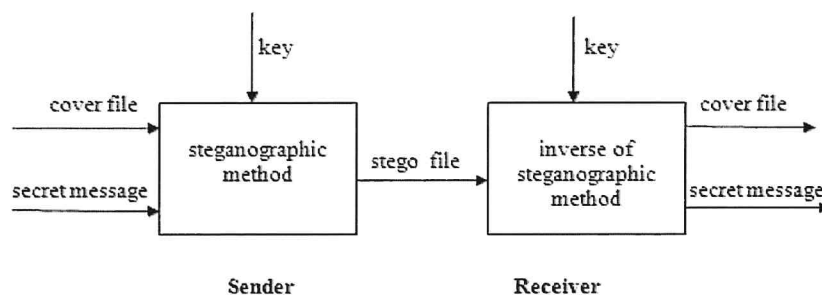
Content Filtering, Online Location, Data Authentication, Copy Protection and device control amongst others [6].



**Figure 1.1 Data hiding Technologies**

Watermarking using digital techniques involve concealing information [7] that are termed as messages into the original multimedia content. It confirms the genuineness of ownership of the content. [8]. Generally, the method is categorized as imperceptible or discernable category [9]. In the latter, the message is inserted within spatial framework and can be observed but is not easily erased. In the former category however, the watermark is not noticeable during routine observation and if attempts are made to remove it the essence would be adversely compromised. Principally, several techniques have been extensively deployed to enhance the protection of intellectual property rights from unauthorized access and include steganography, cryptography and watermarking. Steganography the practice of concealing a message within another content so as to prevent detection by unauthorized entities. In this technique content may be in the form of different file formats including text, images, and videos, though the digital variants are pervasive over world wide web. Figure 1.2 indicates the components of a secure steganographic network demonstrating the following operation.

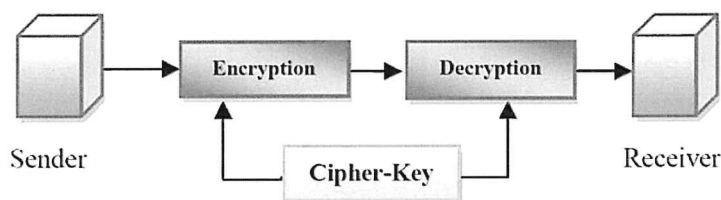
At the source of the transmission a cover file, secret message and key are mixed in the steganographic method. The key contains codes required during the insertion and extraction of the message at the source and the sink. The sending end is known as the steganographic function and its role is to takes the content at the source generating a file as shown below.



**Figure 1.2: A generic Steganography System [10]**

The output stego file is transmitted over the media through appropriate means to the sink/receiver, where it is inputted into the inverse steganographic method.

The next popular data hiding technology that preceded steganography was Cryptography which is shown in Figure 1.3. The process begins at the sending end where information is cipher keys are shared at the encryption and decryption stages in the transmission media. However, its challenge is the availability of the intelligence once the decryption is affected at the sink.



**Figure 1.3: Cryptography System [10]**

Digital watermarking involves embedding a logo or signature in the multimedia content for authentication or verification purposes. It has its vulnerabilities but has however emerged to be a much more superior means of protection of copyright content as it is versatile in a wide variety of functions useful in complimenting other techniques [11].

## 1.2 Digital Watermarking Properties

During water making the transmitted content can carry multiple messages concurrently. In visible watermarking scheme, the information is noticeable on the multimedia content. This is characteristic of normal insignia observed in broadcast content on Television screens whereas imperceptible scheme involves adding content to media data that is transmitted across the Internet ecosystem.

### 1.2.1 Video Watermarking Specifics

This process involves taking a video sample and splitting it into frames, and using the same techniques inherent in image watermarking, inserting a message within each frame and recombining them to enable the concealment of information. The video is then transmitted over communications channels and it's only the creator who remains cognizant of the presence of the additional message embedded in the video frames. The video frames can either be compacted(compressed) or uncompactd. The structure of the frames

Within the compacted frames domain watermarks can be embedded within video streams without the need to un-compress followed by and recombining them. However, in the uncompactd category there is need to un-compact the video stream to embed the watermark, followed by subsequent recompression as part of the watermarking technique. While uncompactd technique has more computational demand, it compensates the complexity with use in larger variety of video types.

Within the two domains there are other sub-groups namely imperceptive versus perceptive. These categories demonstrate whether the process need the cover video frames for detection purposes. In the perceptive process there is need to transmit the cover frames whereas, in the imperceptive category extracting the message is possible without the cover frames. Imperceptive detection is generally complex extorts limits on the data capacity. However, the perceptive schemes though have potential of robustness creates the challenge of requiring massive databases of the original data, making them not useful for certain applications. This research investigation shall be focused on the blind domain video watermarking techniques considering the apparent limited data capacity and costs in running large databases inherent in the non-blind schemes.

### 1.2.2 General classification and requirements of watermarking

The classifications can be described as robustness, embedding effectiveness, non-perceptibility/fidelity, non-detectability, security, complexity, capacity among others. These are discussed next.

**a) Robustness:** A watermarking algorithm is termed robust if the embedded watermark can be retrieved effectively with minimum distortion after the content has been subjected to various attacks.



**b) Embedding Effectiveness:** Termed the possibility that the recovered message will be watermarked, in other words, this is the likelihood that it can be detected after the process. The desired effectiveness is usually pegged to unity (100%), although not always attained.

**c) Non-perceptibility/Fidelity:** Fidelity is the correlation of the host and the processed content depending on perceptibility or otherwise .

In the invisible category, the watermarked content should look like the original one without any perceivable distortion. A perfect non-perceptible scenario if the cover file and the one with the inserted message are indistinguishable [12] through use of the HVS.

**d) Non-detectability:** Content accompanying message is unrecognizable if it dependent on the original one.

**e) Security:** This is the capability arises out of the fact the secret key is unavailable to any attacker who may try to control the process and impair the message.

**f) Complexity:** The cost of hardware or software or difficulty in identifying the embedding and extracting the algorithm used on the watermark. The algorithm should be simpler for the authorized users and very complex for any unauthorized user or intruder.

**g) Capacity:** The amount of data that the content can stored without causing significant loss in quality.

**h) Blind and Non-Blind Detection:** Blind detection defines the capability of the detecting algorithm to extricate the stored message without the reference to the original un-watermarked work. Non-Blind detection however needs the cover image to successfully recover the message

**i) False Alarm:** This is the rate of identifying the message in a content that has none.

### 1.3 Applications

At the turn of century these have included finger printing, checking of images, protection of intellectual property rights and data [13]. However, two decades ago more advanced uses of this concept have emerged accompanied by new algorithms that have tended to create various challenges for certain applications.

The following examples reflect some of the areas in which watermarking are applied:

a) **Copyright Protection & Authentication:** Watermark(s) can be inserted in the cover frames which when decoded defined the creator of the content. Protection and verification is deemed to have taken place when the process is robust to various attacks [14].

- b) **Broadcast Monitoring:** The broadcast monitoring can be determined through use of passive or active mechanisms. However conventional systems require large amounts of storage for monitoring infringements whereas digital video watermarking techniques are able to offer active monitoring in an imperceptible way [15].
- c) **Copy Protection and device control:** The advent of the Internet has popularized video streaming creating the need to control the capacity of unauthorized users stifle innovations through piracy. This achieved by placing watermark decoding means at the Internet Services providers' network nodes [14].
- d) **Video Authentication:** a message can be inserted in the video frames of the cover files and forms the basis of determining its authenticity, e some examples of applications where this concept is important is on such areas as surveillance and medical imaging [16].
- e) Other applications include Fingerprinting, Online Location [17] and Content Filtering [18] and Data Authentication [19]. Other uses in the medical field include among others X-Ray images and Magnetic Resonance Imaging (MRI) which can be watermarks for secure storage [20].

#### **1.4 Problem Statement**

This research work considered the substantial efforts made on watermarking methods have focused on textual and image analysis whereas currently most multimedia content transmitted over the Internet are in video form. Digital Watermarking supersedes other methods as preferred means of copyright security and fingerprinting of content; however, most techniques are susceptible to a variety of attacks. This has prompted efforts to be invested in formulating schemes that attackers should find computationally expensive or the affected content severely damaged when attempts are made to remove the watermark. The main challenge of digital video watermarking is that most watermarking techniques are fragile to either signal processing or geometric distortions. In this study, efficient, resistance, and invisible watermarking procedures algorithms were implemented. They were based on the SVD techniques and its combination with Discrete Wavelets Transform (DWT), that were subjected to various distortions/interference to determine their resistance against them.

## **1.5 Justification of the Study**

Following on from the motivation this study is justified on the grounds that it is important to encourage innovation through guarantee of preservation of Intellectual Copyright. This was done through use of versatile techniques that include among other Singular Value Decomposition that has in recent years proven to be a data driven ecosystem that is used in various numerical analysis techniques for data reduction, among others. Using such techniques as environments for digital watermarking lend ways and means of assuring innovators of the nature and type of environments to protect their works. The use of the SVD/DWT technique provides means of utilizing the exemplary qualities of frequency domain watermarking ecosystems that possess the capacity of mitigating geometric and signal processing attacks among others. Digital watermarking has proven to be a powerful method of protecting unauthorized duplications of multimedia content. It is more superior than encryption, and steganography.

## **1.6 Objectives**

### **1.6.1 Main Objectives**

The research works was to investigate the use of digital video watermarking technologies, comparing SVD and its hybrid with Discrete Wavelet Transform, evaluate robustness of each of under various attacks.

### **1.6.2 Specific Objectives**

- I) To subject the SVD and SVD/DWT watermarked video frames to Histogram equalization, median filtering, JPEG Compression, Gaussian and Salt & Pepper attacks.
- ii) To carry out a comparative analysis of the performance of SVD and Hybrid SVD/DWT frames after Gaussian and Impulse Noise attacks, JPEG Compression attacks.
- iii) Verification of results of the superior techniques by comparison with related works and draw conclusions.

## **1.7 Scope of Work**

The scope of the investigations was restricted to watermarking for copyright protection and authentication purposes. The study carried out an analysis of the effectiveness of video watermarking using Hybrid SVD/DWT as compared to SVD as control. Watermarking of audio and text among others were not considered. The experiments were executed within the

MATLAB/SIMULINK environment. There are other commercial software platforms that are watermarking platforms that are patented however, MATLAB/SIMULINK one is popular for academic work

## **1.8 Thesis Organization**

The rest of this research investigation continues with Chapter 2 the review of the current literature followed by Methodology in Chapter 3. In Chapter 4 the computer simulation results, and discussions of the developed algorithms are presented followed by Chapter 5 that provides the conclusions and recommendations for future works.

## **1.9 Note on Publications**

The contribution from this thesis work was presented and published jointly with the Professor Elijah Mwangi (my supervisor) during the 2017 IEEE African Conference (AFRICON 2017). The paper entitled 'A Robust Video Watermarking Approach based a hybrid SVD/DWT technique'. A copy of the paper is given in the Appendix B.

# CHAPTER 2

## LITERATURE REVIEW

### 2.1 Introduction

Creativity, and innovation of multimedia content that pervades the cyberspace, are prone to attackers that lie in wait to dispossess them of their tireless efforts. The use of digital video watermarking schemes has continued to occupy the minds of researchers and quite substantial works has been invested into coming up with ways and means of not only protecting the innovations from theft, but also from the attendant distortions arising out of failed attempts to interfere with them as they transmitted through various media. This research investigation sought among others to distortions review the main digital video watermarking methods, several watermark attack mechanisms and then provide a justification for pursuing the objectives of this research investigation. Important concepts on video watermarking are presented and review of related works was carried out to identify the knowledge gaps and hence justify the research investigation.

### 2.2 General Digital Watermarking Techniques

Digital watermarks are inserted with using various means, approaches and processes. If the watermark is visible then it will be observable by anyone, however there are invisible ones, that formed the subject of this research investigation.

#### 2.2.1 Basic Principle of Digital Video Watermarking

The three main stages in the watermarking process are *generation and embedding*, *attacks* which are normally experienced within the transmission medium, and the *retrieval/detection* of the output messages. The video stream to be watermarked is referred to as the cover file, and has the secret signature or logo inserted called the message. The inserted message is accompanied by secret key(s) which then provides the watermarked version. Only the owner and the authorized users have access to the key, meaning that it is impossible to remove the message without knowledge or access to it. The processes used

video watermarking techniques and their image versions are similar, since videos must be split into frames, whose properties are like still images.

### 2.2.2 General overview of the Watermarking Process

A simplified message insertion, and extraction process is illustrated in Figure 2.1 which is a typical block diagrams of a watermarking system. As illustrated in the diagram the inputs to the embedder are the message, the host content consisting of the video frames and the security to obtain a watermarked version.

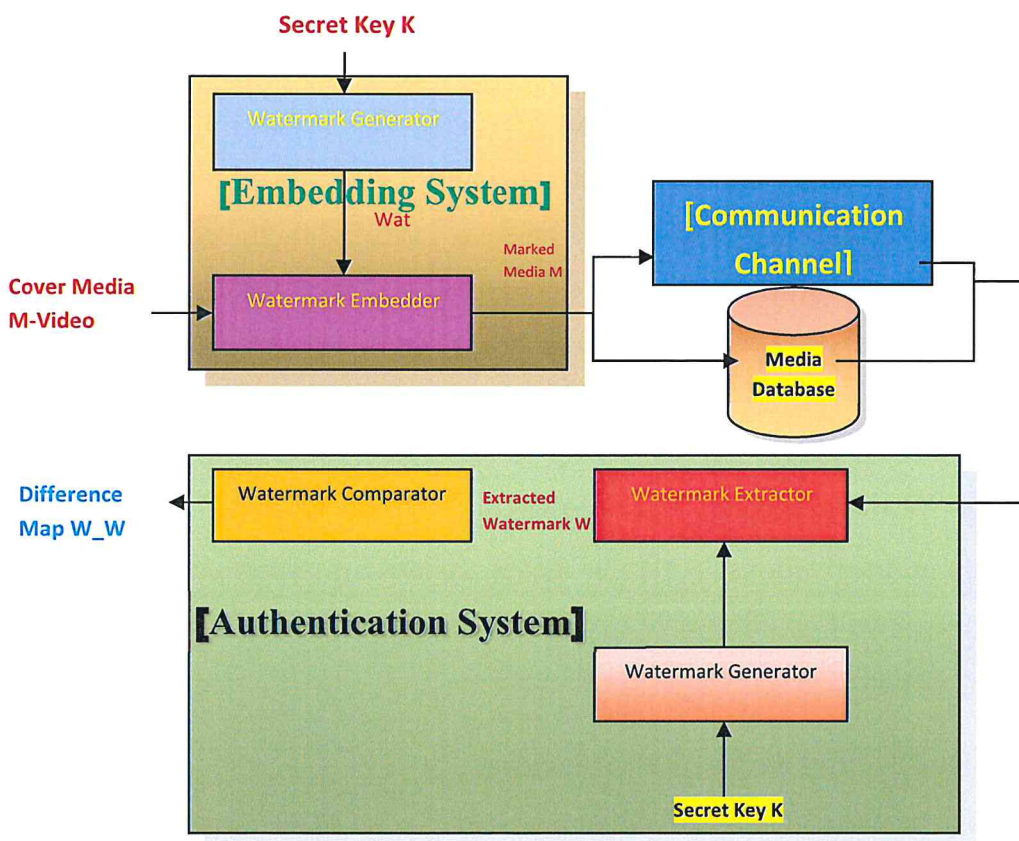
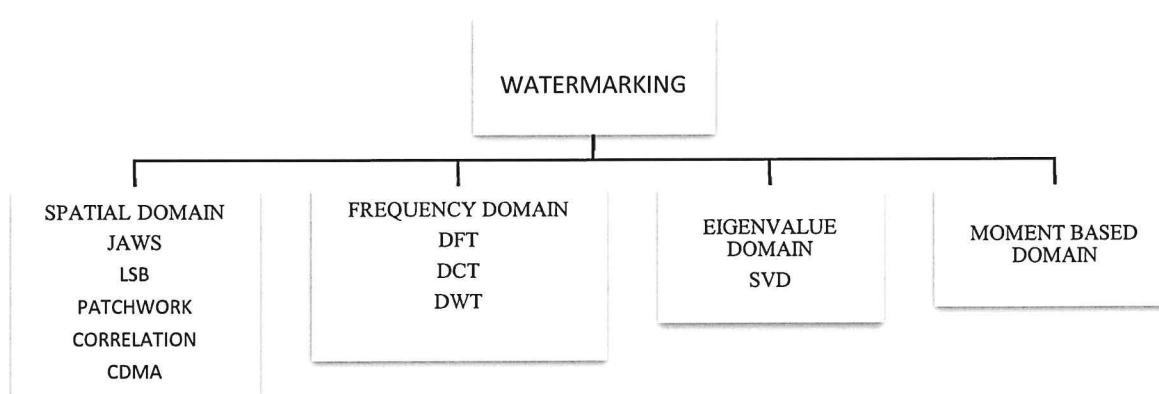


Figure 2.1: Simple watermarking scheme

The Choice of watermark  $k$  can be formulated from text, number or binary bit sequence or an image. Cox et al. [21], have described the watermark/message detection mechanism as a two-step process as shown above. The process can be achieved by either sending the original host image from the source to the sink to assist in the verification of the output or not send it at all. The former is termed blind while the latter non-blind watermarking [22].

## 2.3 Digital Video Watermarking Methods

There are four main categories of the methods as depicted in Figure 2.2 below. These techniques are: Spatial, Frequency, Eigenvalue and Moment based Domain methods. Some examples of spatial methods are the Hardware based Just another Watermarking System (JAWS), and the software based Least Significant bit Based Schemes, Patch Work Based Schemes, Correlation Based Watermarking Scheme and CDMA Based Image Watermarking Scheme.



**Figure 2.2: Digital video watermarking Techniques**

In the Frequency domain we have examples such as DFT, DCT, DWT among others. Singular Value Decomposition as an example of a technique to be found in the Eigenvalue domain and then we have the Moment or invariant based watermarking schemes that utilize the statistical property of an image [23].

## 2.4 Spatial Domain Watermarking

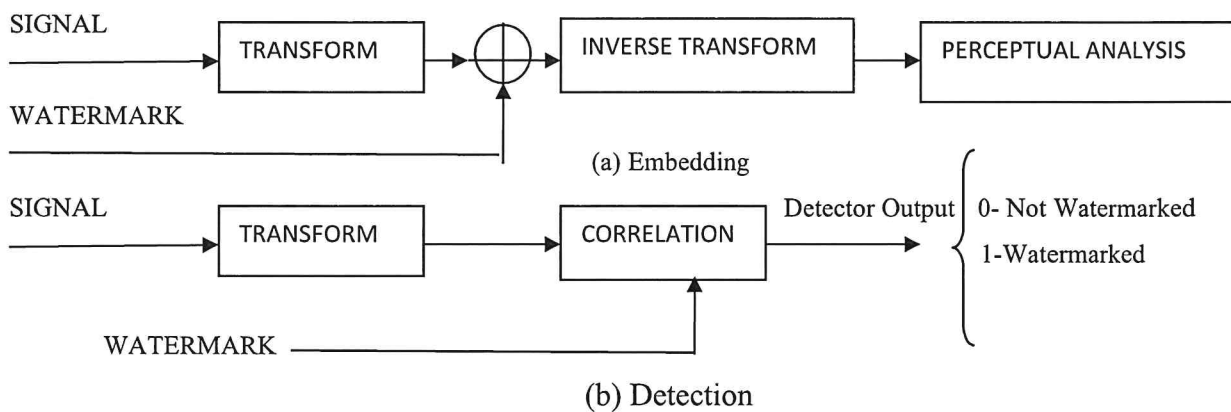
Spatial domain method involves insertion embedding and recovery of messages/watermark without application of complex techniques.

These techniques are relatively computationally efficient and provide sufficient mechanisms for applications where there are limited resources [24]. Some of the existing techniques include just another watermarking Scheme (JAWS) and broadcast monitoring scheme among others where the entire mechanism is implemented in the hardware.

These solutions are often more economical than software-based systems implementation that requires the inclusion of dedicated processors occupy with more area, that are energy hungry and slow.

## 2.5 Frequency Domain Watermarking

In this method which include but not limited to DFT, DCT or Discrete Wavelet Transform the cover content is first changed to frequency coefficients that are subsequently varied message and the inverted appropriately to obtain the product. The mechanisms are more complex than the spatial methods as a result of the computation of transforms. However, the outputs are robust when subjected to various attacks. In the frequency domain watermark detection, correlation techniques are often employed. A simple illustration of such a technique is illustrated in figure 2.3 below. For the embedding process shown in Figure.2.3(a), a signal is converted appropriately in the techniques and mixed with the cover image at the appropriate wavelengths. The mixed signal and the watermark are subjected to an inverse transform and a perceptual analysis carried out.



**Figure: 2.3 Basic block diagrams of frequency watermarking technique**

This whole process is what constitutes the embedding of the watermark, and for purposes of detection to determine the veracity or quality of the message that has undergone a perceptual analysis is transformed again and passed through a correlator with the watermark and the detector output as shown in Figure.2, 3(b) will indicate a null for the case of a media that was not watermarked and a 1 for a watermarked media.



### 2.5.1 The Two-Dimensional Discrete Cosine Transform

DCT is an extensively popular method used within the frequency domain. [25, 26]. This method is often used by subjecting a host image/frame with compressed videos, the process being achieved through the splitting of image/frame into non-overlapping blocks. Each DCT elements are converted and the coefficients quantized through special algorithms [27].

