DEVELOPMENT OF A VIDEO FRAME ENHANCEMENT TECHNIQUE BASED ON PIXEL INTENSITY AND HISTOGRAM DISTRIBUTION FOR IMPROVED COMPRESSION

BY

Abdulkareem Hassan Abubakar

(PhD/ENG/1456/2010-2011)

A PhD THESIS SUBMITTED TO THE SCHOOL OF POSGRADUATE STUDIES AHMADU BELLO UNIVERSITY, ZARIA IN PARTIAL FULFILLMENT OF THE REQUIREMENTS FOR THE AWARD OF DOCTOR OF PHILOSOPHY (PhD) DEGREE IN TELECOMMUNICATION ENGINEERING.

July, 2017

DECLARATION

I,Hassan AbubakarABDULKAEEM, declare that the work in this Thesis titled **"Development of a Video Frame Enhancement Technique Based on Pixel Intensity and Histogram Distribution for Improved Compression"** was carried out by me in the Department of Electrical and Computer Engineering. The information derived from the literature has been duly acknowledged in the text and list of references provided. No part of thesis was previously presented for another degree or diploma at this or any other institution.

Hassan Abubakar ABDULKAREEM _____ _____

(Student) Signature Date

CERTIFICATION

This thesis titled "**DEVELOPMENT OF A VIDEO FRAME ENHANCEMENT TECHNIQUE BASED ON PIXEL INTENSITY AND HISTOGRAM DISTRIBUTION FOR IMPROVED COMPRESSION**" by Hassan Abubakar ABDULKAREEM meets the regulations governing the award of the degree of Doctor of Philosophy (PhD) in Telecommunication Engineering of Ahmadu Bello University, and is approved for its contribution to knowledge and literary presentation.

Chairman, Supervisory Committee _____ _____
(Prof. B. G. Bajoga) Signature Date

Member, Supervisory Committee
(Dr. S. M. Sani) _____ _____
Signature Date

Member, Supervisory Committee
(Dr. Y. Jibril) _____
Signature Date

Member, Supervisory Committee
(Dr. A. M. S.Tekanyi) _____ _____
Signature Date

Head of Department
(Dr. Y. Jibril) _____ _____
Signature Date

Dean, School of Postgraduate Studies
Prof. S. Z. Abubakar _____ _____
Signature Date

DEDICATION

This work is hereby dedicated to Almighty Allah, the Lord of the worlds, for its successful completion, and to my late parents whose dream in life was to ensure I really became useful to humanity generally by acquiring quality university education. May Allah SubhanahuWata'la grant them eternal rest. It is equally dedicated to my immediate family members, particularly my beloved wife and children for their patience while carrying out this research work. Alhamdulillah!

ACKNOWLEDGEMENTS

All praise is due to Allah, the most beneficent, the most merciful. His peace and endless blessings be upon the seal of the prophets (Prophet Muhammad - SAW) that was sent as a mercy for all creatures.

Iacknowledge with thanks and appreciation the tremendous contributions of Prof. B.G.Bajoga to this research work. I really appreciate his fatherly advice, motivation, guidance and useful suggestions to ensure a quality work was produced. I indeed remain grateful to him as a unique man of integrity, respect, and inspiration to all those who work closely with him.

The tremendous contribution of Dr. S.M. Sanito this research work must be acknowledged as he gave the work the right focus it deserved through corrections and other technical advice.I must acknowledge that all the observations and criticisms he made in the course of supervising the work had greatly enriched the Thesis. In fact, Dr Sani's contributions were indeed a lesson in themselves to me as his student.

I must thank our indefatigable Head of Department; Dr Y Jibrilfor ensuring the research work came to its logical conclusion. Equally important was the contribution of Dr. A. M. S. Takanyi who also read the work and made contributions that had greatly added value to it. I also have to mention Dr. A. D. Usman for his assistance in ensuring the work sees the light of the day. I sincerely thank all of them.

Prof. M.B Mu'azu's inspirational contribution will never be ignored. It was not my intention to further my university education beyond the first degree. Prof.Mu'azu inspired me to have a rethink, as he made sure I did not only pursue a master's degree but also a terminal degree which I have successfully completed. He is really an inspiring academic.