### 2.5.2. Discrete Cosine Transform Encoding

Mathematically the two-dimensional (2D) ( $N \times M$  image) DCT is expressed as follows:

$$F(u, v) = \sqrt{\left(\frac{2}{N}\right)} \sqrt{\left(\frac{2}{M}\right) \sum_{i=0}^{N-1} \sum_{j=0}^{M-1} A(i) \cdot A(j) \cdot \cos\left[\frac{\pi \cdot u(2i+1)}{2 \cdot N}\right] \cos\left[\frac{\pi \cdot v(2j+1)}{2 \cdot M}\right]} \cdot f(i, j) \quad (2.1)$$

The inverse  $F^{-1}(u, v)$  is such that

$$A(\xi) = \begin{cases} \frac{1}{\sqrt{2}} & \text{for } \xi = 0 \\ 1 & \text{otherwise} \end{cases}$$

The primary process involves:

- a) The input frame/image dimension is  $N \times M$ .
- b) The picture elements depth  $i^{th}$  and  $j^{th}$  rows and columns respectively is  $f(i, j)$
- c) DCT coefficient  $u_i$  and  $v_j$  rows and columns respectively is  $F(u, v)$ .
- d) Most of energy of the signal resides at lower frequencies for most frames appear in the top corner to the left of matrix.
- e) Considering that the lower right content representing higher frequencies contain negligible amplitudes, compression is achievable but can be omitted with little visible distortion.
- f) The  $8 \times 8$  integers input arrays ranging from 0 to  $2^8$  for a one-byte resolution image each pixel containing gray scale level.

## 2.6 Singular Value Decomposition

Numerous algorithms have been advanced for multimedia content this novel matrix formalism in the last few years. They can be categorized as either pure or hybrid versions. However, in hybrid version the message is normally inserted in the Singular Value Decomposition in the appropriate frequency transform domain.

### **2.6.1 Variants of SVD**

The three variants of SVD based algorithms are pure, block and non-block-based versions.

#### **a) Pure SVD**

Conventional images/frames are normally used in most Singular Value Decomposition techniques use as the test benches prompting open research in the field of medical imagery procedures [28]. Medical image watermarking requires the maintenance reliability, confidentiality of the appropriate content. This guarantees the inherent universally expected doctor-patient confidentiality and privacy requirement. This explains the inapplicability of conventional watermarking schemes for medical data since they focus on authentication, intellectual property protection, ownership, and security.

#### **b) Non-block-based algorithms**

These consist scenarios where the message is inserted directly in the singular Value values as has been proposed by Liu and Tan [29]. They used a blind version that required orthogonal matrices to retrieve the watermark. The output was resistant to filtering compression, cropping among others not robust enough to withstand geometric attacks. Liu and Kong [30] suggested mechanism relying on applications of Singular Value Decomposition algorithm on basic M-sequence as the watermark. Their method entailed central embedding of the watermark within the SVD matrix components of SVD while leaving the original sequence unchanged with the output maintaining the visible quality and robustness criteria, that was able to resist various signal processing distortions though unable to withstand rotation, cropping and translation among others.

#### **c) Block-based algorithms**

Unlike the Pure and Non-blocked based watermarking techniques, the Block based ones are invisible and linear outputting stable watermarks. These results are adequate for the most medical images considering that they deal with security, integrity and reliability. In addition, the Block based scheme can withstand several attacks providing for robustness.

Some of the authors who have carried out research in block-based algorithms include Ganic and Eskicioglu [31]. They have proposed an optimal SVD technique where the watermark is embedded within two strata.

An SVD technique where the cover frame is divided into blocks and the message inserted in them separately have been proposed by Agarwal et al. [32]. The process gave similar results like [31] though the correlation values for rotation operation and resizing tested for small angles were unacceptable.

Zhou, Tang and Liu [33] have presented a technique where the cover content is divided into numerous  $8 \times 8$  sized sections and the message inserted in the singular values. Their result demonstrated resistance to various signal processing but could not withstand RST attacks.

### 2.6.2 Principles and properties of Singular Value Decomposition

Mathematically this is an effective analytical scheme extensively applied in matrix analysis. This tool involves the decomposition of a square matrix into three matrices of similar sizes as the original one. Consider a frame  $A$  denoted as by the expression  $A \in R^{n \times n}$ , where  $R$  represents the real number domain, then singular value decomposition of  $A$  is defined as:

$$A = UDV^T \quad (2.2)$$

Where  $U \in R^{n \times n}$ ,  $V \in R^{n \times n}$ , and  $D \in R^{n \times n}$  is a diagonal matrix, as  $U$  and  $V$  are orthogonal,  $U^*U^T = I$ ,  $V^*V^T = I$  and for a 4x4 matrix.

$$D = \begin{pmatrix} \sigma_1 & 0 & 0 & 0 \\ 0 & \sigma_2 & 0 & 0 \\ 0 & 0 & \sigma_3 & 0 \\ 0 & 0 & 0 & \sigma_4 \end{pmatrix} \quad (2.3)$$

The diagonal elements  $\sigma_n$ 's are singular values satisfying the condition:  $\sigma_1 \geq \sigma_2 \geq \sigma_3 \geq \sigma_4$

The Singular Value Decomposition technique as an analytical tool possess useful attributes that can be applied in digital image processing [34]. Due to these properties of SVD, several algorithms have tested been suggested.

Some of the advantages that SVD are:

- i) Matrices' magnitudes are variable
- ii) Resulting frames/images are resilient
- iii) The resulting frames/images show inherent algebraic attributes

This technique though computationally complex can be used on its own for watermarking. However, when combined with several frequency transforms to form hybrid techniques this tends to lower the complication.

## 2.7 Discrete Wavelet Transform Watermarking

When an image is digitized, it is unnecessary to carry out any abstraction to elicit further details. This implies the use of a mathematical model makes it easy to switch between various levels of detail, or resolutions. This forms the basic principle behind multiresolution analysis, and one of the frameworks that effectively uses this concept are wavelets. As a mathematical formalism, wavelets are amenable to intelligence extraction from several data points that may include but not limited to images, video frames or even audio contents. Within the wavelet concepts there are sets of "complementary" types having the capability of decomposing data without disparity or protrusions allowing for mathematical reversibility of the entire process. These sets are useful in the use of either magnitude reduction or enhancement algorithms where it is important to desirable to retrieve host information with minimal loss of content .

The underlying concept of Discrete Wavelet Transforms as derived from the wavelet functions, mirrors those of the Discrete Fourier Transform considering that they are both orthogonal function which can be applied to a distinct finite data group. The other important aspects are that the functions are convolutions and carry out the transformation in an orthogonal manner. This involves consecutive transfer of a constant signal with the input ones being sets of discrete-time samples. There are several wavelet functions, key among them being the Haar, Morlet, Debauchees, Mallat among others. In this study use was made of the simple Haar wavelet for the simulation environment. The reasons for the choice were based on the fact that the Haar wavelet was the original and elementary orthonormal type wavelet basis function. Conceptually it posses the properties of basic , memory efficacy , absolutely mutable without the boundary effects characteristic of other complex ones and less resource demands .

In the analysis of wavelets if there is a function be identified by ,  $\psi(t)$ :  $\psi(t) \in L^2(R)$ :

$$\text{Then } \psi_{m,n}(t) = 2^{-\frac{m}{2}} \psi(2^{-m}t - n) \quad m, n \in Z \quad (2.4)$$

The signal  $f(t) \in L^2(R)$  can be then represented as

$$f(t) = \sum_m \sum_n d_{m,n} \psi_{m,n}(t) \quad (2.5)$$

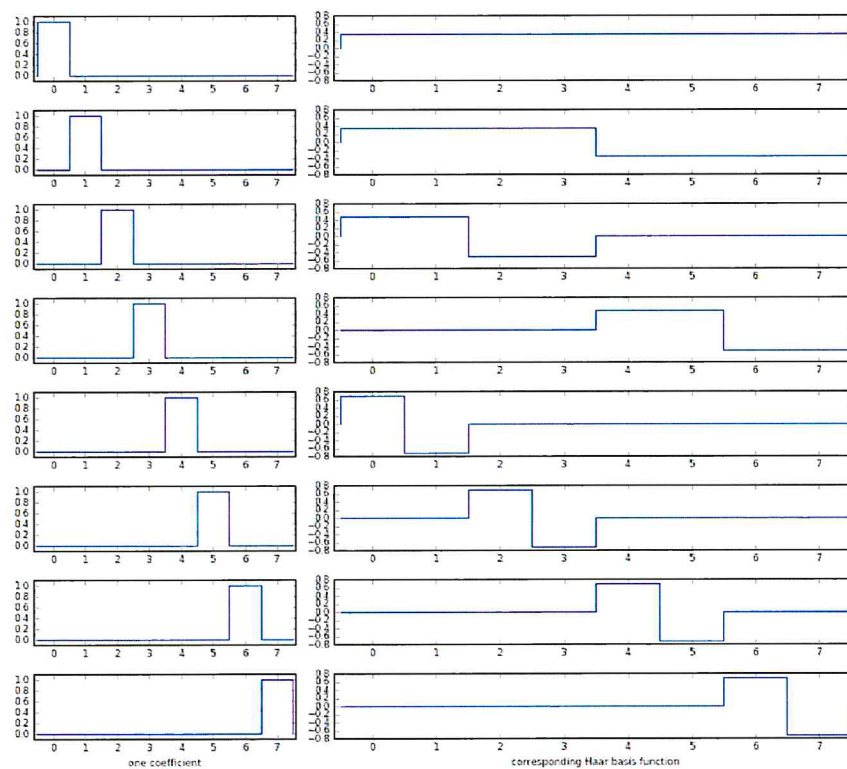
where  $d_{m,n}$  are spectral values

$$d_{m,n} = \langle f(t), \psi_{m,n}(t) \rangle \quad (2.6)$$

For discontinuous data  $f(k) \in L^2(Z)$  hold the same outputs creating an entity termed the Discrete Wavelet Transform (DWT). A simple wavelet used in this process is the orthogonal Haar wavelet transform, described as:

$$\psi_{Haar}(t) = \begin{cases} 1 & \text{for } 0 < t < 0.5 \\ -1 & \text{for } 0.5 < t < 1 \\ 0 & \text{otherwise} \end{cases} \quad (2.7)$$

The basis function for the Haar wavelet and its  $\psi_{m,n}(t)$  functions are shown in Figure 2.4. The lack of linear dependencies between these functions, creates different magnitude types throughout the scales. This important property is extensively utilized in compression of images. Magnitude relationships allow for parent child linkages in statistical m parent child dependencies modeling and estimation between  $d_{m,n}$ . These linkages carry information about correlations within the scales.

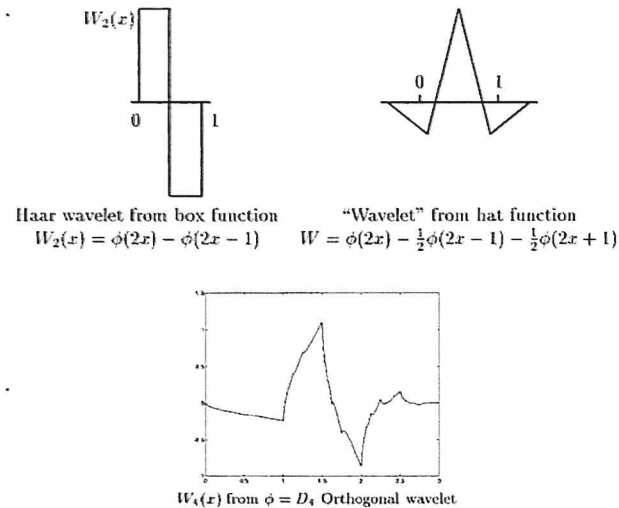


**Figure 2.4 The Haar wavelet basis function ((source code: [haar-basis.py](#))**

It is important to note the Morlet (Mexican hat) is strictly  $\psi(x) = \frac{1}{2} \varphi(2x) - \varphi(2x - 1) + \frac{1}{2} \varphi(2x - 2)$  and inverted to the above diagram.

In existing literature Al-Haj [35], Lai and Tsai [37] have reported that in comparison to DCT, DWT domain slightly improves the perceptual quality of the watermarked image but still has insufficient robustness to geometric attacks.

Al-Haj proposes a combination of Discrete Wavelet and Discrete Cosine Transforms to further improve the perceptual quality and robustness to geometric attacks.



**Figure 2.5: Examples of various Wavelets**

## 2.8 Hybrid Transform-Based Watermarking Algorithms

Ganic and Eskicioglu [31] have presented a hybrid scheme of the SVD and DWT. This scheme demonstrates how the host frame is broken down by the discrete wavelet transform into four sub bands following which Singular Value Decomposition technique is on host video frames. The final stage involves obtaining four sets of discrete wavelets transform coefficients which when subjected to the inverse Discrete Wavelet Transforms using the modified coefficients generates the processed frame. The algorithm was found to be robust against signal processing distortions among others. The recovered messages from each sub-band and the correlation coefficients compared to the host image/frame indicated that the type obtained from the low frequency band had the best visual quality.

Other digital techniques, are Moment-based watermarking [37] and use of principal component analysis [38,39].

## 2.9 Video watermarking

Some customized video watermarking schemes that are gaining popularity including frequency techniques like DWT [40]. In these methods, messages are embedded in the split

frames of digital video. In normal circumstances users are not able to distinguish the difference between the host and the processed contents. However, a message recovery mechanism can retrieve it since it is part of the content, rather than the file format.

The primary aim of the whole process is to safeguard PR of the content creator. It has been established that several existing methods are open to attackers to freely circumvent it by manipulating the processed image. Splitting the video stream and insertion of messages can facilitate the identification of the offenders, and in the process protect Intellectual property rights, hence encouraging innovations.

### **2.9.1 Spatial domain**

The technique involves the insertion of a message precisely into the luminance or chrominance segments of the cover frames, that should have two important traits: high payload and low complexity [41]. in addition, to spread spectrum modulation schemes [42] among others. The LSB method the most popular of these two methods has been proposed by H. Kaur et al [43] and more work done by Bayouhd et al [44] to improve its robustness. Spread spectrum based watermarking algorithms on processes are also effective in this technique [45, 46].

For example, a semi-blind scheme based on based block classification was proposed by Bahrami et al [47]. An algorithm involving use of spatial uniform mapping was proposed by Li et al [48] meant to withstand scalable recompression and transcoding attacks. Spatial domain algorithms are widely used in the early parts of the process because of their simplicity.

### **2.9.2 Transform Domain video Watermarking schemes**

The process initially involves the conversion of frames within the frequency ecosystem, followed by appropriate variations of the coefficients to allow the insertion of the message. This process is then followed by the conversion of the video frames into spatial sphere to extract the content. Several techniques used in these processes and in some cases, they are combined to improve the outputs through leveraging on their different characteristics.

## 2.10 Related works

In the frequency domain DCT, DWT as well as SVD and hybrid techniques made up of combinations of SVD and DWT are the most used widely methods in this domain as demonstrated in the works [49]. SVD, DWT transform can be used alone in the watermarking process among others.

Mostafa et al [50] have presented in their works a DWT based watermarking algorithm that satisfies the invisible characteristics. In their works the watermark is hidden by selection of blocks with highest motion vector magnitude  $k$  after being processed using DWT technique. Shukla et.al [51] suggest use of three-level DWT watermarking technique where the message is inserted in the low frequency portions on scene-changed frames. A blind approach using Discrete Cosine Transform has been proposed by Tuan et al [52] proposed in which the luminance segment is obtained from the cover frame is broken down into  $8 \times 8$  DCT blocks where the message is inserted on randomly chosen coefficients.

Digital Video watermarking techniques have also been used based on a combination of domain transformation techniques. Narasimhulu, [53] has propped schemes based on combination of frequency methods where the host video frames are broken down and the message subjected to SVD video using an additive non-blind algorithm. Techniques based on several frequency and eigenvalues techniques have been suggested by Naved [54], Jeebananda et al [55] and Reyes *et al.* [56].

A two (2)-level DWT video frame component dependent techniques has been suggested by Kareem Ahmed *et al.* [57]. In their works independent watermarks were embedded into separate slots by their method where choice was made between the *HLL* of RGB components per frame based on a code that embedded disparity mitigation measures into one of them. However, they were unable to justify the use of any of these components considering that they have different pixel contents. In Rao [58],] he proposed two blinds, based on Singular Value Decomposition employing only PSNR as the image quality assessments metric.

In a spatial watermarking scheme investigated by Gayer [59] the message was rendered visibly subtle making it undetectable under ordinary observation it appeared after the colours



separations during the printing process. However, the removal of the message necessary from the colour band for the document to be useful for printing. Frequency and Spatial-domain methods gaps were presented by Behal et.al [60], where the former was more widely applied than the latter. Chang et. al [61], established that Discrete Cosine Transforms possess similar properties like Fourier Transform a useful concept since it compares well to the mechanisms by which human beings recognize light. It has been observed that the frequency techniques demonstrate better attributes of the human visual systems than the others [62]. Szczypiński et. al [63], have presented the use of DWT and in their results explained the suitability of wavelets as means of enabling image analysis at different resolution. The main features of HVS were studied by [64] demonstrating how their means of translation into watermarking techniques.

The absolute application of discrete cosine transform coefficients have been suggested in [65] where they were split into an arbitrary number of sections and their energy levels computed. Messages were then inserted into the highest energy selected peaks per segment and recovery carried out through the inversion watermarking process.

Manaf et. al [66] used DWT or DCT and concluded that Discrete Wavelet Transform results manifested more robustness and higher imperceptibility than DCT in watermarking based on genetic algorithm (GA). Digital image watermarking technique used for copyright protection of digital information have also been proposed by Khanna et. al [67].

### **2.11. Forms of Attacks on Watermarks**

It is also important to take cognizance of the fact that in real systems, there are different forms of attack that are basically processes that either introduce noise or create multifarious distortions on signals and content. In any communication channel, due to the natural configuration, some form of distortion can be introduced by offenders with a view of attempting to erase the message or manipulate it in such a manner as to take possession of the creators' intellectual properties. This is normal in society; hence it is critical that there is an opportunity to classify the attacks in general since in the experimental settings, they provide a form of risk analysis, and tests the efficacy of the algorithms. Voloshynovskiy et al. [68] has presented substantial works in this area and Figure 2.6 summarizes some types of attacks.

**i). Removal:** This is an attempt to obliterate the message from the host content [69].

ii) **Cryptographic:** These types of distortions are related to the ability to compromise the security of the content being transmitted.

iii) **Geometric:** Manipulations affecting the geometry known as rotation, scaling and translation (RST) attacks should be detectable, though videos are not affected by such which affect images in general during watermarking.

iv) **Protocol:** These distortions are focused on semantic challenges of the watermarking applications [70].

Lai and Tsai [37] proposed the use of Discrete Wavelet Transform and SVD to improve robustness to signal processing and geometric attacks. This proposal exhibits improved robustness to numerous attacks while at the same time increasing computational complexity of the embedding and extraction process.

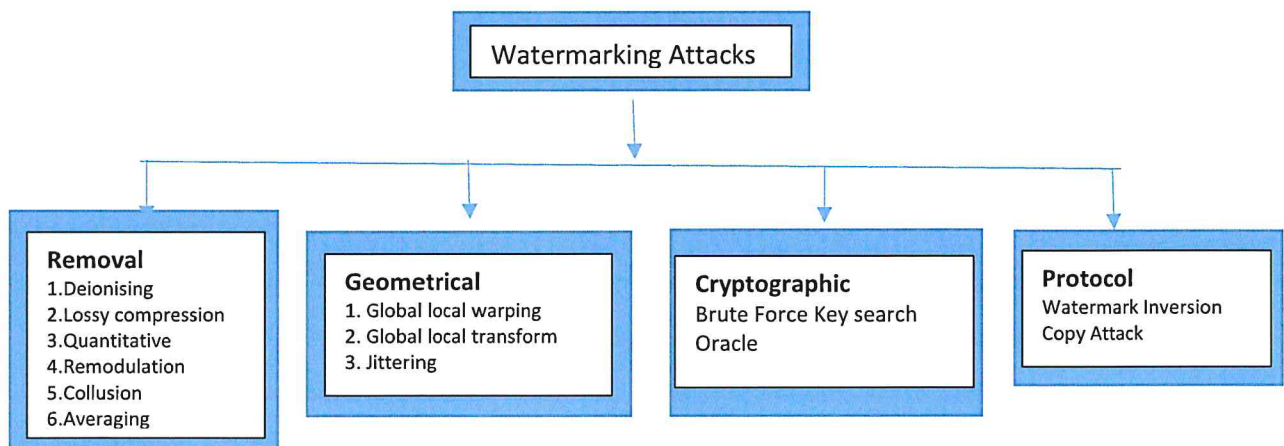


Figure 2.6: Different attacks on watermark [71]

## 2.12 Knowledge Gaps

From the literature review the following weaknesses have been identified.

- a) Some research investigation does not to justify the use of any of three colour components red, blue, or green considering that they have different pixel contents. The lack of criteria for choosing any of the three leads to an open investigation.
- b) In most cases the formulated algorithms are not explicit on how the videoframes are split to enable the watermarking process.

- c) Most of the reports reflect watermarking exclusively within the spatial or frequency domain, with little or no mention of Singular value decomposition nor emphasis on the watermarking of video media. Though research carried out by Rao [58]] on Singular Value Decomposition did use PSNR, it did not however come up with substantial conclusions. This is premised on the fact that in their conclusions they state that they gave a brief review of the current technologies, they mention DWT which it is not clear why it was mentioned since their works did not include it. They even stated that their method was inadequate for general use pointing to the fact that it was basically a localized academic exercise.
- d) Most of these previous research investigations, have not used image quality measure, such as Structural Similarities Index Measurement (SSIM), which is known to be closer the Human Visual System. In most of the cases use has been made of the traditional PSNR as the objective measure that has limitations as envisaged by Wang et al [75].

This research investigation sought to address these weaknesses brings out substantial output on robust digital watermarking schemes, though with some shortcomings which have been recommended for future works. It is instructive to appreciate e that digital video watermarking is a fast-growing field, as the Internet continues to expand with diverse content globally.

# CHAPTER 3

## METHODOLOGY

### 3.1 Developed Watermarking Scheme

In this research investigation the watermark was embedded in the video data using SVD techniques. The software and hardware materials, the data and the methods used achieve the watermarking insertion and retrieval are presented. In addition, measures used to quantify the results are also discussed.

### 3.2 Materials

The materials that were used for this research investigations were made up of the Software, Hardware and Data are listed below.

#### 3.2.1 Software resources

The entire research investigations were carried out in the MATLAB/Simulink environment using MATLAB version R2014b.

**Windows Specifications:** Edition: Windows10 Home, Version: 1803 with OS build: 17134.345 and Internal Hard-Drive: 1Terabyte (TB) memory capacity.

#### 3.2.2 Hardware resources

The Hardware that was used in the research investigation was an HP ENVY 15 Notebook PC with the following specifications.

The computer processor was an Intel(R) Core (TM) i7-4710HQ CPU@ 2.50GHz

The Random-Access Memory of the Computer was 12.0GB

System Type: 64-bit operating system, x64-based processor.

#### 3.2.3 Videos Data

The video resources used in this research investigation for testing and evaluating the embedding and extracting schemes were standard test clips widely used for these types of research investigations namely *Foreman*, *Bus*, *Container*, *Akiyo*, *Carphone*, *Coastguard*.

During the research investigation, it was found useful to use the already available test videos from Derf's Test Media Collection [72]. The video clips themselves were of varying lengths and resolution, so each video clip was split into several frames and the selected types were in CIF and QCIF. The QCIF on the other hand is a lower resolution standard for application of colour coding in audio-video applications. The quarter common intermediate format comes from the standard common intermediate format that can be used to provide colour translation for multiple frames of video.

The difference between the common intermediate format and a quarter common intermediate format is the specific resolution and translation of colour pixelization. Using the quarter common intermediate format will present a 'quarter screen' in relation to the CIF whole screen, or in other words, it provides one quarter of the resolution. Users choose pixelization formats according to the desired resolution and the needs of a business or other enterprise. Appendix A shows details the video Frames used for the simulation experiments.

### 3.2.4 The Watermark logo

The message used as the logo in all the simulation was the IEEE trademark in various sizes from 64x64, to 512x512 pixels. The trademark is given in Fig. 3.1

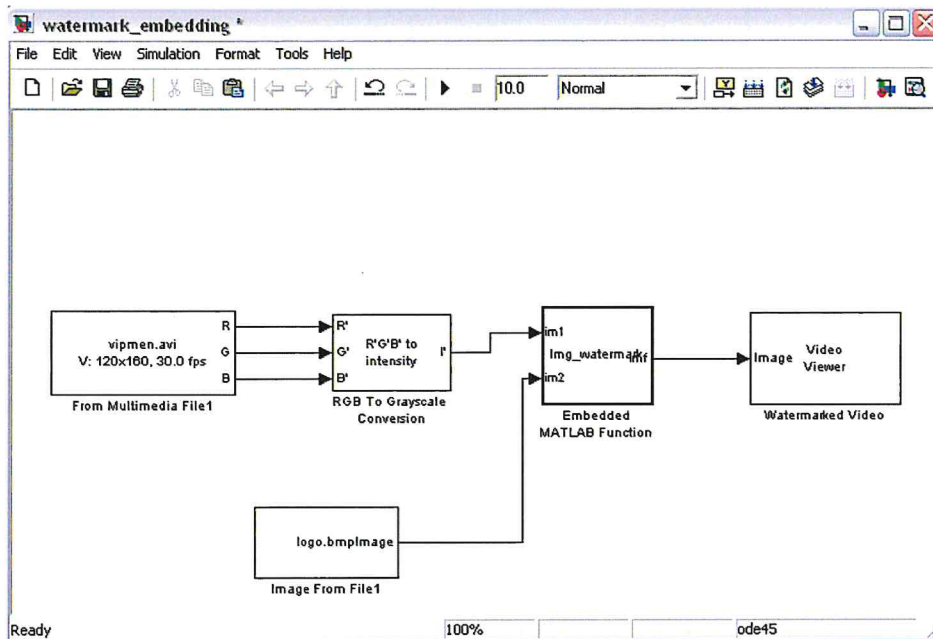


**Figure 3.1: A 256x256 IEEE trademark**

## 3.3 Methodology for Experimental Simulations

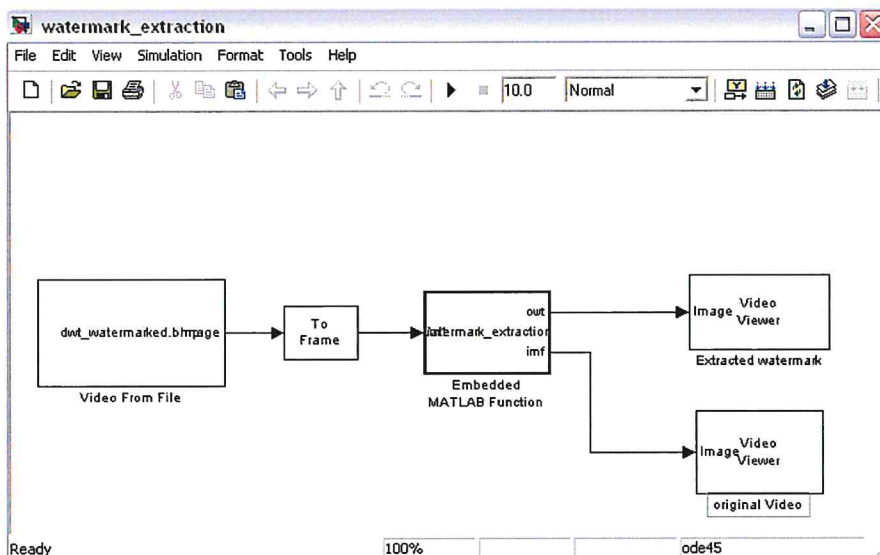
The MATLAB/SIMULINK based implementation was adopted for the entire research investigation and is illustrated in Fig.3. 2. The environment provides for means of inserting the multimedia file in the *RGB* format, which is converted to the gray scale version. The message/logo is mixed with the converted content within the function block resulting in the watermarked image/frame.

The frames are recombined to form the video streams which are transmitted over the communications path.



**Figure 3.2: Watermark embedding model in Simulink [73]**

The recovery process is shown in fig.3.3, where the split frames and subjected to the process that executes that identifies the watermark/message. The results display the video stream and the output message for verification.



**Figure-3.3: Watermark extraction model in Simulink [73]**

### 3.4 MATLAB/Simulink SVD Watermarking

The simplified process illustrated in figure 3.3 which describes in a block diagram the Singular Value Decomposition algorithm's procedures. In the following section, the procedure details are explained, with the help of the diagram.

The video clip is first divided into video scenes  $V_{si}$  and the algorithm then divides part of this into coloured frames which are processed through the SVD matrix processing.

#### 3.4.1 Procedure for embedding watermark

The procedure involves the use of the SVD operator to transform the cover video files, inserting the message diagonally in the three  $S$ ,  $U$ , or  $V$  matrices. The algorithm that was used consisted of two procedures, where the initial phase involved the insertion of the message within the split frames of, while the next extraction. Only the pixels in the foreground were inserted within the frames.

The message insertion process as follows:

*Step 1:* Split the video streams into frames  $V_{si}$ .

*Step 2:* Using steps from 3 to 7 the frames are processed using SVD technique.

*Step 3:* The frames  $F_v$  are transformed into  $Y C_B C_R$  colour format from  $RGB$ .

*Step 4:* The  $Y$  matrix in each frame  $F_v$  is calculated to generate  $U, S, V$  matrices the after undergoing the SVD process which is made up of :

*Step1:* Split the watermarked Video clip  $V'$  into scenes  $V_{si}'$ .

*Step 2:* Using SVD process each watermarked video frames described in steps the following steps:

*Step 3:* The frames  $F'_v$  are converted to  $Y C_B C_R$  colour matrix format from  $RGB$  .

*Step 4:* The  $Y$  matrix in each frame  $F'_v$  is calculated to generate  $U, S, V$  matrices the after undergoing the SVD process.

*Step 5:* If it is desired to extract  $U$  from  $U, V$ , or  $S$  which in the SVD process, the following steps are followed:

$MV_{si} (i) = (\text{fix}(x))$  the 7th LSB

(6) Matrix  $V$  processing :

$MV_{si} (i) = (\text{fix}(x))$  the 7th LSB

(7) Matrix  $S$  processing :

$MV_{si} (i) = (\text{fix}(s i, i))$  the 7th LSB

(8) *Step 6:* Compose the frame message  $MV_{si}$  by combining all the extracted message bits

### 3.4.2 Block-wise embedding in Matrix U

*Step (i):*  $U$  matrix partitioned into blocks with size  $8 \times 8$ .

*Step (ii):* The inverse values ( $P_v$ ) for five prearranged picture elements( $x$ ) (pixels) for the  $U$  matrix in each odd block is obtained to satisfy the condition of  $x$  being the inverse of the pixel value the equation  $x = P_v^{-1}$ .

*Step (iii):* The binary bits of message  $MV_{si}$  is inserted in the integer part of  $x$  by replacing the message bit  $M_i$  with the 7<sup>th</sup> *LSB* bit of  $x$ .

*Step (iv):* To obtain the modified values of  $U$  matrix value invert each of the  $x$  elements.

*Step (v):* Apply inverse SVD Using the modified  $U$  coefficient matrix ' administer the inverted SVD process such that:

$$Y' = U_Y' S_Y V_Y^T$$

### 3.4.3 Block-wise embedding in Matrix V

*Step (i):*  $V$  matrix partitioned into blocks with size  $8 \times 8$ .

*Step (ii):* The inverse values ( $P_v$ ) for five prearranged picture elements( $x$ ) (pixels) for the  $V$  matrix in each odd block is obtained to satisfy the condition of  $x$  being the inverse of the pixel value the equation  $x = P_v^{-1}$  shown in Figure 3.4

*Step (iii):* The binary bits of message  $MV_{si}$  is inserted in the integer part of  $x$  by replacing the message bit  $M_i$  with the 7<sup>th</sup> *LSB* bit of  $x$ .

*Step (ii):* invert the pixel value for 5 predetermined pixels in each odd block shown in Figure 3.5 in each odd block in The  $V$  matrix such as  $x = 1/\text{pixel value}$ .

*Step (iv):* To obtain the modified values of  $V$  matrix value invert each of the  $x$  elements.

*Step (v):* Using the modified  $V'$  coefficient matrix use the SVD to obtain the  $Y'$  matrix such that:  $Y' = U_Y' S_Y V_Y$

Where the matrix  $Y'$  represents the new  $YCbCr$  colour luminance format.

*Step (vi):* Transform the frames  $F'$  from  $YCbCr$  to  $RGB$  colour matrix.

*Step (vii)* Reassemble frames into the final watermarked Video scene  $V_{si}'$ .

*Step (viii):* Reassemble watermarked scenes to get the final watermarked Video



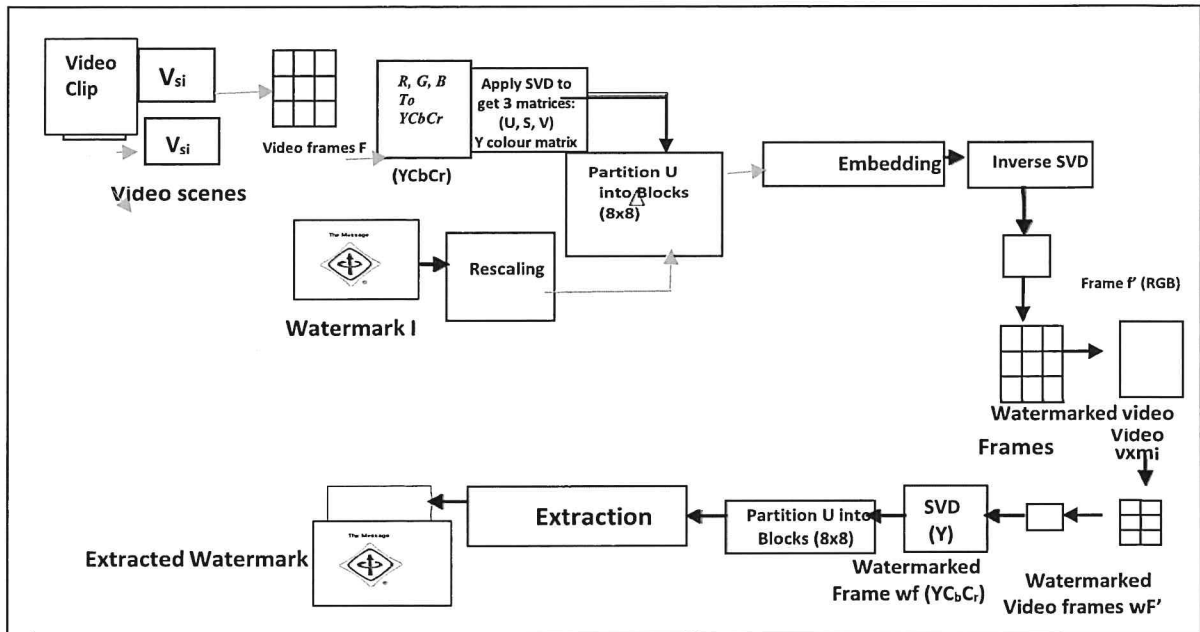


Figure 3.4: Illustration for the SVD process [74]

### 3.4.4 Matrix S Diagonal-wise embedding

- Insert  $MV_{si}$  message bits into each  $S$  matrix,  $s_i$  diagonal value through replacement of the  $M_i$  message bit with the 7th bit of  $si_i$
- Using the modified  $V'$  coefficient matrix use the SVD to obtain the  $Y'$  matrix such that:  

$$Y' = U_Y' S_Y V_Y$$

Where the matrix  $Y'$  represents the new  $YCbCr$  colour luminance format.

This process generates the final  $F'_v$  watermarked frame

Step 7: Transform the frames  $F'$  to  $RGB$  from  $YCbCr$  to colour matrix.

Step 8: Obtain scene  $V_{si}'$  from the reassembled frames.

Step 9: Obtain the final watermarked Video streams by reassembling watermarked frames.

### 3.4.5 Watermark Extraction Procedure

Step 1: Split the watermarked Video stream  $V'$  into frames  $V_{si}'$

Step 2: Apply (iii) to (viii) in 3.4.3 above transform the watermarked frames of each watermarked video stream

Step 3: Transform the frames  $F'_v$  to  $YCbCr$  colour matrix from  $RGB$ .

Step 4: The  $Y$  matrix in each frame  $F_v$  is calculated to generate  $U, S, V$  matrices the after

undergoing the SVD process

*Step 5:* The inverse values ( $P_v$ ) for five prearranged picture elements(x) (pixels) for the  $V$  matrix in each odd block is obtained to satisfy the condition of x being the inverse of the pixel value the equation  $x = P_v^{-1}$ .

*Step 6:* The inserted message is obtained from the integer part of x.

*Step 7:* Compose the frame message  $MV_{si}$  by combining all extracted message bits. from all frames.

*Step 8:* Repeat The same procedure is repeated for all the frames.

### **3.6 Hybrid SVD/DWT Watermarking Technique**

In hybrid SVD/DWT procedure, the host video was split into non-overlapping frames. The first step involved the transformation of the Red, Green, Blue colour space into the Y-luma components,  $C_b$  and  $C_r$ , the blue and the red difference Chroma components. Use is made of a two-dimensional DWT to execute the wavelet breaking down of the luminance  $Y$  components per frame of the host. An SVD is then performed, and these singular value matrices are used to build the watermarked video frames.

#### **3.6.1 Watermark embedding process:**

The following steps were taken during the process of embedding the watermark:

*Step 1:* The host video is split into groups of  $k$  frames. The number is determined by the frame rate and the normal rate is 24fps (fps-Frames per second), meaning that for the each second of video is composed of 24 distinct still images.

*Step 2:* Conversion is affected on the frames from the Red, Green and Blue into the Luma- $Y$ , Chromium Blue- $C_b$  and Red- $C_r$  differences colour space, processing being executed  $Y$  luminance values frames.

*Step 3:* Through use of  $L = 2$  resolution levels transform every  $Y$  luminance frame into the DWT framework.

*Step 4:* The Singular Value Decomposition is performed on the 2-D matrix of selected wavelet detail coefficients of the luminance  $Y$  for each frame of the group  $j$  of frames ( $Bi(j)$  matrix) of original video to obtain the SVs ( $Si(j)$  matrix), where  $i = 1, 2, 3, \dots, C$ , and  $C$  is the number of frames:

$$Bi(j) = Ui(j)Si(j)VT(j) \quad 3.11$$

*Step 5:* To protect the watermark against bit errors, a Hamming error correction code (there are others such as BCH(Bose–Chaudhuri–Hocquenghem), MDPC (multidimensional parity-check code) and Golay codes, however for this experiment, the Hamming code was more appropriate) with a code length  $m$ -bits of size  $w$ , where  $P = AxB$ , The size of the resulting watermark  $w_c$  is such that :

$$P' = P \frac{m}{n} \quad (3.1)$$

*Step 6:* The watermark  $w_c$  is partitioned into several  $C/k$  sequences  $w_c(j)$  of size

$$P' \frac{k}{c} \quad \text{and } j = 1, \dots, \frac{c}{k} \quad (3.2)$$

Where  $c$  is the number of video frames

*Step 7:* The same watermark sequence ( $w_c(j)$  matrix) is added to the  $S_i(j)$  matrix of each video frame of the group  $j$  of frames.

$$D_i(j) = S_i(j) + Kw_c(j) \quad (3.3)$$

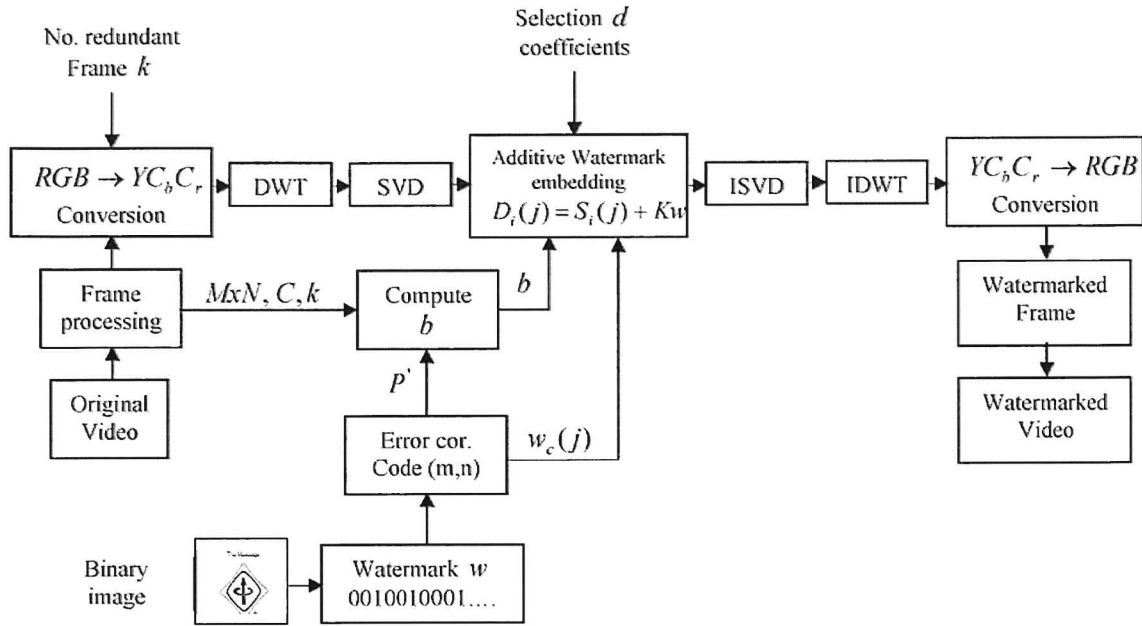
*Step 8:* The SVD is performed on each  $D_i(j)$  matrix of each video frame of the group  $j$  of frames to obtain the SVs of each one of the group  $j$  of frames ( $S_{wi}(j)$  matrix).

$$D_i(j) = U_{wi}(j)S_{wi}(j)VT(j) \quad (3.4)$$

*Step 9:* The  $S_{wi}(j)$  matrices of each one of the group  $j$  of frames are used to build the watermarked frames.

$$B_{wi}(j) = U_i(j)S_{wi}(j)VT(j) \quad (3.5)$$

*Step 10:* After the entire watermark has been embedded, the Inverse Discrete Wavelet Transform is computed to obtain the watermarked video, these activities are summarized in figure 3.5 that shows the block diagram of the SVD/DWT video watermark embedding process.



**Figure 3.5: Block diagram of SVD/DWT Watermark embedding process [74]**

### 3.6.2 Watermark extraction process

The process was executed through the following steps;

*Step 1:* The obtained watermarked video is divided into groups of  $k$  frames

*Step 2:* All video frames were converted from the  $RGB$  into the  $YCbCr$  colour space.

*Step 3:* Use was made of the Wavelet decomposition with  $L = 2$  resolution levels to transform the  $Y$  luminance frame into the DWT domain.

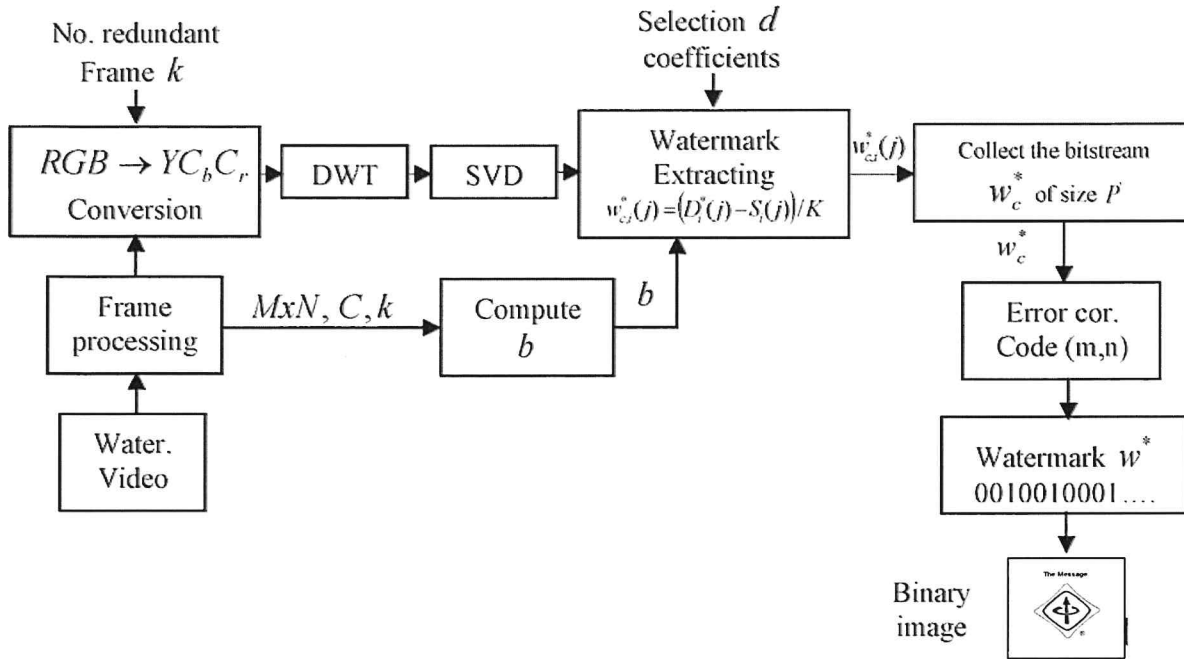
*Step 4(iv):* The watermarked wavelet coefficients are selected according to the model in Figure 3.7.