The contributions of Alhaji F. O. Sadiq (OgaFatai) to the success of this research must also be acknowledged. He gave me useful advice to achieve the required standard. I always see

OgaFatai as my mentor, as he has been guiding me in the affairs of my studies since I was an undergraduate student when he even personally taught me how to construct different types oftransformers as an electrical engineering student.

Equally, I would want to thankMalamTsanni of the National Agricultural Extension Research Liaison Service (NAERLS),Ahmadu Bello University, Zaria and Engr. (Alhaji) Sadiq Musa of Nigerian Television Authority (NTA) for allowing me to use their reading facilities to acquire the relevant data for this research work. Ialso thank Prof. BoyiJimoh for being quite helpful in his wisdom, advice, encouragement, and better still, for allowing me access to his personal library to get the right books for the work.

Iwould also like to express my appreciation to Dr. K. A. Abu-Bilal, Engr. B. O. Sadiq, and A. Muhammed (Minister) for their immense individual contributionsto the success of thisresearch work. I must acknowledge the encouragement of my colleagues, especially Dr Yusuf Sha'aban, Dr. T. H. Sikiru and Engr. M. Jibrin (Muazu Junior).

I cannot, in any way, forget the kind support and encouragement of myGeneral Manager AlhajiNasiru Abdul of AhmaduBello University,Samaru FM, where I workand MalamAuwalu Umar of the Faculty of Social Sciences, Ahmadu Bello University, Zaria. Others acknowledged for their support, include Engr. ZubairuInusaAteiza, Engr. AbdullahiHassanand Engr. Ya'uMukhtarall of the Equipment Maintenance Development Centre, Ahmadu Bello University, Zaria. I am also appreciative of the contributions, in one way or the other to the success of this academic work by several others too numerous to mention here. I am indeed grateful to all. And, to my family, Sa'adatu ,Ummi, Fatimah, Aisha Al'amin, Maryam, Abdulrahaman and Zainab, for their understanding as the work consumed most of my time and attention.

ABSTRACT

Research attention has been focussed on the reduction of image data size (major problem) for its efficient compression, storage, and transmission. In this work, the developed brightness enhancement model was used to enhance the Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT), Lifting Wavelet Transform (LWT), and Firefly Optimization Algorithm (FOA) compression, which were then used to compress the six video data. These video samples were obtained from the cameras of National Agricultural Extension Research Liaison Service (NAERLS),Ahmadu Bello University, Zaria (as NAERLS1.avi and NAERLS2.avi), Nigerian Television Authority (NTA), Abuja (as NTA1.avi and NTA2.avi), and the benchmark online media database (as Foreman.avi and Akiyo.avi). In the pre-processing stage, the video data were converted into frames of pictures for easy analysis. The hue and saturation were then extracted and the images were noised and filteredusing MATLAB R2014b simulation environment. Based on the analysis of random variation of pixel intensity and histogram distribution, the developed brightness enhancement technique was used to improve frame signal as a result of loss during compression. The performances of various enhanced compression techniqueswere evaluated through a number of MATLAB R2014b simulations using Peak signal to noise ratio(PSNR) as a performance metric. The results showed that the PSNR values for the grey level (black and white) imageswere improved by 31.95dB and 22.30dB for NAERLS1.avi and NAERLS2.aviwhen subjected to brightness enhancement technique.Also, PSNR improvements of 17.71dB and 23.31dB were obtained for the NTA1.avi and NTA2.avi, respectively, as well as15.06dB and 19.17dB improvements were obtained for the Foreman.avi and Akiyo.avi benchmark samples respectively. Similarly, improvement in terms of PSNR was also registered when coloured images were subjected to the developed brightness enhancement technique. The research implemented four video compression techniques DCT, DWT, LWT, and FOA compression,which were used as benchmarks for the developed modified FOA (mFOA)compression technique. Their respective outputswereimproved using the developed brightness enhancement model in order to account for the loss of signal quality which mighthave occurredduring compression. PSNR simulation results showed that themFOA compression technique performed better than DCT, DWT, LWT, and FOA compression techniques. For example,before enhancement,it was found that the mFOAPSNR result was better than the LWT by 73.64%, 80.04%, 80.03%, and 80.40%,respectively for NAERLS1.avi, NAERLS2.avi, NTA1.avi and NTA2.avi captured video frames and an improvement of 75.78% and 77.56% for Akiyo.avi and Forman.avi benchmark video frames.The mFOA was also discovered to outperform the FOA by 7.34%, 3.30%, 4.90%, and 5.75% for NAERLS1.avi, NAERLS2.avi, NTA1.avi and NTA2.avi captured video frames before enhancement and an improvement of 3.56% and 3.86% for Akiyo.avi and Forman.avi benchmark video frames. Similarly, the enhanced mFOA (E-mFOA) compression technique also producedPSNR improvement of 72.09%, 79.04%, 79.51% and 78.81% over enhanced LWT (E-LWT) for NAERLS1.avi, NAERLS2.avi, NTA1.avi and NTA2.avi capture video frames and an improvement of 74.67% and 76.08% for Akiyo.avi and Forman.avi benchmark video frames.The E-mFOA compression technique also produced a better PSNR improvement of 4.59%, 1.14%, 2.08%, and 1.17% over E-FOA for NAERLS1.avi, NAERLS2.avi, NTA1.avi and NTA2.avi captured video frames, except for the Akiyo.avi and Forman.avi benchmark video frames, where an insignificant improvement of 0.41% and -0.06%