*Step 5:* The SVD is performed on each possibly distorted watermarked frame of the group  $j$  of frames ( $B_{wi}^*(j)$  matrix) to obtain the Singular Values of each one ( $S_{wi}^*(j)$  matrix).

$$B_{wi}^*(j) = U_i^*(j)S_{wi}^*(j)V_i^{*T}(j) \quad (3.6)$$

*Step 6(vi):* The matrices that may contain the watermark for each frame of the group  $j$  of frames are obtained using the  $U_{wi}(j)$ ,  $V_{wi}(j)$ ,  $S_{wi}^*(j)$  matrices.

$$D_i^*(j) = U_{wi}(j)S_{wi}^*(j)V^T(j) \quad (3.7)$$



**Figure. 3.6: Block diagram SVD/DWT Watermark extracting process [74]**

*Step 7:* From every frame, of a group of  $k$  frames a binary sequence of corrupted watermark  $w_{c,i}^*(j)$  is extracted from the  $D_i$  matrices, where  $i=1, k$ . The watermark sequence corresponding to the group  $j$  of frames is obtained using Eq. (3.8):

$$w_{c,i}^*(j) = \frac{D_i^*(j) - S_i(j)}{K} \quad (3.8)$$

*Step 8:* The resulting bit- stream  $w_{c,i}^*$  of size  $P'$  is error corrected resulting in the watermark  $w^*$  of size  $P$ .

*Step 9:* The extracted binary image is obtained by reshaping the vector  $w^*$  to a matrix of size  $A \times B$ .

*Step 10 :* The correlation coefficient between extracted  $w'$  and the original watermark is estimated. If at least one of the coefficients is higher than the threshold, then it is present. These activities are illustrated in the Block diagram SVD/DWT Watermark extracting process indicated in figure 3.8.

### 3.7 Metrics for Visual Quality

Several metrics have been proposed and continue to be used in assessing the quality of the outputs of experimentation in mainstream engineering that include PSNR, SSIM and Normalized Correlation (NC). All these metrics were used in the research investigation though Wang et al [75] have raised some substantial issues with PSNR where they reported it was

not a fair metric for determination of signal fidelity. However, they did indicate that PSNR approximates human perception of reconstruction quality, so it was considered appropriate to use it in this research investigation as a metric for determining the extent of the noise errors introduced during the various attacks.

PSNR is given as;

$$PSNR_k = 10 \log_{10} \frac{255^2}{MSE_k} \quad 3.21$$

$$\text{Where } \frac{1}{M \times N} \sum_{m=1}^M \sum_{n=1}^{LN} [f_k(m, n) - f_{kw}(m, n)]^2 \quad 3.22$$

where  $f_k$  is the k-th original video frame;  $f_{kw}$  is the k-th watermarked frame; SSIM is shown as:

$$SSIM(f_k, f_{kw}) = \frac{(2\mu_{f_k} \mu_{f_{kw}} + C_1)(2\sigma_{f_k} \sigma_{f_{kw}} + C_2)}{(\mu_{f_k}^2 + \mu_{f_{kw}}^2 + C_1)(\sigma_{f_k}^2 + \sigma_{f_{kw}}^2 + C_2)} \quad 3.24$$

where  $\mu_{f_k}$  and  $\mu_{f_{kw}}$  represent the average values of the host frame and the processed one respectively;  $\sigma_{f_k}$  and  $\sigma_{f_{kw}}$  are the variances of the host and the processed frame, respectively;  $\sigma_{f_k f_{kw}}$  show the denotes the covariance; and  $C_1, C_2$  are constants.

### 3.7.3 Preference for the Green Frames

Real-time images and videos are stored in *RGB* color space, which has a large bandwidth. In digital image processing the *YCbCr* color space is often used in order to take advantage of the lower resolution capability of the human visual system for color with respect to luminosity. Thus, *RGB* to *YCbCr* conversion is widely used in image and video processing. During the simulation the *RGB* Frames converted to *YCbCr* and use was made of the converted Green frames in preference to the other two components. This was premised on the fact that in a 24-bit colour image representation, each pixel is recoded with red, green, and blue components to achieve the *RGB* image. There are  $2^{24} = 16.777216 \times 10^6$  with non-uniform the distribution of the pixel's values with the distribution of red, green and blue frames being 66, 132 and 32 pixels per byte (8-bits) each respectively [89]. The green component is preferred as it has the highest number of pixels thus providing more data capacity than the other two. It also minimizes the payload with the net utilization of more resource's usage and hence mitigates perceptible distortion of the frames in the video after they are recombined. There are some issues with use of the Blue frames, attributable to what is known as the "blue light hazard" (BLH). [76].

# CHAPTER 4

## RESULTS AND DISCUSSIONS

### 4.1. Introduction to Simulations

This chapter presents the computer simulation outcomes of the suggested watermarking techniques. The video watermarking process was simulated in the MATLAB/Simulink environment through use SVD and SVD/DWT hybrid techniques. The results of Histogram equalization and Median Filtering attacks on each of the techniques and a summary is provided and the subsequent section deal with those generated from the attacks on the SVD/DWT hybrid schemes. This included Gaussian noise attacks at various noise standard deviation values, Impulse noise and finally JPEG Compression attacks on different video frames since they are not usually prone to RST attacks. In the Gaussian noise attacks, graphical analysis is carried out on the Bus frame with noise standard deviation from 0 to 1 with 0.1-unit levels and then from 1 to 2, and graphical analysis carried out. In the JPEG Compression exercise, compression quality ranges from 30% to 70% in steps of 5%. A graphical analysis is then carried out on each of these results to determine the trends, which forms the basis for the discussions and conclusions for each of the simulations.

Video watermarking was executed using SVD and its combination with DWT domains. The CIF and QCIF videos were in Red, Blue and Green colour space were broken down and transformed into the Luma, Chrominance Blue and Green of which from the generated from decomposed version the green components was chosen as the host image. The cover frames used in these experiments were the video clips, “Bus”, “Foreman”, “Container”, “Carphone”, “Coastguard”, “Akiyo” whose resolutions were CIF except the Carphone which was QCIF. These frames that were used as cover media were selected randomly, since the site Derf Collections has multitude of video clips. The watermark/message was the ‘ieee.jpg’ logo with pixel values ranging from 64x64 to 256x256 were also extracted after the split video frames were watermarked and subjected JPEG compression, Gaussian, and Impulse noise. All the algorithm for the SVD and the SVD/DWT were implemented using MATLAB (R2014b).

## 4.2 Experimental Results

Five experiments were carried out as follows.

- a) Video watermarking process without attacks,
- b) Four experiments where the logo was subjected to Signal processing attacks namely Histogram Equalization, Median Filtering, Gaussian Noise attack at various noise standards deviations, JPEG Compression attacks at various compression levels and Impulse Noise attack.

### 4.2.1 Experiment 1-Video Watermarking process without attacks.

The results SVD and the SVD and SVD/DWT are shown in table 4.1 indicate that the logo is extracted successfully, and the numerical results for four different frames from five clips.

**Table 4.1: First set of results of SVD and SVD/DWT Extracted Logo without attacks**

Quality Metric	VIDEO FRAME					
	BUS		AKIYO		CONTAINER	
	SVD	SVD/DWT	SVD	SVD/DWT	SVD	SVD/DWT
SSIM	0.95	0.99	0.90	0.99	0.98	0.99
PSNR	37.75	42.93	36.70	45.70	36.74	45.75
NC	0.95	0.996	0.990	0.999	0.996	0.994

**Table 4.2: Second sets of results of SVD and SVD/DWT Extracted Logo without attacks**

Quality Metric	VIDEO FRAME					
	COAST GAURD		CAR PHONE		FOREMAN	
	SVD	SVD/DWT	SVD	SVD/DWT	SVD	SVD/DWT
SSIM	0.95	0.99	0.90	0.99	0.90	0.97
PSNR	37.75	42.93	36.70	45.70	37.20	42.67
NC	0.95	0.996	0.990	0.999	0.990	0.993



The visual quality results indicate that the output values are quite high demonstrating the robustness of the algorithms for each of the techniques indicating minimal distortion on the extracted message from the original (NC values ranging from 0.93 to 0.95).

#### 4.2.2 Discussions on results of Experiment 1

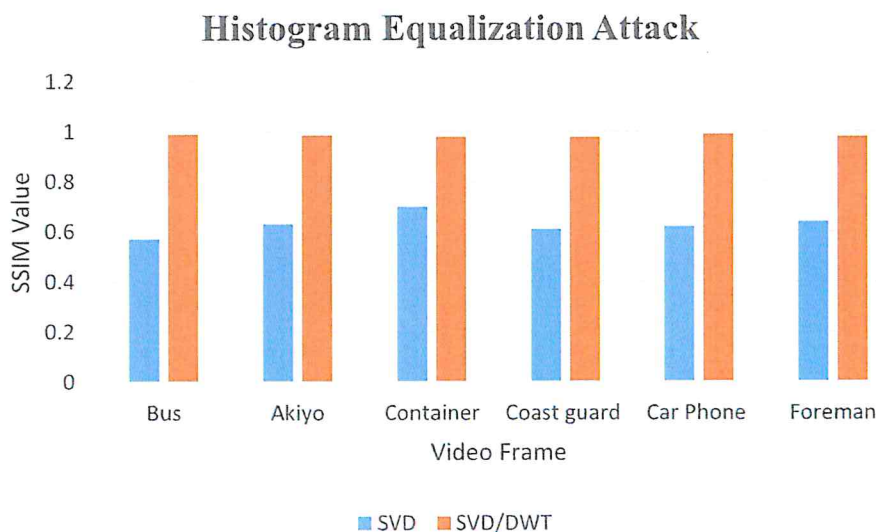
From Table 4.1 and 4.2 the average value of the watermark quality for the SVD watermarking technique considering SSIM output is 0.9325, and for the hybrid SVD/DWT was 0.985. Though the hybrid SVD/DWT was superior, the SVD technique performed well. The average PSNR values for the SVD technique were also lower than those of the hybrid SVD/DWT, though the NCC values were comparable for both techniques.

#### 4.3. Signal processing Attacks

There were five Signal processing namely Histogram equalization, Median filtering, Impulse Noise, additive Gaussian noising, and JPEG Compression.

##### 4.3.1 Experiment 2-Histogram equalization Attack

The results for the Histogram equalization indicate that they had a larger impact on the SVD as compared to SVD/DWT hybrid as is demonstrated by the results in Table 4.2. This premise don the fact that most of the results for hybrid SVD/DWT



**Figure 4.1: Results of Histogram Equalization Attack**

### 4.3.2 Discussions on Experiment 2 - Histogram equalization

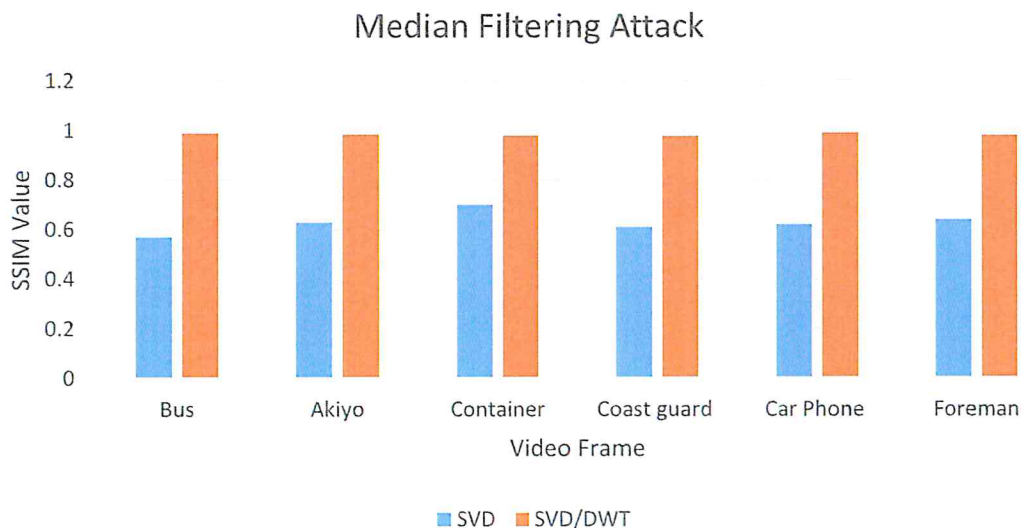
Figure 4.1 which depicted the results of the quality of the watermark after being subjected to Histogram equalization indicated that the SSIM values for the SVD technique ranged from 0.57 to a maximum of 0.78, whereas those for the hybrid SVD/DWT ranged from 0.95 to 0.99. This was a substantially poor performance, hence demonstrating the superiority of the hybrid SVD/DWT over the SVD technique.

### 4.3.3 Experiment 3- Median Filtering Attack

The results for the Median filtering attacks indicate that they had more impact on the SVD, compared to the hybrid of SVD/DWT as demonstrated by the results in Table 4.3.

### 4.3.4 Discussions on Experiment 3 Median Filtering Attack

The Median filtering attack results also showed a similar trend with the SSIM results for SVD ranging from 0.83 to 0.89 and those for the hybrid SVD/DWT technique ranging from 0.97 to 0.99, with the same trends seen in the PSNR and Normalized Cross Correlation. The results indicate that the hybrid SVD/DWT is far more resilient to the Median filtering attack than the SVD.



**Figure 4.2: Results of Median Filtering Attack**

**Table 4.3: Results for Histogram equalization Attack**

QUALITY METRIC	VIDEO FRAME											
	BUS		AKIYO		CONTAINER		COAST GUARD		CAR PHONE		FOREMAN	
	SVD	SVD/DWT	SVD	SVD/DWT	SVD	SVD/DWT	SVD	SVD/DWT	SVD	SVD/DWT	SVD	SVD/DWT
SSIM	0.57	0.99	0.63	0.986	0.70	0.98	0.61	0.978	0.62	0.99	0.64	0.98
PSNR	20.55	42.25	26.50	46.50	27.56	48.70	27.5	48.27	25.86	45.75	25.70	46.60
NC	0.81	0.951	0.87	0.965	0.88	0.980	0.86	0.974	0.87	0.976	0.85	0.973













**Table 4.4: Results Median Filtering Attack**





































QUALITY METRIC	VIDEO FRAME											
	BUS		AKIYO		CONTAINER		COAST GUARD		CAR PHONE		FOREMAN	
	SVD	SVD/DWT	SVD	SVD/DWT	SVD	SVD/DWT	SVD	SVD/DWT	SVD	SVD/DWT	SVD	SVD/DWT
SSIM	0.86	0.97	0.89	0.99	0.84	0.98	0.88	0.987	0.86	0.98	0.83	0.97
PSNR	37.10	45.93	38.5	48.2	37.31	47.60	38.2	47.9	37.50	42.50	37.20	41.24
NC	0.92	0.93	0.945	0.99	0.93	0.98	0.94	0.985	0.95	0.98	0.93	0.985

#### 4.3.5 Experiment 4-Gaussian Noise Attack on hybrid SVD/DWT technique

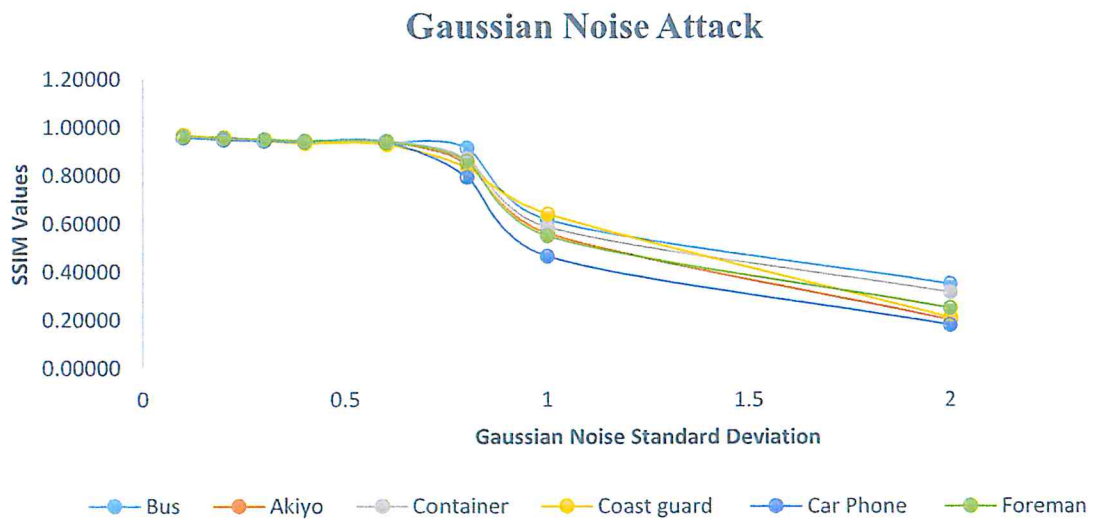
The Simulation experiment was carried out with a message size of size 128x128 and the results are detailed in Table 4.5 where the SSIM values and the extracted Watermarks have been shown.

**Table 4.5: Results for extracted watermark after Gaussian Noise Attack**

Standard Deviation	Quality Metric	Coast Guard	Car phone	Akiyo	Foremen	Bus	Container
0.1	SSIM	0.969	0.960	0.968	0.9650	0.970	0.962
	Extracted Watermark						
0.2	SSIM	0.958	0.950	0.960	0.957	0.959	0.956
	Extracted Watermark						
0.3	SSIM	0.9450	0.945	0.950	0.952	0.946	0.952

	Extracted Watermark						
0.4	SSIM	0.9350	0.942	0.944	0.945	0.944	0.946
	Extracted Watermark						
0.6	SSIM	0.93	0.940	0.941	0.942	0.938	0.944
	Extracted Watermark						
0.8	SSIM	0.833	0.794	0.85	0.860	0.915	0.870
	Extracted Watermark						
1.0	SSIM	0.641	0.466	0.5623	0.550	0.617	0.587
	Extracted Watermark						
2.0	SSIM	0.21	0.18	0.200004	0.25	0.350	0.315
	Extracted Watermark						

During the experiment Gaussian Noise attack with the S.D ranging from 0.1 to 2 in steps of 0.1, as shown in table 4.5 was used to demonstrate the robustness of the algorithm.



**Figure 4.3 Results of Gaussian Noise Attack**






















### 4.3.6 Discussion of the Results of Experiment 4-Gaussian Noise Attack

The results are graphically exhibited in Figure 4.3 and on analysis indicated an almost level value of SSIM tending to unity (1). However, from of a standard deviation of less than 0.7 the falls were steep when the deviation approached unity (1)1. The curves then reflected a smooth degradation towards the deviation of 2. Mapping these trends to Table 4.5 it is apparent that the watermarks begin to darken after unity standard deviation and almost obliterated when the standard deviation tends to 2.

### 4.3.7 Experiment 4- Impulse Noise Attack on hybrid SVD/DWT

Impulse noise is that which arises in video frames due to signal miss-transmission. The video frames were subjected to these attacks, and the frames that were used includes the bus, car phone, container, Foremen and Coast guard. The Noise density used were at 0.1 and 1.0 and the results are shown in Table 4.6.

**Table 4.6 Results of Impulse Noise Attacks on hybrid SVD/DWT Technique**

The Message 	Extracted message @ Noise Density =0.1 	SSIM	Extracted message @ Noise Density = 1.0 	SSIM
Bus 		0.9775		0.9760
Car Phone The Original Frame 		0.9778		0.91094
Container 		0.9681		0.9051
		0.9850		0.983
		0.979		0.9215
		0.998		0.986

From Table 4.6, there is not much effect of the attack since the SSIM values obtained on the quality of the watermark is almost 1. From Table 4.6, there is not much effect of the attack since the SSIM values obtained on the quality of the watermark is almost 1. Visual observations also

indicate that the Watermark is quite clear is detectable for all the frames that were subjected to the attack.














































**4.3.8 Discussions of the results of Experiment 4- Impulse Noise Attack**

The results as shown in Table 4.6 indicated that there is not much effect of the attack since the SSIM values obtained on the quality of the watermark is almost 1. Visual observations also indicate that the Watermark is quite clear is detectable for all the frames that were subjected to the attack.

**4.3.9 Experiment 5-JPEG Compression Attack on SVD/DWT technique**
























In the determination of the simulation results after subjecting the various frames and the messages after JPEG Compression attack the following parameters were computed. The PSNR of watermarked logo/message to determine the message degradation after the attack; PSNR of Watermarked video frame to determine the frame degradation quality after the attack; Structured Similarity Measure to assess the quality of the extracted message after attack and Normalized Cross Correlation to determine the extent of quality reduction on the video frames. Table 4.7 shows the output messages after each JPEG Compression attack.

**Table 4.7: First set of results of extracted logo after JPEG Compression attack**

JPEG Compression (%) The Message	30%	35%	40%	45%	50%	55%	60%	65%	70%	90%
										
Bus 			The Extracted Message 	The Extracted Message 	The Extracted Message 	The Extracted Message 	The Extracted Message 	The Extracted Message 	The Extracted Message 	The Message 
Car Phone 										The Message 
Container 										The Message 
Foreman 										The Message 

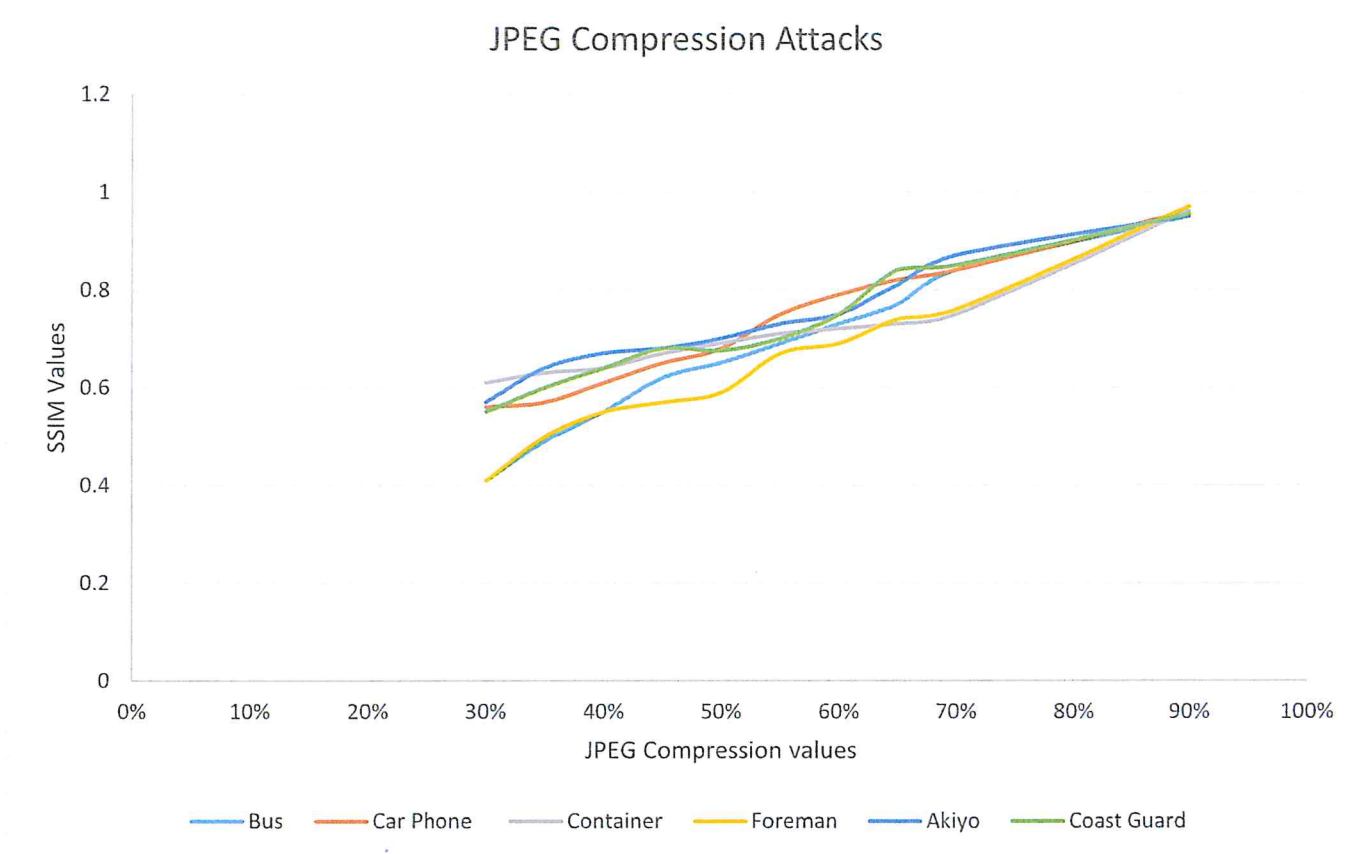
The Bus, Foreman and Container video frames were of the same quality, namely CIF Frames of 288x384 pixels and watermarked with a logo of 128x128 pixel, and the Car Phone was a QCIF frame of 144x192 pixels and watermarked with a logo of 64x64 pixels.

**Table 4.8: Second set of results of extracted logo after JPEG Compression attack**

JPEG Compression (%) The Message 	30%	35%	40%	45%	50%	55%	60%	65%	70%	90%
Akiyo 			The Extracted Message 	The Extracted Message 	The Extracted Message 	The Extracted Message 	The Extracted Message 	The Extracted Message 	The Extracted Message 	The Message 
Coast Guard 										The Message 

#### 4.3.10 Discussion of Experiment 5-JPEG Compression Attack on SVD/DWT technique

The results that are shown in Figure 4.4 clearly shows that as the compression increases, the SSIM value decreases towards the left side of the graph, and this applies to all the video frames. This means that at 100%, the JPEG Compression is zero, and it progresses towards the left. The trends indicate that as the compression increases towards the left, the SSIM and the NC tends to deteriorate proportionality. The extracted watermarks/logos shown in Tables 4.7 and 4.8 demonstrates the effects of the compression. They clearly show that as the compression increases, the watermark becomes darker. The algorithm was robust enough to provide outputs that were detectable even at 30% JPEG compression.



**Figure 4.4: Results of extracted Message after JPEG compression**



**Table 4.9 Results of extracted watermark after JPEG Compression Attacks**

JPEG Compression (%)	30%		35%		40%		45%		50%		55%		60%		65%		70%		90%		
	SSIM	N.C	SSIM	N.C	SSIM	N.C	SSIM	N.C	SSIM	N.C	SSIM	N.C	SSIM	N.C	SSIM	N.C	SSIM	N.C	SSIM	N.C	
Quality Metric																					
Bus-CIF Frame	0.41	0.46	0.49	0.51	0.55	0.57	0.62	0.65	0.59	0.65	0.61	0.69	0.73	0.68	0.77	0.72	0.84	0.79	0.95	0.99	
Car Phone - QCIF Frame	0.56	0.57	0.57	0.60	0.61	0.65	0.65	0.67	0.68	0.69	0.69	0.75	0.79	0.78	0.82	0.81	0.84	0.83	0.96	0.99	
Container CIF- Frame	0.61	0.62	0.63	0.64	0.64	0.65	0.67	0.67	0.69	0.68	0.68	0.71	0.72	0.74	0.73	0.75	0.75	0.78	0.96	0.985	
Foreman CIF- Frame	0.41	0.49	0.44	0.52	0.47	0.54	0.51	0.56	0.54	0.57	0.61	0.57	0.60	0.63	0.64	0.65	0.66	0.67	0.97	0.99	
Akiyo CIF Frame	0.57	0.63	0.64	0.65	0.67	0.66	0.68	0.65	0.70	0.72	0.72	0.73	0.75	0.80	0.81	0.84	0.87	0.88	0.95	0.9985	
Coast Guard CIF Frame	0.550	0.58	0.60	0.62	0.64	0.67	0.68	0.67	0.675	0.69	0.72	0.70	0.75	0.78	0.80	0.81	0.85	0.87	0.955	0.999	

#### 4.3.11 EXPERIMENT 6-VALIDATION METHODS FROM RELATED WORKS

In this concluding section, a comparison was made with the performance of the techniques used in the research investigation with three existing ones; Shukla et.al [51], Narasimhulu [53] and Jeebananda et.al [55]. In Table 4.10 values of similar outputs from simulation experiments carried out involving attacks on the hybrid SVD/DWT techniques were tabulated to carry out an analysis of the performance of the algorithms in this research investigation. From the Normal Correlation values in table 4.10, the results from the technique used in Jeebananda et.al [55] was able to withstand Gaussian noise attack. However, the scheme used in this research investigation demonstrated more superior effectiveness in resisting the same attack with the highest value almost equal to 1.

**Table 4.10: Comparison of Thesis Results with three other related Works**

	Shukla et.al [51]	Narasimhulu [53]	Jeebananda et.al [55]	Thesis results
Impulse Noise	0.92067	0.9816	0.9896	0.998
Gaussian Noise	0.41932	-	0.9454	0.970
Median Filter	-	-	0.9985	0.9995
Histogram Equalization	-	-	-	0.990
Compression@90%	0.95236	-	0.999	0.999

However, the method presented in Shukla et.al [51] has a poor robustness to Gaussian noise attack. Besides, both the technique used in the thesis and Jeebananda et.al [55] successfully survive the Median filter attack. The algorithms used in this thesis techniques returned better Normal Correlation value of one (1).

# CHAPTER 5

## CONCLUSION AND RECOMMENDATIONS

### 5.1 Evaluation of the Study

The results of the experiments in this research investigation have demonstrated that the hybrid SVD/DWT technique was superior to the SVD scheme., when subjected to Histogram equalization where the former had SSIM metric value ranged from 0.97 to 0.99 and the latter 0.57 to 0.70.

In the Median filter attacks the SVD scheme results had an average SSIM value of 0.86 compared to the hybrid SVD/DWT technique whose average was 0.98.

When the SVD/DWT technique was subjected to Gaussian noise, Impulse Noise and JPEG Compression, the algorithm was robust against these intrusions resulting in the detections of watermark at very severe compression levels like 30% for the case of JPEG Compression.

Comparison of the results from this research investigation three other previous works by Shukla et al [56], Narsimhulu [53] and Jeebananda et al [55] proved that the scheme proposed provided high level of robustness to Salt & Pepper noise having a Normalized correlation value of 0.998 and Gaussian Noise attack of 0.970. The JPEG compression value obtained in these works compared to the algorithms was more superior than the ones proposed in previous works by Jeebananda et al [55] and Shukla et al [51] with a normalized correlation value equal to 0.999. Considering the experimental results, the e conclusion is that the objectives of the research investigations were successfully met.

### 5.2 Scope for Future Work

This research investigation concludes that the algorithm used in hybrid SVD/DWT hybrid transform domain demonstrated better resilience from all the attacks and hence provide the best protection of videos transiting through the Internet space, except in extreme case of Gaussian Noise attack beyond Noise standard deviation higher than unity (1).

In this thesis investigation, SSIM, PSNR and NC (Normal Cross correlation) were employed as metrics to determine the visual quality of the watermarked video frames, and the criteria for robustness of the algorithm. For future works, it would be recommended that more advanced

metrics such as Universal Image Quality Index (UIQI) to establish how it best compares with SSIM.

It is recommended that means are made to reverse such severe distortions as was apparent in the use of Gaussian Noise attack beyond the noise standard deviation above unity (1). In addition, more work should be carried out on use of different wavelet such as (Morlet wavelet in contrast to the Haar wavelet that was used in this thesis. Finally, it is recommended that algorithms that emulate Subjective metric be prepared as image quality metrics, to bring out the real Human Visual Systems (HVS) analysis of the output from the watermarking techniques.

## REFERENCES

- [1] P.S Sethuraman, R. Srinivasan. Survey of Digital Video Watermarking Techniques and Its Applications. Engineering Science. Vol. 1, No. 1, December 2016, pp. 22-27. doi: 10.11648/j.es.20160101.14
- [2] K.H. Jung: A survey of reversible data hiding methods in dual images. IETE Tech. Rev. April 2016, 33, 441–452. [CrossRef]
- [3] B .Carpentieri, A .Castiglione, A.De Santis, F.Palmieri, R.Pizzolante, : One-pass lossless data hiding and compression of remote sensing data. Future Gener. Comput. Syst. January 2019, 90, 222–239. [CrossRef]
- [4] S.A. Parah, J.A .Sheikh, J.A .Akhoon, N.A .Loan, G.M. Bhat, :Information hiding in edges: A high capacity information hiding technique using hybrid edge detection. Multimedia Tools Appl. January 2018, 77, 185–207. [CrossRef]
- [5] Li, Fufang & Li, Binbin & Huang, Yongfeng & Feng, Yuanyong & Peng, Lingxi & Zhou, Naqin.). Research on covert communication channel based on modulation of common compressed speech codec. Neural Computing and Applications. April,2020, 10.1007/s00521-020-04882-y.
- [6] H.X Zhang,Z.Y Zhang,. P.L Qiu: A novel algorithm of covert communication., Acta Electron. Sin. 2003, 31, 514–517.
- [7] R. Bala ‘A Brief Survey on Robust Video Watermarking Techniques’ The International Journal Of Engineering And Science (IJES) Volume 4 Issue 2 Pages PP.41-45, February 2015.
- [8] P. D. Sonawane<sup>1</sup> , S. S. Mane , R. N. Nazirkar , P. P. Barhalikar , . A. Verma ‘A Survey on Efficient Video Watermarking and Image Data Encryption Technique’ Vol. 4, Issue 10, October 2016
- [9] R. Rawat, N. Kaushik, S. Tiwari ‘Digital Watermarking Techniques’ , International Journal of Advanced Research in Computer and Communication Engineering Vol. 5, Issue 4, April 2016
- [10] J. K. Saini and H. K. Verma, “A hybrid approach for image security by combining encryption and steganography,” in Image Information Processing (ICIIP), 2013 IEEE Second International Conference pp. 607–611, 9<sup>th</sup> -11<sup>th</sup> December 2013,Waknaghat, Shimla, Himachal Pradesh, INDIA
- [11] G. Qiu, P. Marziliano, A.T.S.HO, D.He, and Q.Sun, “A hybrid watermarking scheme for H.264/AVC video”, Proceedings of the 17th International Conference on Pattern Recognition (ICPR’04), Cambridge, England, U.K, 2004
- [12] C. Li, Y. Wang, B.Ma, Z. Zhang “Multi-block dependency based fragile watermarking scheme for fingerprint images protection” Springer ScienceBusiness Media, LLC 2012
- [13] X Li, X. J Wang,.; W, MYang, X,Wang. “A robust video watermarking scheme to scalable recompression and transcoding”. In Proceedings of the International Conference on Electronics Information and Emergency Communication, Beijing, China, 17–19 June 2016.

- [14] P.S. Sethuraman, R.Srinivasan, Survey of Digital Video Watermarking Techniques and Its Applications, Engineering Science. Vol. 1, No. 1, October,2016, pp. 22-27. doi: 10.11648/j.es.20160101.14
- [15] N.Tiwari and Sharmila: Digital Watermarking Applications, Parameter Measures and Techniques, IJCSNS International Journal of Computer Science and Network Security, VOL.17 No.3, March 2017
- [16] M. Shahid and P. Kumar :Digital Video Watermarking: Issues and Challenges, International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 7, Issue 4, April 2018, ISSN: 2278 – 1323
- [17] R. Artru , L. Roux and T. Ebrahimi: Digital watermarking of video streams: review of the state-of-the-art, arXiv:1908.02039v2 [eess.IV] 23 Aug 2019
- [18] D. Kumar R. A , Dinesh S , S.Kumar D , Chitra R: Prevention of Video Piracy through Advanced Video Watermarking, Vol. 7, Special Issue 2, March 2018 International Journal of Innovative Research in Science, Engineering and Technology
- [19] C.Panyindee. ; C. Pintavirooj, . Reversible watermarking algorithm in application for medical images” Proceedings of 5th Biomedical Engineering International Conference (BMEiCON), Ubon Ratchathani, Thailand-2012,Page(s):1 – 5 ISBN:978-1-4673-4890-4, December 2012.
- [20] D. Zheng., Y.Liu., J.Zhao., and A.E. Saddik. “A survey of RST invariant image watermarking algorithms”, ACM Computing Surveys, Volume 39, No. 2, Article 5, June 2007.
- [21] I.J. Cox, M.L. Miller, J.A. Bloom, J. Fridrich, and T.Kalker, “Digital Watermarking and Steganography, 2nd Edition”, San Francisco, CA, USA.: Morgan Kaufmann Publishers, 2008.
- [22] S. Rashmi, “DWT based Invisible Watermarking on Images”- International Journal of Advance Research, Ideas and Innovations in Technology, ISSN: 2454-132X Impact factor: 4.295 (Volume3, Issue1) January-February 2017
- [23] T. Zong, Y. Xiang1 S. Elbadry, S.Nahavandi-Modified Moment-based Image Watermarking Method Robust to Cropping Attack; International Journal of Automation and Computing 13(3), June 2016 .
- [24] Y.Varshney, “Attacks on Digital Watermarks: Classification, Implications, Benchmarks” International Journal on Emerging Technologies (Special Issue NCETST-2017) 8(1): 229-235(2017) (Published by Research Trend, Website: www.researchtrend.net)
- [25] Y.-S. Lee, Y.-H. Seo , and D.-W. Kim: Blind Image Watermarking Based on Adaptive Data Spreading in n-Level DWT Subbands, Hindawi Security and Communication Networks Volume 2019, Article ID 8357251, 11 pages <https://doi.org/10.1155/2019/8357251>
- [26] C.Shoemaker, “Hidden Bits: Survey of Techniques for Digital Watermarking,” Independent Study, Springer 2002
- [27] L,T.Rajab, T. Al-Khatib, and A. Al-Haj (2009). Video Watermarking Algorithms Using the SVD Transform. European Journal of Scientific Research ISSN 1450-216X Vol.30 No.3 (2009), pp.389-401 © EuroJournals Publishing, Inc. 2009 <http://www.eurojournals.com/ejsr.htm>
- [28] D.G Savakar,, A Ghuli,. Robust Invisible Digital Image Watermarking Using Hybrid Scheme. Arab J Sci Eng 44, 3995–4008 February 2019 . <https://doi.org/10.1007/s13369-019-03751-8>

- [29] R. Liu and T. Tan, "A SVD-Based Watermarking Scheme for Protecting Rightful Ownership", *IEEE Transactions on Multimedia*, 4(1), March 2002, pp.121-128.
- [30] J. Liu, X. Niu and W. Kong, "Image Watermarking based on Singular Value Decomposition", *Proceedings of the 2006 International Conference on Intelligent information Hiding and Multimedia Signal Processing (IIH-MSP'06)*, pp. 457-460, 2006, Pasadena, California USA
- [31] E. Ganic and A. M. Eskicioglu, "Secure DWT-SVD Domain Image Watermarking: Embedding Data in All Frequencies," *ACM Multimedia and Security Workshop 2004*, Magdeburg, Germany, September 20-21, 2004.
- [32] R. Agarwal "Block based digital watermarking using Singular Value Decomposition on Colour Images"- *International Conference on Computing, Communication & Automation -2015* 6 July 2015
- [33] Z. Zhou, B. Tang and X. Liu, "A Block-SVD Based Image Watermarking Method", *Proceedings of the 6th World Congress on Intelligent Control and Automation*, June 21 - 23, 2006, Dalian, China
- [34] N. Singh, S. Joshi and S. Birla, "Suitability of Singular Value Decomposition for Image Watermarking," *2019 6th International Conference on Signal Processing and Integrated Networks (SPIN)*, Noida, India, 2019, pp. 983-986, doi: 10.1109/SPIN.2019.8711749.
- [35] A Kerbiche, A.; Jabra, S.B.; Zagrouba, E.; Charvillat, V. A robust video watermarking based on feature regions and crowdsourcing. *Multimedia Tools Appl.* 2018, 77, 26769–26791. [CrossRef]
- [36] A.Al-Haj, "Combined DWT-DCT Digital Image Watermarking" *Journal of Computer Science*, Vol 3, No.9, pp. 740-746, 2007, 2007 ISSN 1549-3636 © 2007 Science Publications
- [37] M.Masoumi.; , S. A ,Amiri blind scene-based watermarking for video copyright protection. *AEU-Int. J. Electron. Commun.* June 2013, 67, 528–535. [CrossRef]
- [38] P Meerwald, "Digital Image Watermarking in the Wavelet Transform Domain", *MSc thesis in University of Salzburg*, 2001.
- [39] D.Y Jiang,.; D.Li,; J.W.Kim, A spread spectrum zero video watermarking scheme based on dual transform domains and log-polar transformation. *Int. J. Multimedia Ubiquitous Eng.* Vol.10, No.4 (2015). pp. 367–378. [CrossRef]
- [40] P. D. Sonawane1 , S. S. Mane , R. N. Nazirkar , P. P. Barhalikar , . A. Verma 'A Survey on Efficient Video Watermarking and Image Data Encryption Technique' Vol. 4, Issue 10, October 2016
- [41] S Jindal,.; S Goel,.; T Puri,.; , A Bhardwaj,; I.Mahant,; S Singh,.; D Sood,. Performance analysis of LSB based watermarking for optimization of PSNR and MSE. *Int. J. Secur. Its Appl.* Vol10, No.3(2016).pp. 345–350. [CrossRef]
- [42] I.J Cox,.; J.Kilian,; F.T Leighton,.; T Shamoon,. Secure spread spectrum watermarking for multimedia. *IEEE Trans. Image Process.* 1997, 6, 1673–1687. [CrossRef] [PubMed]
- [43] H.Kaur,.; E.V Kaur,. Invisible video multiple watermarking using optimized techniques. In *Proceedings of the Online International Conference on Green Engineering and Technologies*, Coimbatore, India, 19 November 2016.







- [44] I Bayouhdh,.; S.B Jabra,.; E Zagrouba,. Online multi-sprites based video watermarking robust to collusion and transcoding attacks for emerging applications. *Multimedia Tools Appl.* July 2017, 77, 14361–14379. [CrossRef]
- [45] M Masoumi,.; M Rezaei,.; A.B Hamza,.A blindspatio-temporal data hiding for video ownership verification in frequency domain. *AEU-Int. J. Electron. Commun.* Vol 69, Issue 12, December 2015, Pages 1868-1879. [CrossRef]
- [46] R.O Preda,.; N. Vizireanu, New robust watermarking scheme for video copyright protection in the spatial domain. *UPB Sci. Bull.* 2011, 73, 93–104.
- [47] Z Bahrami, F A Tab, “A new robust video watermarking algorithm based on SURF features and block classification. *Multimedia Tools Appl.* January 2018, 77, 327–345. [CrossRef]
- [48] X.,Li,; X.,J Wang,.; W.M Yang,.; X ,Wang,. A robust video watermarking scheme to scalable recompression and transcoding. In *Proceedings of the International Conference on Electronics Information and Emergency Communication*, Beijing, China, 17–19 June 2016.
- [49] M.Othmani, , W.Bellil, , B Amar, C., and A. M.Alimi, A new structure and training procedure for multi-mother wavelet networks. *International Journal of Wavelets, Multiresolution and Information Processing (IJWMIP)*, 8(1):pp.149-175. January 2010). ISSN (print): 0219-6913 | ISSN (online): 1793-690X
- [50] S Mostafa,., and A Ali,. (2016). A Multiresolution video watermarking algorithm exploiting the block-based motion estimation. *Journal of Information Security*, 2016, 7(4), pp.260-268 Published Online July 2016 in SciRes. <http://www.scirp.org/journal/jis>
- [51] D., ShuklaM Sharma,. Robust Scene-Based Digital Video Watermarking Scheme Using Level-3 DWT: Approach, Evaluation, and Experimentation. *Radioelectron.Commun.Syst.* 61, 1–12 (2018). Published12 March 2018, <https://doi.org/10.3103/S0735272718010016>
- [52] T. T. Nguyen and D. Dinh Nguyen, "A robust blind video watermarking in DCT domain using even-odd quantization technique," *2015 International Conference on Advanced Technologies for Communications (ATC)*, Ho Chi Minh City, 2015, pp. 439-444, doi: 10.1109/ATC.2015.7388367.
- [53] C. V. Narasimhulu, "A robust hybrid video watermarking algorithm using NSCT and SVD," *2017 IEEE International Conference on Power, Control, Signals and Instrumentation Engineering (ICPCSI)*, Chennai, 2017, pp. 1495-1501, doi: 10.1109/ICPCSI.2017.8391961.
- [54] A. Naved: “A robust video watermarking technique using DWT, DCT, and FFT”. *International Volume 6(6): pp 490-494.*, Issue 6, June 2016 ISSN: 2277 128X *International Journal of Advanced Research in Computer Science and Software Engineering*
- [55] P Jeebananda,., and Prince G. (2016). An efficient video watermarking approach using scene change detection, pp.1-5, *Proceedings of 1st India International Conference on Information Processing (IICIP)*, Delhi, India 12<sup>th</sup> -14<sup>th</sup> August 2016,
- [56] R. Reyes, C. Cruz, M. Nakano-Miyatake, H. Pérez-Meana, Digital video watermarking in DWT domain using chaotic mixtures, *IEEE Lat. Am. Trans.* 8 (3) (2010) 304–310.



- [57] K Ahmed, I El-Henawy, A Atwan, "Novel DWT video watermarking schema" *Machine Graphics & Vision International Journal*, Volume 18 Issue 3, February 2009-Pages 36338
- [58] Y. Raghavender Rao, "Video Watermarking Algorithms Using the Svd Transform" *International Journal of Engineering Research and Development*-ISSN: 2278-067X, p-ISSN: 2278-800X, www.ijerd.com Volume 8, Issue 11 (October 2013), PP. 01-08
- [59] G M. El-Gayyar, "Watermarking Techniques Spatial Domain Digital Rights Seminar", Media Informatics University of Bonn Germany 2006
- [60] M. Kaur, S. Jindal, S. Behal, "A Study of Digital image watermarking", *The IJES* Volume 2, Issue 2, 2012. www.theijes.com
- [61] V. M. Potdar, S. Han and E. Chang, "A survey of digital image watermarking techniques," *INDIN '05. 2005 3rd IEEE International Conference on Industrial Informatics*, 2005., Perth, WA, Australia, 2005, pp. 709-716, doi: 10.1109/INDIN.2005.1560462.
- [62] C.C Chang, P. Tsai, C.C. Lin "SVD-based digital image watermarking scheme"- *ACM Digital Library*, Volume 26 Issue 10, July, 2005 Pages 1577-1586 -Elsevier Science Inc. New York, NY, USA
- [63] E. Brannock, M. Weeks and R. Harrison, "Watermarking with wavelets: Simplicity leads to robustness," *IEEE SoutheastCon 2008*, Huntsville, AL, 2008, pp. 587-592, doi: 10.1109/SECON.2008.4494361.
- [64] M. Kociołek, A. Materka, M. Strzelecki P. Szczypiński Discrete wavelet transform – derived features for digital image texture analysis, *Proc. of International Conference on Signals and Electronic Systems*, 18-21 September 2001, Lodz, Poland, pp. 163-168.
- [65] J. F. Delaigle, C. Devleeschouwer, B. Macq and L. Langendijk, "Human visual system features enabling watermarking," *Proceedings. IEEE International Conference on Multimedia and Expo*, Lausanne, Switzerland, 2002, pp. 489-492 vol.2, doi: 10.1109/ICME.2002.1035653.
- [66] P. K. Dhar, M. I. Khan, S. Ahmad, "A New DCT - Based Watermarking Method For Copyright Protection of Digital Audio", (*IJCSIT*), Vol.2, No. 5, October 2010
- [67] A. Shaamala, M Shahidan. A .Abdullah., A.Manaf, "Study of the effect DCT and DWT domains on the imperceptibility and robustness of Genetic Watermarking", *International journal of computer science issue*, Vol.8, issue 5, No.2, September 2011.
- [68] P. Dabas, K. Khanna, "A Study on Spatial and Transform Domain Watermarking Techniques", *International journal of computer application*, vol.71, No.14, pp. 38-41, June 2013
- [69] S.Voloshynovskiy, S.Pereira, S. T Pun,. J.J Eggers,. J.K Su, , "Attacks on digital watermarks: classification, estimation-based attacks, and benchmarks", *IEEE Communications Magazine*, Vol. 39, pp. 118–126, August 2001.
- [70] P Singh, R S Chadha, "A Survey of Digital Watermarking Techniques, Applications and Attacks", *IJEIT*, Vol 2, Issue 9, March 2013
- [71] E Hussein, A M. Belal, "Digital Watermarking Techniques, Applications and Attacks Applied to Digital Media: A Survey", *IJERT*, Vol. 1, Issue 7, September 2012 .

- [72] xiph.org, Derf's test media collection (March 2013), <http://media.xiph.org/video/derf/>
- [73] P.V. Powar, S.S.Agrawal "Design of digital video watermarking scheme using MATLAB SIMULINK", IJRET: International Journal of Research in Engineering and Technology, Volume: 02 Issue: 05 | May-2013
- [74] O.S. Faragallah, "Efficient video watermarking based on singular value decomposition in the discrete wavelet transform domain", International Journal of Electronics and Communications (AEÜ), AEUE-50941; 67(3) .pp (189-196) (2013)
- [75] Z Wang and A C. Bovik-Mean Squared Error: Love It or Leave It? [A new look at signal fidelity measures] -Digital Object Identifier 10.1109/MSP.2008.930649 IEEE Signal Processing Magazine January 2009
- [76] ICNIRP Guidelines on Limits of exposure to incoherent visible and infrared radiation. Health Physics. 105(1):74-96; 2013 (available from [www.icnirp.org](http://www.icnirp.org)).

## APPENDIX A: VIDEO DATA USED IN SIMULATIONS

Video Frame	Number of Frames	Frames per Second	Format	First Frame
Costgaurd.avi	300	30	CIF (44MB)	
Container.avi	300	30	CIF (44MB)	 The Original Frame
Akiyo.avi	300	30	CIF (44MB)	
Foreman.avi	300	30	CIF (44MB)	 The Original Frame
bus.avi	150	30	CIF (22MB)	 The Original Frame
carphone.avi	382	30	QCIF (14MB)	 The Original Frame

## APPENDIX B: MATLAB PROGRAMS

### APPENDIX-MATLAB CODES FOR SIMULATION RESULTS

#### 1. SINGULAR VALUE DECOMPOSITION (SVD)

```
% Authors: VINCENT ADUL, 2018 University of Nairobi
%Embedding
%-----
clear;
clc
tic
% read in the cover object
imag=imread('frame001.jpg');
red=imag(:,:,1);
blue=imag(:,:,3);
file_name=imag(:,:,2);
cover_object=double(file_name);

%imag= imread('cameraman.tif');
%imag=imresize(imag, [256 256]);
%[M,N]=size(cover_object);
cover_object=double(cover_object);
[UIm,SIm,VIm]=svd(cover_object);
Simg_temp=SIm;

% read message
message= imread('ieee_logo.tif');
% message= imread('logomy.bmp');
%message=imresize(message, [256 256]);
message=imresize(message, [128 128]);
k= 0.05; %Choose the embedding factor
[x y]=size(message);
message=double(message);
for i=1:x
    for j=1:y
        SIm(i,j) =SIm(i,j) + k * message(i,j);
    end
end
% SVD for SIm
[U_SHL_w, S_SHL_w, V_SHL_w]=svd(SIm);
WImG =UIm* S_SHL_w * VIm';
WIm = cat(3, red, WImG, blue);
figure(1)
imshow(uint8(imag));
title('The Original Frame')
figure(2)
imshow(uint8(message));
title('The Message')
figure(3)
imshow(WIm);
title('The Watermarked Frame')

% display psnr of watermarked image
psnr=psnr2(cover_object, WImG, 256);
```

```

%-----
% %% Extraction
%-----

%Split the frame into the various colour components
red=WIm(:,:,1);
blue=WIm(:,:,3);
WImG=WIm(:,:,2);
%Embedding component chosen (green)
WImG=double(WImG);
[UWimg,SWimg,VWimg]=svd(WImG);
D_1=U_SHL_w * SWimg * V_SHL_w';
for i=1:x
    for j=1:y
        Watermark(i,j)= (D_1(i,j) - Simg_temp(i,j) )/k ;
    end
end
figure(4)
imshow(uint8(Watermark));
title('The Extracted Message')

%SSIM- Structural Similarity Index Measure
%Function is done by Zhou Wang of Univ. of Waterloo
[MSSIM ssim_map] = ssim_index(cover_object,WImG);
MSSIM

%Calculating the Normalised Cross-correlation value
nc=nxc(message,Watermark);

toc

```

## 2. DISCRETE WAVLET TRANSFORM/SINGULAR VALUE DECOMPOSITION ( SVD/DWT) HYBRID

```

% Author: Vincent Adul, %University of Nairobi, 2018
%Embedding
%
%-----

```

```

clear;
clc

```

```

% read in the cover object
imag=imread('frame001.jpg');
red=imag(:,:,1);
blue=imag(:,:,3);
file_name=imag(:,:,2);
cover_object=double(file_name);
[M,N]=size(cover_object);
k = 0.05;

```

```

% DWT transform
[cA1,cH1,cV1,cD1] = dwt2(cover_object,'haar');

```

```

% read message
message= imread('ieee_logo.tif');
message=double(message);
message=imresize(message,[128 128]);
[x y]=size(message);
[UIm,SIm,VIm]=svd(cD1);
Simg_temp=SIm;

for i=1:x
    for j=1:y
        SIm(i,j) =SIm(i,j) + k * message(i,j);
    end
end

[U_SHL_w,S_SHL_w,V_SHL_w]=svd(SIm); %SVD of SIm

%Inverse SVD
wm_SIm =UIm* S_SHL_w * VIm';

%IDWT
WIm_G = idwt2(cA1,cH1,cV1,wm_SIm,'haar',[M,N]);

%Concatenate the Picture
WIm = cat(3,red,WIm_G,blue);

figure(1)
imshow(uint8(imag));
title('The Original Frame')
figure(2)
imshow(uint8(message));
title('The Message')
figure(3)
imshow(WIm);
title('The Watermarked Frame')

% display psnr of watermarked image
psnr=psnr2(cover_object,WIm_G,256);

%-----
% %% Extraction
% -----

%Split the frame into the various colour components
red=WIm(:,:,1);
blue=WIm(:,:,3);
WImG=WIm(:,:,2);
%Embedding component chosen (green)
WImG=double(WImG);
%WIm=double(WIm);

% DWT transform
[cAw1,cHw1,cVw1,cDw1] = dwt2(WImG,'haar');

[UWimg,SWimg,VWimg]=svd(cDw1);

```

```

%SVD of Horizontal component
D_1=U_SHL_w * SWimg * V_SHL_w';

%Extract the watermark
for i=1:x
    for j=1:y
        Watermark(i,j)= (D_1(i,j) - Simg_temp(i,j) )/k ;
    end
end
figure(4)
imshow(uint8(Watermark));
title('The Extracted Message')

%SSIM- Structural Similarity Index Measure
%Function is done by Zhou Wang of Univ. of Waterloo
[MSSIM ssim_map] = ssim_index(cover_object,WIm_G);
MSSIM

%Calculating the Normalised Cross-corelation value
nc=nxc(message,Watermark);

```

### 3. FUNCTION TO ADD ADDITIVE WHITE GAUSSIAN NOISE

```

%% Copyright(c) Naushad Ansari, 2017.
% %% Please feel free to use this open-source code for research purposes
only.
% %%
% %% contact at naushadansari09797@gmail.com.
% %%
% %%
% %% This function adds additive white Gaussian noise (with zero mean and
% %% given snr) to a signal. Signal can be any n-D signal.
%%-----%%
%%-----%%
% %% output: noisySig-> resultant n-dimensional noisy signal.
%
% %% input:  sig-> original n-dimensional signal
%           reqSNR-> required/given SNR of the noise, to be added in the
%           given signal.
%%-----%%
%%-----%%
function noisySig = addGaussianNoise(sig,reqSNR)
sigEner = norm(sig(:))^2;           % energy of the signal
noiseEner = sigEner/(10^(reqSNR/10)); % energy of noise to be added
noiseVar = noiseEner/(length(sig(:))-1); % variance of noise to be
added
noiseStd = sqrt(noiseVar);         % std. deviation of noise to
be added
noise = noiseStd*randn(size(sig)); % noise
noisySig = sig+noise;              % noisy signal

```

## IMPULSE NOISE ATTACK

```
% Author Deepak M.S.I.T, 09 Oct 2010.

% Just provide the path of the image to add the noise or to restore the
% image.
clc;
f = imread('Place the path of your image file');
figure
imshow(f),title('Original Image')
[M N] = size(f);
% Any type of noise can b added to the image provided. Just see the
% imnoise2 file to see the various noise effects. Here an example is shown
% where we are adding the pepper noise to the image.
r = imnoise2('salt & pepper',M,N,0,0.1);
figure
imshow(r),title('Noise to be added');
c = find(r == 1);
gp = f;
gp(c) = 255;
figure
imshow(gp),title('Image after adding the Noise');
% The image distorted by adding the noise can be restored by using the
% function imrest. Here an example is shown by using contraharmonic
% filter.
% you can use various type of other filter to restore the image. For
% more
% detail just see the funcion imrest.
fp = imrest(gp,'chmean',3,3,-5.5);
figure
imshow(fp),title('Image after restoration.')
```

## 4. JPEG COMPRESSION FUNCTION

```
%% Copyright(c) Vincent Adul, University of Nairobi 2019.
%% %% Please feel free to use this open-source code for research purposes
only.
%% %% %%
%% %% This function is for JPEG Compression
%% %% given any video frame as described below

function y=jcomp(x,quality)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%% Image compression
% y=jcomp(x,q)
% x is the image to be compressed
% q is the quality of compressed image default value = 100
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
% Check the arguments

if nargin < 2
    quality = 100; % default value for quality
end
if quality <= 0
    error('Input parameter QUALITY must be greater than zero.');
```



```

end
if quality > 100
    error('Input parameter QUALITY must be less than or equal to 100.');
```

```

end

m = [16 11 10 16 24 40 51 61           % default JPEG quantization array
     12 12 14 19 26 58 60 55
     14 13 16 24 40 57 69 56
     14 17 22 29 51 87 80 62
     18 22 37 56 68 109 103 77
     24 35 55 64 81 104 113 92
     49 64 78 87 103 121 120 101
     72 92 95 98 112 100 103 99] .* quality;

% defining the operations on the 8x8 blocks
f1=@(block_struct) dct2(block_struct.data);
f2=@(block_struct) idct2(block_struct.data);

m = round(100-quality);

%imwrite(I, 'new.tif');
red=x(:, :, 1);
I=x(:, :, 2);
blue=x(:, :, 3);
%I=x;
%figure, imshow(I)
J=blockproc(I, [8 8], f1);
%figure, imshow(J)
depth = find(abs(J) <m);
J(depth) = zeros(size(depth), 'uint8');
kf=blockproc(J, [8 8], f2)/255;
Gr=im2uint8(kf);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%figure, imshow()
RedJ=blockproc(red, [8 8], f1);
figure, imshow(J)
depth = find(abs(RedJ) <m);
RedJ(depth) = zeros(size(depth), 'uint8');
Redkf=blockproc(RedJ, [8 8], f2)/255;
redComp=im2uint8(Redkf);
%figure, imshow(WImGr)
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%figure, imshow()
BlueJ=blockproc(blue, [8 8], f1);
%figure, imshow(J)
depth = find(abs(BlueJ) <m);
BlueJ(depth) = zeros(size(depth), 'uint8');
Bluekf=blockproc(BlueJ, [8 8], f2)/255;
blueComp=im2uint8(Bluekf);
%figure, imshow(WImGr)

y = cat(3, redComp, Gr, blue);
%y = cat(3, CompRed, WImGr, CompBlue);
figure, imshow(y)
title('The JPEG compressed image')
%imwrite(y, 'newc.tif');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%

```

## 5. FUNCTION FOR DETERMINING MSE AND PSNR

```
% Author: Vincent Adul, University of Nairobi 2017
%function for finding MSE and PSNR
% This is a standard Matlab/Simulink function for determining the
% Mean squared error and Peak Signal to Noise ration
%
function p=psnr2(x,xc,g)
x=double(x);
xc=double(xc);
t=(x-xc).^2;
%disp('s');
s=sum(sum(t));

%p=g*g*(size(x,1)*size(x,2))/(s);
n=size(x,1)*size(x,2);

MSE=s/n;

disp('PSNR = ');
p=(g*g)/(s/n);
p=10*log10(p);
disp(p);
```

## 6. FUNCTION REGION SELECT

```
function varargout = regionselect(lbl, pred, varargin)
% Author Vincent Adul University of Nairobi 2011
%PIC = REGIONSELECT(LABELED_IMAGE, PREDICATE, OPTIONS)
% [PIC, N] = REGIONSELECT(LABELED_IMAGE, PREDICATE, OPTIONS)
% Using PREDICATE, generate a new image PIC based on LABELED_IMAGE that
% contains `objects' that pass PREDICATE. `Objects' are those
% represented by the indices of LABELED_IMAGE. N is the number of
% objects in the returned image PIC.
%
% The function doesn't just allow object that passes PREDICATE, however.
% Instead, it selects the `maximum' element - ie the winner of sequential
% pairwise testing of PREDICATE. OPTIONS determine a REGIONPROPS (a la
% REGIONDATA) stat struct that is passed to PREDICATE. PREDICATE should
% have the following form:
%     BOOL = PREDICATE(PREVIOUS_STATS, NEXT_STATS)
%
% So, for example, if we have the predicate:
%     @(p, n) n.Area > p.Area
% Then, we are testing the `next' area against the `previous' area, and
% we will arrive at an image that has the object of the greatest area.
%
% For options, everything should be in the style of regionprops and
% regiondata, except for the following:
%% n, a positive integer
% Here, instead of just finding the `maximum' object, find n maximal
% objects. When the i'th (i<n) object is displaced by a new stat, then
% the displaced stat is checked against the rest of the `current' list.
% So for the above Area example, an n of 3 would result in the three
% largest objects in the image being selected.
```

```

%% f, a function handle
% Use f as a predicate on each object individually. ie, we can have
% (@(o) o.Area > 200) to ensure that all found objects are atleast 200
% pixels large. There can be any number of these, and all must be
% passed for an object to be accepted.
%% 'close'
% `Close' the objects. ie, make all the found objects be apart of the
% same object. So if n is 3, using the close argument would find the
% three maximum objects, and return a labelled image where those three
% are all the same object (ie, all are '1').
%
% 'near', 'really-near'
% Demand that all objects be `near' one another. The first object in
% the ordering is automatically in. Everything that is either near
% that, or indirectly near it (ie transitively) is in. Near means
% being within a MajorAxis length of an already accepted object.
% Really near means within a MinorAxis length.
%% 'similar'
% Try to find objects that are relatively near one another in size.
% The first object in the ordering is automatically in. Everything
% that is either similar in size to that, or another accepted object,
% is accepted. Here, similar means that the area is within a factor of
% three (ie, either 1/3 or 3x larger).
%% 'fill'
% Fill in the image, as in regionprops filled image.
%

```

```

idx = unique(lbl);
idx = idx(idx ~= 0);

n = 1;
[args, funcs, fill, close, near, reallyNear, similar] = deal({}, {}, 0,
0, 0, 0, 0);
if ~isempty(varargin)
    args = cell(1, numel(varargin));
    argi = 1;
    for i = 1:numel(varargin)
        arg = varargin{i};
        if isnumeric(arg)
            arg = floor(arg);
            if arg < 1
                error('RegionSelect:Arguments', 'Number of objects must be
positive integer.');
            end
            n = arg;
            continue;
        end
        if islambda(arg)
            funcs{end+1} = arg; %#ok<AGROW>
            continue;
        end
        if matchesi(arg, '^close$')
            close = 1;
            continue;
        end
        if matchesi(arg, '^(really-?)?near$')

```

```

        near = 1;
        reallyNear = matchesi(arg, 'really');
        continue;
    end
    if matchesi(arg, '^similar$')
        similar = 1;
        continue;
    end
    if matchesi(arg, '^fill$')
        fill = 1;
        continue;
    end
    args{argi} = arg;
    argi = argi + 1;
end
end

if argi == 1
    args = {};
else
    args = args(1:argi-1);
end

if near
    args{end+1} = 'Centroid';
    args{end+1} = 'MajorAxisLength';
    args{end+1} = 'MinorAxisLength';
end

if similar
    args{end+1} = 'Area';
end

add = @(i) foldl(funcs, 1, @(id, ff) id && ff(i));

pic = zeros(size(lbl), class(lbl));
if isempty(idx), return; end

found = zeros(n, 1);
stats = cell(n, 1);

func = regiondata(lbl, args{:});

for i = idx(:)'
    cIdx = i;
    sts = func(i);

    for j = 1:n
        if ~found(j)
            if add(sts)
                found(j) = cIdx;
                stats{j} = sts;
            end
            break;
        end
        if pred(stats{j}, sts)

```

```

    tmp = stats{j};
    stats{j} = sts;
    sts = tmp;

    tmp = found(j);
    found(j) = cIdx;
    cIdx = tmp;
end
end
end

function compFunc(compPred)
    pass = false(1, numel(found));
    pass(1) = true;
    changed = false;

    % Mark everything that is within range of a found object.
    while 1
        for unpassed = 1:numel(pass)
            % Ignore this if we've already accepted it.
            if pass(unpassed), continue; end

            % Otherwise, we need to look at everything it could be near.
            for parent = 1:numel(pass)
                % Only consider what it could be near if it matters.
                if parent == unpassed, continue; end
                if ~pass(parent), continue; end

                % If we are near, then make note!
                if compPred(unpassed, parent)
                    changed = 1;
                    pass(unpassed) = true;
                    break;
                end
            end
        end
    end

    % If there wasn't any change, then we're done.
    if ~changed, break; end
    changed = false;
end

stats = stats(pass);
found = found(pass);
end

stats = stats(found ~= 0);
found = found(found ~= 0);

function d = centroidDist(a, b)
    [ax, ay] = deal(a.Centroid(1), a.Centroid(2));
    [bx, by] = deal(b.Centroid(1), b.Centroid(2));
    d = sqrt((ax - bx) ^ 2 + (ay - by) ^ 2);
end

if near

```

```

dist = zeros(numel(found), 'double');
for i = 1:numel(found)
    for j = 1:numel(found)
        dist(i, j) = centroidDist(stats{i}, stats{j});
    end
end

field = 'MajorAxis';
if reallyNear, field = 'MinorAxis'; end

compFunc(@(unpassed, parent) dist(unpassed, parent) <=
stats{parent}.(field));
end

if similar
    areas = @(u, p) [stats{u}.Area, stats{p}.Area];
    simPred = @(a) max(a) / min(a) <= 3;
    compFunc(@(unpassed, parent) simPred(areas(unpassed, parent)));
end

logIdx = ismember(lbl, found);
if close
    pic(logIdx) = 1;
else
    pic(logIdx) = lbl(logIdx);
end
% Calculate the filled image, if desired.
if fill
    idx = unique(pic);
    idx = idx(idx ~= 0);
    func = regiondata(pic, 'FilledImage', 'BoundingBox');
    for i = 1:numel(idx)
        st = func(idx(i));
        box = round(st.BoundingBox);
        [x, y, w, h] = deal(box(1), box(2), box(3), box(4));
        zimg = zeros(size(pic), class(pic));
        zimg(y:y+(h-1), x:x+(w-1)) = st.FilledImage;
        logIdx = (zimg ~= 0);
        pic(logIdx) = idx(i);
    end
end

varargout{1} = pic;
if nargout > 1
    varargout{2} = numel(found);
end
end

```

## 7. FUNCTION STRUCTURED SIMILARITY INDEX QUALITY (SSIM) MEASURE

```
function [mssim, ssim_map] = ssim_index(img1, img2, K, window, L)

%=====
%SSIM Index, Version 1.0
%Copyright(c) 2003 Zhou Wang
%All Rights Reserved.
%
%The author is with Howard Hughes Medical Institute, and Laboratory
%for Computational Vision at Center for Neural Science and Courant
%Institute of Mathematical Sciences, New York University.
%
%-----
%Permission to use, copy, or modify this software and its documentation
%for educational and research purposes only and without fee is hereby
%granted, provided that this copyright notice and the original authors'
%names appear on all copies and supporting documentation. This program
%shall not be used, rewritten, or adapted as the basis of a commercial
%software or hardware product without first obtaining permission of the
%authors. The authors make no representations about the suitability of
%this software for any purpose. It is provided "as is" without express
%or implied warranty.
%-----
%
%This is an implementation of the algorithm for calculating the
%Structural SIMilarity (SSIM) index between two images. Please refer
%to the following paper:
%
%Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image
%quality assessment: From error measurement to structural similarity"
%IEEE Transactions on Image Processing, vol. 13, no. 1, Jan. 2004.
%
%Kindly report any suggestions or corrections to zhouwang@ieee.org
%
%-----
%
%Input : (1) img1: the first image being compared
%        (2) img2: the second image being compared
%        (3) K: constants in the SSIM index formula (see the above
%            reference). default value: K = [0.01 0.03]
%        (4) window: local window for statistics (see the above
%            reference). default window is Gaussian given by
%            window = fspecial('gaussian', 11, 1.5);
%        (5) L: dynamic range of the images. default: L = 255
%
%Output: (1) mssim: the mean SSIM index value between 2 images.
%          If one of the images being compared is regarded as
%          perfect quality, then mssim can be considered as the
%          quality measure of the other image.
%          If img1 = img2, then mssim = 1.
%          (2) ssim_map: the SSIM index map of the test image. The map
%          has a smaller size than the input images. The actual size:
%          size(img1) - size(window) + 1.
```

```

%
%Default Usage:
%   Given 2 test images img1 and img2, whose dynamic range is 0-255
%
%   [mssim ssim_map] = ssim_index(img1, img2);
%
%Advanced Usage:
%   User defined parameters. For example
%
%   K = [0.05 0.05];
%   window = ones(8);
%   L = 100;
%   [mssim ssim_map] = ssim_index(img1, img2, K, window, L);
%
%See the results:
%
%   mssim                                %Gives the mssim value
%   imshow(max(0, ssim_map).^4)          %Shows the SSIM index map
%
%=====

if (nargin < 2 || nargin > 5)
    ssim_index = -Inf;
    ssim_map = -Inf;
    return;
end

if (size(img1) ~= size(img2))
    ssim_index = -Inf;
    ssim_map = -Inf;
    return;
end

[M N] = size(img1);

if (nargin == 2)
    if ((M < 11) || (N < 11))
        ssim_index = -Inf;
        ssim_map = -Inf;
        return
    end
    window = fspecial('gaussian', 11, 1.5); %
    K(1) = 0.01;                            % default settings
    K(2) = 0.03;                            %
    L = 255;                                %
end

if (nargin == 3)
    if ((M < 11) || (N < 11))
        ssim_index = -Inf;
        ssim_map = -Inf;
        return
    end
    window = fspecial('gaussian', 11, 1.5);
    L = 255;
    if (length(K) == 2)

```



```

        if (K(1) < 0 || K(2) < 0)
            ssim_index = -Inf;
            ssim_map = -Inf;
            return;
        end
    else
        ssim_index = -Inf;
        ssim_map = -Inf;
        return;
    end
end

if (nargin == 4)
    [H W] = size(window);
    if ((H*W) < 4 || (H > M) || (W > N))
        ssim_index = -Inf;
        ssim_map = -Inf;
        return
    end
    L = 255;
    if (length(K) == 2)
        if (K(1) < 0 || K(2) < 0)
            ssim_index = -Inf;
            ssim_map = -Inf;
            return;
        end
    else
        ssim_index = -Inf;
        ssim_map = -Inf;
        return;
    end
end

if (nargin == 5)
    [H W] = size(window);
    if ((H*W) < 4 || (H > M) || (W > N))
        ssim_index = -Inf;
        ssim_map = -Inf;
        return
    end
    if (length(K) == 2)
        if (K(1) < 0 || K(2) < 0)
            ssim_index = -Inf;
            ssim_map = -Inf;
            return;
        end
    else
        ssim_index = -Inf;
        ssim_map = -Inf;
        return;
    end
end

C1 = (K(1)*L)^2;
C2 = (K(2)*L)^2;
window = window/sum(sum(window));

```

```

img1 = double(img1);
img2 = double(img2);
mul   = filter2(window, img1, 'valid');
mu2   = filter2(window, img2, 'valid');
mul_sq = mul.*mul;
mu2_sq = mu2.*mu2;
mul_mu2 = mul.*mu2;
sigma1_sq = filter2(window, img1.*img1, 'valid') - mul_sq;
sigma2_sq = filter2(window, img2.*img2, 'valid') - mu2_sq;
sigma12 = filter2(window, img1.*img2, 'valid') - mul_mu2;

if (C1 > 0 & C2 > 0)
    ssim_map = ((2*mul_mu2 + C1).*(2*sigma12 + C2))./((mul_sq + mu2_sq +
C1).*(sigma1_sq + sigma2_sq + C2));
else
    numerator1 = 2*mul_mu2 + C1;
    numerator2 = 2*sigma12 + C2;
    denominator1 = mul_sq + mu2_sq + C1;
    denominator2 = sigma1_sq + sigma2_sq + C2;
    ssim_map = ones(size(mul));
    index = (denominator1.*denominator2 > 0);
    ssim_map(index) =
(numerator1(index).*numerator2(index))./(denominator1(index).*denominator2(
index));
    index = (denominator1 ~= 0) & (denominator2 == 0);
    ssim_map(index) = numerator1(index)./denominator1(index);
end

mssim = mean2(ssim_map);

return

```

## 8. FUNCTION FOR FINDING MEAN SQUARED ERROR (MSE) AND PEAK SIGNAL NOISE RATION (PSNR)

```

% Author: Owalla Felix, University of Nairobi 2011
%function for finding MSE and PSNR
% This is a standard Matlab/Simulink function
function p=psnr2(x,xc,g)
x=double(x);
xc=double(xc);
t=(x-xc).^2;
%disp('s');
s=sum(sum(t));

%p=g*g*(size(x,1)*size(x,2))/(s);
n=size(x,1)*size(x,2);

MSE=s/n;

disp('PSNR = ');
p=(g*g)/(s/n);

```

```
p=10*log10(p);
disp(p);
```

## 9. FUNCTION REGION SELECT

```
function varargout = regionselect(lbl, pred, varargin)
% Author Owalla Felix, University of Nairobi 2011
%PIC = REGIONSELECT(LABELED_IMAGE, PREDICATE, OPTIONS)
% [PIC, N] = REGIONSELECT(LABELED_IMAGE, PREDICATE, OPTIONS)
% Using PREDICATE, generate a new image PIC based on LABELED_IMAGE that
% contains `objects' that pass PREDICATE. `Objects' are those
% represented by the indices of LABELED_IMAGE. N is the number of
% objects in the returned image PIC.
%
% The function doesn't just allow object that passes PREDICATE, however.
% Instead, it selects the `maximum' element - ie the winner of sequential
% pairwise testing of PREDICATE. OPTIONS determine a REGIONPROPS (a la
% REGIONDATA) stat struct that is passed to PREDICATE. PREDICATE should
% have the following form:
%     BOOL = PREDICATE(PREVIOUS_STATS, NEXT_STATS)
%
% So, for example, if we have the predicate:
%     @(p, n) n.Area > p.Area
% Then, we are testing the `next' area against the `previous' area, and
% we will arrive at an image that has the object of the greatest area.
%
% For options, everything should be in the style of regionprops and
% regiondata, except for the following:
%
% n, a positive integer
% Here, instead of just finding the `maximum' object, find n maximal
% objects. When the i'th (i<n) object is displaced by a new stat, then
% the displaced stat is checked against the rest of the `current' list.
% So for the above Area example, an n of 3 would result in the three
% largest objects in the image being selected.
%
% f, a function handle
% Use f as a predicate on each object individually. ie, we can have
% @(o) o.Area > 200) to ensure that all found objects are atleast 200
% pixels large. There can be any number of these, and all must be
% passed for an object to be accepted.
%
% `close'
% `Close' the objects. ie, make all the found objects be apart of the
% same object. So if n is 3, using the close argument would find the
% three maximum objects, and return a labelled image where those three
% are all the same object (ie, all are '1').
%
% `near', `really-near'
% Demand that all objects be `near' one another. The first object in
% the ordering is automatically in. Everything that is either near
% that, or indirectly near it (ie transitively) is in. Near means
% being within a MajorAxis length of an already accepted object.
% Really near means within a MinorAxis length.
```

```

%
% 'similar'
% Try to find objects that are relatively near one another in size.
% The first object in the ordering is automatically in. Everything
% that is either similar in size to that, or another accepted object,
% is accepted. Here, similar means that the area is within a factor of
% three (ie, either 1/3 or 3x larger).
%
% 'fill'
% Fill in the image, as in regionprops filled image.
%

idx = unique(lbl);
idx = idx(idx ~= 0);

n = 1;
[args, funcs, fill, close, near, reallyNear, similar] = deal({}, {}, 0,
0, 0, 0, 0);
if ~isempty(varargin)
    args = cell(1, numel(varargin));
    argi = 1;
    for i = 1:numel(varargin)
        arg = varargin{i};
        if isnumeric(arg)
            arg = floor(arg);
            if arg < 1
                error('RegionSelect:Arguments', 'Number of objects must be
positive integer.');
            end
            n = arg;
            continue;
        end
        if islambda(arg)
            funcs{end+1} = arg; %#ok<AGROW>
            continue;
        end
        if matchesi(arg, '^close$')
            close = 1;
            continue;
        end
        if matchesi(arg, '^(really-?)?near$')
            near = 1;
            reallyNear = matchesi(arg, 'really');
            continue;
        end
        if matchesi(arg, '^similar$')
            similar = 1;
            continue;
        end
        if matchesi(arg, '^fill$')
            fill = 1;
            continue;
        end
        args{argi} = arg;
        argi = argi + 1;
    end
end

```

```

    end
end

if argi == 1
    args = {};
else
    args = args(1:argi-1);
end

if near
    args{end+1} = 'Centroid';
    args{end+1} = 'MajorAxisLength';
    args{end+1} = 'MinorAxisLength';
end

if similar
    args{end+1} = 'Area';
end

add = @(i) foldl(funcs, 1, @(id, ff) id && ff(i));

pic = zeros(size(lbl), class(lbl));
if isempty(idx), return; end

found = zeros(n, 1);
stats = cell(n, 1);

func = regiondata(lbl, args{:});

for i = idx(:)'
    cIdx = i;
    sts = func(i);

    for j = 1:n
        if ~found(j)
            if add(sts)
                found(j) = cIdx;
                stats{j} = sts;
            end
            break;
        end
        if pred(stats{j}, sts)
            tmp = stats{j};
            stats{j} = sts;
            sts = tmp;

            tmp = found(j);
            found(j) = cIdx;
            cIdx = tmp;
        end
    end
end
end
end

```

```

function compFunc(compPred)
    pass = false(1, numel(found));
    pass(1) = true;
    changed = false;

    % Mark everything that is within range of a found object.
    while 1
        for unpassed = 1:numel(pass)
            % Ignore this if we've already accepted it.
            if pass(unpassed), continue; end

            % Otherwise, we need to look at everything it could be near.
            for parent = 1:numel(pass)
                % Only consider what it could be near if it matters.
                if parent == unpassed, continue; end
                if ~pass(parent), continue; end

                % If we are near, then make note!
                if compPred(unpassed, parent)
                    changed = 1;
                    pass(unpassed) = true;
                    break;
                end
            end
        end
        end

        % If there wasn't any change, then we're done.
        if ~changed, break; end
        changed = false;
    end

    stats = stats(pass);
    found = found(pass);
end

stats = stats(found ~= 0);
found = found(found ~= 0);

function d = centroidDist(a, b)
    [ax, ay] = deal(a.Centroid(1), a.Centroid(2));
    [bx, by] = deal(b.Centroid(1), b.Centroid(2));
    d = sqrt((ax - bx) ^ 2 + (ay - by) ^ 2);
end

if near
    dist = zeros(numel(found), 'double');
    for i = 1:numel(found)
        for j = 1:numel(found)
            dist(i, j) = centroidDist(stats{i}, stats{j});
        end
    end

    field = 'MajorAxis';
    if reallyNear, field = 'MinorAxis'; end
end

```

```

    compFunc(@(unpassed, parent) dist(unpassed, parent) <=
stats{parent}.(field));
end

if similar
    areas = @(u, p) [stats{u}.Area, stats{p}.Area];
    simPred = @(a) max(a) / min(a) <= 3;
    compFunc(@(unpassed, parent) simPred(areas(unpassed, parent)));
end

logIdx = ismember(lbl, found);
if close
    pic(logIdx) = 1;
else
    pic(logIdx) = lbl(logIdx);
end
% Calculate the filled image, if desired.
if fill
    idx = unique(pic);
    idx = idx(idx ~= 0);
    func = regiondata(pic, 'FilledImage', 'BoundingBox');
    for i = 1:numel(idx)
        st = func(idx(i));
        box = round(st.BoundingBox);
        [x, y, w, h] = deal(box(1), box(2), box(3), box(4));
        zimg = zeros(size(pic), class(pic));
        zimg(y:y+(h-1), x:x+(w-1)) = st.FilledImage;
        logIdx = (zimg ~= 0);
        pic(logIdx) = idx(i);
    end
end

varargout{1} = pic;
if nargout > 1
    varargout{2} = numel(found);
end
end

```

## 10. FUNCTION STRUCTURED SIMILARITY INDEX QUALITY (SSIM) MEASURE

```
function [mssim, ssim_map] = ssim_index(img1, img2, K, window, L)
```

```

%=====
%SSIM Index, Version 1.0
%Copyright (c) 2003 Zhou Wang
%All Rights Reserved.
%
%The author is with Howard Hughes Medical Institute, and Laboratory
%for Computational Vision at Center for Neural Science and Courant
%Institute of Mathematical Sciences, New York University.
%
%-----

```

%Permission to use, copy, or modify this software and its documentation  
%for educational and research purposes only and without fee is hereby  
%granted, provided that this copyright notice and the original authors'  
%names appear on all copies and supporting documentation. This program  
%shall not be used, rewritten, or adapted as the basis of a commercial  
%software or hardware product without first obtaining permission of the  
%authors. The authors make no representations about the suitability of  
%this software for any purpose. It is provided "as is" without express  
%or implied warranty.

-----  
%

%This is an implementation of the algorithm for calculating the  
%Structural SIMilarity (SSIM) index between two images. Please refer  
%to the following paper:

%Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image  
%quality assessment: From error measurement to structural similarity"  
%IEEE Transactions on Image Processing, vol. 13, no. 1, Jan. 2004.

%Kindly report any suggestions or corrections to zhouwang@ieee.org

-----  
%

%Input : (1) img1: the first image being compared  
% (2) img2: the second image being compared  
% (3) K: constants in the SSIM index formula (see the above  
% reference). default value: K = [0.01 0.03]  
% (4) window: local window for statistics (see the above  
% reference). default window is Gaussian given by  
% window = fspecial('gaussian', 11, 1.5);  
% (5) L: dynamic range of the images. default: L = 255  
%  
%Output: (1) mssim: the mean SSIM index value between 2 images.  
% If one of the images being compared is regarded as  
% perfect quality, then mssim can be considered as the  
% quality measure of the other image.  
% If img1 = img2, then mssim = 1.  
% (2) ssim\_map: the SSIM index map of the test image. The map  
% has a smaller size than the input images. The actual size:  
% size(img1) - size(window) + 1.

%Default Usage:  
% Given 2 test images img1 and img2, whose dynamic range is 0-255  
%  
% [mssim ssim\_map] = ssim\_index(img1, img2);

%Advanced Usage:  
% User defined parameters. For example  
%  
% K = [0.05 0.05];  
% window = ones(8);  
% L = 100;  
% [mssim ssim\_map] = ssim\_index(img1, img2, K, window, L);

%See the results:  
%



```

% mssim %Gives the mssim value
% imshow(max(0, ssim_map).^4) %Shows the SSIM index map
%
%=====

if (nargin < 2 || nargin > 5)
    ssim_index = -Inf;
    ssim_map = -Inf;
    return;
end

if (size(img1) ~= size(img2))
    ssim_index = -Inf;
    ssim_map = -Inf;
    return;
end

[M N] = size(img1);

if (nargin == 2)
    if ((M < 11) || (N < 11))
        ssim_index = -Inf;
        ssim_map = -Inf;
        return
    end
    window = fspecial('gaussian', 11, 1.5); %
    K(1) = 0.01; % default settings
    K(2) = 0.03; %
    L = 255; %
end

if (nargin == 3)
    if ((M < 11) || (N < 11))
        ssim_index = -Inf;
        ssim_map = -Inf;
        return
    end
    window = fspecial('gaussian', 11, 1.5);
    L = 255;
    if (length(K) == 2)
        if (K(1) < 0 || K(2) < 0)
            ssim_index = -Inf;
            ssim_map = -Inf;
            return;
        end
    else
        ssim_index = -Inf;
        ssim_map = -Inf;
        return;
    end
end

if (nargin == 4)

```

```

[H W] = size(window);
if ((H*W) < 4 || (H > M) || (W > N))
    ssim_index = -Inf;
    ssim_map = -Inf;
    return
end
L = 255;
if (length(K) == 2)
    if (K(1) < 0 || K(2) < 0)
        ssim_index = -Inf;
        ssim_map = -Inf;
        return;
    end
else
    ssim_index = -Inf;
    ssim_map = -Inf;
    return;
end
end

if (nargin == 5)
[H W] = size(window);
if ((H*W) < 4 || (H > M) || (W > N))
    ssim_index = -Inf;
    ssim_map = -Inf;
    return
end
if (length(K) == 2)
    if (K(1) < 0 || K(2) < 0)
        ssim_index = -Inf;
        ssim_map = -Inf;
        return;
    end
else
    ssim_index = -Inf;
    ssim_map = -Inf;
    return;
end
end

C1 = (K(1)*L)^2;
C2 = (K(2)*L)^2;
window = window/sum(sum(window));
img1 = double(img1);
img2 = double(img2);
mu1 = filter2(window, img1, 'valid');
mu2 = filter2(window, img2, 'valid');
mu1_sq = mu1.*mu1;
mu2_sq = mu2.*mu2;
mu1_mu2 = mu1.*mu2;
sigma1_sq = filter2(window, img1.*img1, 'valid') - mu1_sq;
sigma2_sq = filter2(window, img2.*img2, 'valid') - mu2_sq;
sigma12 = filter2(window, img1.*img2, 'valid') - mu1_mu2;

if (C1 > 0 & C2 > 0)

```

```

    ssim_map = ((2*mu1_mu2 + C1).*(2*sigma12 + C2))./((mu1_sq + mu2_sq +
C1).*(sigma1_sq + sigma2_sq + C2));
else
    numerator1 = 2*mu1_mu2 + C1;
    numerator2 = 2*sigma12 + C2;
    denominator1 = mu1_sq + mu2_sq + C1;
    denominator2 = sigma1_sq + sigma2_sq + C2;
    ssim_map = ones(size(mu1));
    index = (denominator1.*denominator2 > 0);
    ssim_map(index) =
(numerator1(index).*numerator2(index))./(denominator1(index).*denominator2(
index));
    index = (denominator1 ~= 0) & (denominator2 == 0);
    ssim_map(index) = numerator1(index)./denominator1(index);
end

mssim = mean2(ssim_map);

return

```

## APPENDIX C: PUBLICATION

This appendix consists of work that was accepted and presented at the IEEE Africon 2017 Conference which was held from 18th to 20th September 2013 in Cape Town, Republic of South Africa. The paper has been indexed and published in the digital library of the IEEE Xplore under the conference proceedings. Figure C.1 below shows an image taken from the homepage of the IEEE Africon 2017 website. <http://africon2017.org>



*Fig. B.1 Homepage of the IEEE Africon 2017 (<http://africon2017.org>)*

# A Robust Video Watermarking Approach based on a hybrid SVD/DWT Technique

Vincent Adul and Elijah Mwangi  
School of Engineering  
University of Nairobi  
Nairobi, Kenya  
vadul@ncs.go.ke

*Abstract*— A comparative analysis to determine the most effective video watermarking algorithm between SVD/DWT hybrid and Singular Value Decomposition (SVD) is reported in this paper. Blind video watermarking schemes are simulated and attempts made to recover the watermark after some signal processing attacks such as median filtering and Histogram equalization. The quality of the extracted watermark was then measured using the SSIM index. From the computer simulation results using a diverse set of standard video clips the SVD/DWT hybrid performed better than the reference. An average value of the SSIM index of 0.98 was obtained. The SVD transform values varied from 0.57 to 0.78 for histogram equalization attacks and 0.83 to 0.9 for median filtering attacks. The results reveal the superiority of the SVD/DWT hybrid technique over SVD for digital rights enforcement.

**Keywords**—Video processing, watermarking, DWT applications, SVD applications, Digital rights enforcement

## **Introduction**

The advent of high-speed computer networks, the Internet and the World Wide Web have revolutionized the way in which digital data is distributed. The quality of digital video, audio and image media offer several distinct advantages over their analogue counterparts. Editing is easy because one can access the exact discrete locations that need to be changed. Copying is simple with no loss of fidelity and the results are identical to the original. However, this has threatened the enforcement of Intellectual Property Rights due to the possibility of unlimited copying. The widespread and easy access to multimedia contents and the possibility of making unlimited copies without loss, created the need for digital rights management. Some management methods like encryption prevents unauthorized access by keeping communication secure between the parties involved. This is done by

'scrambling' the information sent from one entity to another into a lengthy code making it unreadable for anybody else attempting to access it. When the data is encrypted, the sender and the receiver are the only entities capable of decrypting the scrambled information back to a readable condition. This is achieved via 'keys', which grant only the users involved access to modify the data to make it unreadable and then readable again. Although encryption does provide overall protection, the data can easily be decrypted and distributed freely or manipulated over the Internet. It is for this reason that digital watermarking has emerged as a solution. It can serve the purpose of authentication of the genuine owner of the content. Digital watermarking technique embeds a watermark within the original content, while ensuring considerable fidelity of information about the copyright status of the work to be protected. A large number of digital watermarking schemes have been proposed using diverse transforms and data hiding strategies [1].

## **Basic Principles of Watermarking**

Digital watermarking is a technique of hiding some data [2] that are called logo or watermark in the original image, video, or audio. It effectively handles the protection of digital data by providing a means of verification of the authenticity of the originator [3]. Digital Watermarking schemes can be classified into either visible or invisible [4]. In visible watermarking the logo is embedded in the spatial domain and clearly visible under the normal viewing condition, but the watermark cannot be taken away.

In invisible watermarking the logo is not seen under the normal viewing condition. The quality of the content would be severely degraded if an attempt is made to remove it. The watermark can be embedded within either the spatial domain or transform domain or both. In the spatial domain, an algorithm operates directly on the pixel values of the video content while

in the transform domain, the transfer of pixel values into another domain is carried out. The various transforms that are useful for this purpose are the Discrete Cosine Transform (DCT), Singular Value Decomposition (SVD) and Singular Value Decomposition (SVD)/Discrete Wavelet Transform (DWT) hybrid.

Lai and Tsai [5] proposed the use of Discrete Wavelet Transform and Singular Value Decomposition to improve robustness to signal processing and geometric attacks. This proposal exhibits improved robustness to numerous attacks while at the same time increasing computational complexity of the embedding and extraction process. In this paper, invisible watermarking in which an image is embedded within green pixel values in the digital video is employed. The three colour components with RGB a wider pixel values which are able to guard against distortions. This was a reasonable precaution, since there would be less flexibility within the Red and Blue pixel frames.

The paper is arranged as follows. Section II gives a brief introduction to DCT compression techniques for images and or videos, Singular Value Decomposition (SVD) and an overview of application of the DWT in watermarking. The proposed watermarking algorithms are presented in Section III. The computer simulation results are presented and discussed in section IV. Finally, concluding remarks and suggestion for future work are given in section V.

#### *Metrics for Visual Quality/Imperceptibility*

It is important at the outset to have an effective algorithm for measurement of the quality of the output message which is the extracted watermark in comparison to the embedded watermark in order to establish the robustness of the watermarking scheme. The algorithm should relate the difference between two images (the original and the distorted image). Computing the quality of the still images that arise out of splitting video into frames can be a complex process considering that the human's visual systems opinion are affected by multifaceted psychosomatic schemes. Many schemes have been proposed for image quality metrics though perfection is yet to be realized. Some of the quality measurements techniques that have been used include Mean Squared Error (MSE), Peak Signal to Noise Ratio (PSNR), Structural Similarity Index Measurement (SSIM) and several others [6]. The metric that was preferred for the experimental

environment in this paper was the SSIM given the deficiencies of the MSE/PSNR and inability to provide a basis for reasonable signal fidelity measure as given in Wang and Bovik [7]. However, PSNR values have also been retained for purposes of completeness for historical precedence, and do not affect the final human visual system interpretation as reflected by the SSIM values.

## **TRANSFORM DOMAIN WATERMARKING TECHNIQUES**

### **Discrete Cosine Transform**

Discrete Cosine Transform (DCT) is the most commonly used transform domain watermarking technique due to its energy compaction properties and ease of computation [8]. Lin and Chin [9] proposed embedding the watermark in the mid-frequency DCT coefficients of the cover content in order to attain sufficient trade-off between robustness to signal processing attacks and the perceptual quality of the watermarked output. Shoemaker [10] proposed a DCT domain technique that relies on JPEG quantisation, where the coefficients that have equal levels are compared and exchanged in such a way as to signify if a bit has been embedded in them or not. DCT techniques are often used to watermark compressed video streams. The DCT coefficients in video streams can be modified without having to first un-compress the video or compress it again after watermarking.

### **C. SVD Watermarking principles**

This is an eigenvalue technique that can be used in image compression techniques, but can also be applied to watermarking [11]. The SVD is performed, after which the singular values are usually modified to embed the watermark. A pseudo-inverse SVD is then applied to obtain the original content. The SVD can be used on its own for watermarking, but is also often used in hybrid techniques such as [12] which combine the SVD and the discrete cosine transform. The SVD is relatively computationally complex, but by applying it in hybrid techniques it may not be necessary to perform an SVD on the entire image, thus lowering the computational complexity. A matrix  $A$  of size  $M \times N$  and of rank  $R$  can be decomposed as:

$M \times N$  matrix  $A$ , of rank  $R$ , can be expressed as the product,

$$A = UDV^T \quad (1)$$

U and V are both orthogonal and D is a diagonal matrix, thus:

U and V are both orthogonal and D is a diagonal matrix, thus:

$$D = \begin{pmatrix} \sigma_1 & 0 & 0 & 0 \\ 0 & \sigma_2 & 0 & 0 \\ 0 & 0 & \sigma_3 & 0 \\ 0 & 0 & 0 & \sigma_4 \end{pmatrix}$$

Energy in this matrix is such that,  $\sigma_1 > \sigma_2 > \sigma_3 > \sigma_4 > 0$

Where  $A = USV^T$ , and  $S = D$ , U is  $M \times R$  term matrix, S is  $R \times R$  diagonal matrix, V is  $R \times N$  document matrix and if A is  $N \times N$  matrix,  $R = N$ , we have  $AU = US$  (2)

#### Embedding one Bit;

To carry out this process, the following procedure is followed:

- i) Let it be assumed that the host image is an  $M \times N$  matrix;
- ii) Perform the SVD to get S matrix;
- iii) Embed one bit in the S matrix according to the equation;

$$s_i' = s_i + kb \quad (3)$$

where  $\{s_i\}$ : original coefficients

$\{s_i'\}$ : watermarked coefficients

$\{b\}$ : the bit to be embedded which is 0 or 1

k : watermark strength, adjusted by the just-noticeable-difference (JND) standard

$$\text{After embedding: } A' = US'VT \quad (4)$$

where  $S'$  is the watermarked singular matrix and  $A'$  is the corresponding watermarked image.

Detection; If it is assumed that the watermarked image  $A'$  is obtained, then the Eigen-decomposition or SVD is performed to get the matrix  $S'$ . The output  $S'$  is then compared to S to obtain the watermark.

#### DWT Watermarking principles

Discrete Wavelet Transform (DWT) watermarking [13] consists of a multi-scale decomposition of an image into low frequency approximation, with horizontal, vertical, and diagonal details. Two channel filter banks are used in obtaining these different levels of resolutions defined by the equations;

$$\psi H(x,y) = \psi(x)\varphi(y) \quad (5a)$$

$$\psi V(x,y) = \varphi(x)\psi(y) \quad (5b)$$

$$\psi D(x,y) = \psi(x)\varphi(y) \quad (5c)$$

Where

$$\psi(x) = \sum h\psi(n) \sqrt{2}\varphi(2x-n)$$

$$\varphi(x) = \sum h\varphi(n) \sqrt{2}\varphi(2x-n)$$

Where  $h\psi$  and  $h\varphi$  represent the scaling and the wavelet vectors respectively

It is known that the DWT domain models the human visual system more closely than the Fourier based transform techniques however embedding the watermark in the DWT domain is more computationally complex. It is also known that if encoders of the same level of technology were employed in both the Fourier based transform techniques and in the DWT domain, the differences in their performance would be insignificant while the computational costs would increase [14]. One of the important mathematical properties of SVD is that slight variations of singular values do not affect the visual perception of the host image, which motivates the watermark embedding procedure to achieve good transparency and robustness

#### SIMULATION PROCESSES

Video watermarking was simulated in the Singular Value Decomposition (SVD) and hybrid SVD/DWT domains following the process described below. The Video sequences were first decomposed into frames. The frames, which are in RGB colour space, were then decomposed into the component colour images and the green component was chosen as the cover image.

#### SVD Watermarking technique

The Singular Value Decomposition (SVD) of the image is obtained and the Singular matrix is chosen as the embedding medium. The message is then embedded into the singular matrix. Inverse SVD is then performed to obtain the watermarked green component. The colour components are then concatenated to obtain the watermark frame. The frames are then combined to obtain the watermarked video.

The procedure of embedding the watermark is as follows:

- i) Extract the frames from the video and split the frame image into its colour component images.
- ii) Then SVD transform is performed on the component image chosen for embedding.
- iii) The message is then embedded in the singular values matrix as follows:

$$Sw(i,j) = S(i,j) + kW(i,j) \quad (6)$$

Where  $Sw(i,j)$  is the watermarked singular value coefficient,  $S(i,j)$  is the original singular value

coefficient,  $k$  is the embedding factor and  $W(i,j)$  is the message.

iv) Perform the inverse SVD of the watermarked coefficients and then concatenate the watermarked colour component image to the other colour component images to obtain the watermarked frame.

v) The frames are then inserted to the video to obtain the watermarked video.

The procedure for extracting the watermark is as follows:

i) Extract the frames from the video, split the frame into colour component images

ii) Perform SVD on the image.

iii) Extract the watermark as follows:

$$W(i,j) = S_w(i,j) - S(i,j)/k \quad (7)$$

iv) Reorganize the coefficients to obtain the extracted watermark.

SVD/DWT Domain

The procedure of embedding the watermark is as follows:

i) Extract the frames from the video and split the frame image into its colour component images.

ii) Perform the DWT on the cover image.

iii) The diagonal detail coefficients are chosen for embedding the watermark. Then SVD is performed on the Diagonal detail coefficients.

iv) The message is then embedded in the Singular values matrix as in equation 6.

v) Perform the inverse SVD and inverse DWT of the watermarked coefficients and then concatenate the watermarked colour component image to the other colour component images to obtain the watermarked frame.

vi) The frames are then inserted to the video to obtain the watermarked video.

The procedure for extracting the watermark is as follows:

i) Extract the frames from the video, split the frame into colour component images

ii) Perform DWT and SVD on the image.

iii) Extract the watermark as in equation 5:

iv) Reorganize the coefficients to obtain the extracted watermark.

## RESULTS AND DISCUSSIONS

The above techniques were simulated on MATLAB to test the visual quality degradation of the video streams and robustness to some signal processing attacks. The transform techniques were tested on various test

videos including Container, Carphone, Foreman and Bus whose resolutions were CIF and the watermark used had a size of  $128 \times 128$  pixels.

SVD Watermarking results

During the SVD watermarking process the original frame sized of  $256 \times 256$  had a  $128 \times 128$  pixels message embedded within it, after splitting the video into frames. The green pixels were chosen for the exercise, given the well-known fact that it has the largest spread of picture elements over red and blue areas. Fig 4.1 shows the original frame and the watermark (message), where the IEEE logo was used. The output of the process is shown in Fig.4.2 and was established to have a PSNR of 50.93dB and SSIM of 0.99 which was good result considering that the SSIM value is almost 1.



Fig 4.1 :The original frame and embedded message using the SVD technique.



Fig 4.2: The watermarked frame at a PSNR of 50.93dB and the extracted watermark at SSIM of 0.99

### Impact of Signal processing attacks

The results of histogram equalization and median filtering attacks are shown in Fig 4.3 and 4.4 respectively. They indicate that histogram equalization attack, degraded the SSIM value from the 0.99 above to 0.57 whereas the Median filtering degraded the value to 0.86.





Fig 4.3: The watermarked frame and the extracted watermark after histogram attack at an SSIM of 0.57



Fig 4.4: The watermarked frame and the extracted watermark after Median filtering attack at an SSIM of 0.86

**SVD/DWT hybrid**

The hybrid SVD and the DWT Techniques hybrid Watermarking was carried out as per the procedure already described above and the results showing the original frame, message, watermarked and extracted messages obtained are shown in Fig 4.6 and 4.7 respectively where the latter had a PSNR was 36dB and the SSIM of 0.98



Fig 4.6 :The original frame and embedded message using the SVD/DWT hybrid



Fig 4.7: SVD/DWT hybrid watermarked frame with PSNR of 36.6dB at SSIM of 0.98

**Signal processing attacks**

Like in the case of SVD the SVD/DWT hybrid was subjected to Histogram Equalization and Median filtering attacks and the results shown versions. Fig 4.4 shows the result of the Histogram equalization attack, did not degrade SSIM value which remained at 0.99 as shown in Fig 4.6.



Fig 4.5: The original frame and embedded message using the SVD/DWT hybrid technique.

The original frame and message



The Watermarked frame and the extracted message PSNR with 37dB and SSIM of 0.98

**Summary of results**

The table below shows the summary of performance of watermarking techniques to various attacks for both the SVD and the SVD/DWT hybrid algorithms.

Table 1: Summary of Simulation results

Nature of Attack	VIDEO							
	Carphone		Bus		Container		Foreman	
	SV D	SV D/WT	SV D	SV D/WT	SV D	SV D/WT	SV D	SVD/DWT
Hist. Equal.	0.68	0.99	0.57	0.99	0.78	0.98	0.61	0.98
Median Filter.	0.90	0.99	0.86	0.97	0.84	0.99	0.83	0.97

The results have clearly shown that although the Singular Value Decomposition (SVD) is a simple Eigenvalue technique, its resilience to the various attacks is weak.

This was considering the return SSIM values as low as 0.57 and only 0.9 which was obtained while it was subjected to Median filtering attack. These results indicate that SVD may not be the best technique to provide sufficient protection for watermarked video content transmitted through the Internet. It is

instructive that the hybrid between SVD and DWT, gives very positive returns and this applied across the board irrespective of the attack and the type of video clip used. The SVD/DWT hybrid transform domain provides the best protection for video transiting through the Internet space. It is important to note that videos are not subject to geometric attacks such as rotation, scaling and translation among others

### CONCLUSIONS

This investigation concludes that algorithm using SVD/DWT hybrid transform domain has demonstrated better resilience from all the simulated attacks and hence provide the best protection of videos transiting through the Internet space. In this paper SSIM and PSNR were used as metrics to determine the visual quality of the watermarked video frames, and the criteria for robustness of the algorithm. As an extension to further investigations, other image quality metrics such as the Universal Image Quality Index (UIQI) can also be employed. The work will also be extended to include scaling and cropping attacks as well as the use of other wavelet transforms.

### References

P,S Sethuraman, R. Srinivasan. Survey of Digital Video Watermarking Techniques and Its Applications. *Engineering Science*. Vol. 1, No. 1, 2016, pp. 22-27. doi: 10.11648/j.es.20160101.14

R. Bala 'A Brief Survey on Robust Video Watermarking Techniques' *The International Journal Of Engineering And Science (IJES)* Volume 4 Issue 2 Pages PP.41-45 2015

P.D. Sonawane<sup>1</sup> , S. S. Mane , R.N. Nazirkar , P.P. Barhalikar , A Verma 'A Survey on Efficient Video Watermarking and Image Data Encryption Technique' Vol. 4, Issue 10, October 2016- Available online at: [www.ijarcsse.com](http://www.ijarcsse.com)

R. Rawat, N.Kaushik, S. Tiwari 'Digital Watermarking Techniques' , *International Journal of Advanced Research in Computer and Communication Engineering* Vol. 5, Issue 4, April 2016

H.H. Tsai, Y.J. Jhuang, Y.S. Lai, "An SVD-based image watermarking in wavelet domain using SVR and PSO," *Applied Soft Computer*, Vol.12, No.8, pp.2442-2453, 2012.

Y. A. Y. Al-Najjar, D. C. Soong 'Comparison of Image Quality Assessment: PSNR, HVS, SSIM,

UIQI', *International Journal of Scientific & Engineering Research*, Volume 3, Issue 8, August-2012 1 ISSN 2229-5518

Z. Wang, A. C. Bovik-'Mean Squared Error: Love It or Leave It? [A new look at signal fidelity measures]' -*Digital Object Identifier* 10.1109/MSP.2008.930649 *IEEE Signal Processing Magazine* 98 January 2009

N. Alam 'A Robust Video Watermarking Technique using DWT, DCT, and FFT' Volume 6, Issue 6, June 2016-Available online at: [www.ijarcsse.com](http://www.ijarcsse.com)

S.D. Lin and C.-F. Chen 'A robust DCT-based watermarking for copyright protection' *IEEE Transactions on Consumer Electronics* ( Volume: 46, Issue: 3, Aug 2000 )

C. Shoemaker, "Hidden Bits: A Survey of Techniques for Digital Watermarking," *Independent Study*, Spring 2002.

S.F. Abdulla and S. Anjaneyulu ;A Hybrid DWT-SVD Method for Digital Video Watermarking Using Random Frame Selection- *Research Inveny: International Journal of Engineering And Science* Vol.6, Issue 6 (July 2016), PP -43-48 Issn (e): 2278-4721, Issn (p):2319-6483, [www.researchinveny.com](http://www.researchinveny.com)

C. W. H. Fung ; W. Godoy Jr. 'A New Approach of DWT-SVD Video Watermarking'- 2011 Third International Conference on Computational Intelligence, Modelling & Simulation, added to *IEEE Xplore*: 15 November 2011

S. Banyal, S. Sharma 'Digital Video Watermarking Using DWT and SVD Techniques'- *International Journal of Advanced Research in Computer and Communication Engineering* ISO 3297:2007 Certified Vol. 5, Issue 10, October 2016.

Namit T and Sharmila, " Digital Watermarking Applications, Parameter Measurements and Techniques"- *IJCSNS International Journal of Computer Science and Network Security*, VOL.17 No.3, March 2017

**Electronic ISSN: 2153-0033**

**IEEE CODE: PID4793757**

A ROBUST VIDEO  
WATERMARKING APPROACH  
BASED ON SINGULAR VALUE  
DECOMPOSITION AND  
WAVELET TRANSFORM by Adul  
Vincent Otieno

by Adul Vincent Otieno

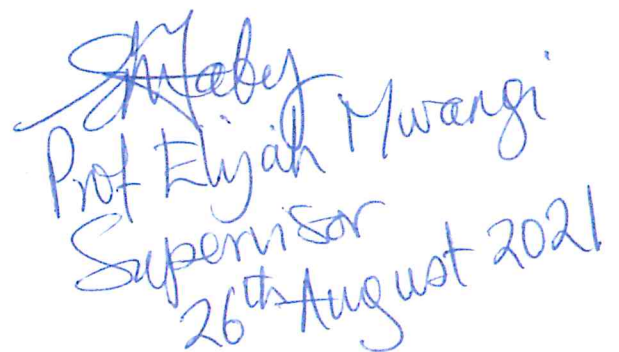
Submission date: 26-Aug-2021 06:05AM (UTC+0300)

Submission ID: 1636067055

File name: Vincent\_Adul-MSc\_Thesis\_Final\_3274.pdf (1.94M)

Word count: 15365

Character count: 80504

  
Prof Elijah Mwangi  
Supervisor  
26th August 2021

  
Dean  
27.08.21

# A ROBUST VIDEO WATERMARKING APPROACH BASED ON SINGULAR VALUE DECOMPOSITION AND WAVELET TRANSFORM by Adul Vincent Otieno

ORIGINALITY REPORT

10%

SIMILARITY INDEX

7%

INTERNET SOURCES

8%

PUBLICATIONS

3%

STUDENT PAPERS

*Prof. Elijah Mwangi*  
*Supervisor* 26<sup>th</sup> August 2021

PRIMARY SOURCES

- 1 Osama S. Faragallah. "Efficient video watermarking based on singular value decomposition in the discrete wavelet transform domain", AEU - International Journal of Electronics and Communications, 2013  
Publication 2%
- 2 Vincent Adul, Elijah Mwangi. "A robust video watermarking approach based on a hybrid SVD/DWT technique", 2017 IEEE AFRICON, 2017  
Publication 1%
- 3 [www.eurojournals.com](http://www.eurojournals.com)  
Internet Source 1%
- 4 [link.springer.com](http://link.springer.com)  
Internet Source <1%
- 5 Submitted to University of Sunderland  
Student Paper <1%