were registered. These might have been as a result of the low level of light present when the video clips were taken.

TABLE OF CONTENTS

Title page- - - - - - - - - - - i

Declaration- - - - - - - - - - - ii

Certification- - - - - - - - - - - iii

Dedication- - - - - - - - - - - iv

Acknowledgements- - - - - - - - - - - v

Abstract- - - - - - - - - - - vii

Table of Contents- - - - - - - - - - ix

List of Figures- - - - - - - - - - xiii

List of Tables-- - - - - - - - - - xiv

Abbreviations And Acronyms- - - - - - - - - xvii

Mathematical Symbols - - - - - - - - - - xxi

CHAPTER ONE: INTRODUCTION

1.1 Background- - - - - - - - - - 1

1.2 Aim and Objectives- - - - - - - - - - 3

1.3 Statement of Problem- - - - - - - - - 4

1.4 Scope and Limitations- - - - - - - - - 5

1.5 Research Motivation- - - - - - - - - 5

1.6 Hypothesis- - - - - - - - - - 6

1.7 Research Methodology- - - - - - - - - 6

1.8 Thesis Organization- - - - - - - - - 7

CHAPTER TWO: LITERATURE REVIEW

2.1 Introduction- - - - - - - - - - 9

2.2 Review of Fundamental Concept- - - - - - - - 9

2.2.1 Different Types of Digital Image- - - - - - - - 9

2.2.1.1 Image layout - - - - - - - - 10

2.2.1.2Image Colour - - - - - - - - 10

2.2.2 Digital Image Definitions- - - - - - - - 13
2.2.2.1 Common Values- - - - - - - - - 14
2.2.3 Resolution and Quantization - - - - - - - 16
2.2.4 Digital Video - - - - - - - - - 18
2.2.4.1 Digital Image Statistics - - - - - - - 18
2.2.5 Digital video Processing- - - - - - - - 22
2.2.5.1 Video Quality Measure- - - - - - - - 22
2.2.6 Pixel Operations - - - - - - - - 24
2.2.6.1. Intensity Scaling - - - - - - - 24
2.2.6.2 Image Histogram Equalization - - - - - - - 25
2.2.6.3 Histogram Shaping - - - - - - - 27
2.2.7 Arithmetic Operations between Images - - - - - 29
2.2.8 Image Noise - - - - - - - - - 30
2.2.8.1 Sources of Noise in Images- - - - - - - - 30
2.2.8.2 Different Noise Types- - - - - - - 31
2.2.9 Image De-noising - - - - - - - - 34
2.2.10 Linear Filters and Non Linear Filters - - - - - - 35
2.2.10.1 Different Type of Linear and Non-Linear Filters- - - - - 35
2.2.11 Fundamental Steps in Image Processing - - - - - 37
2.2.12 Methods of Noise Suppression - - - - - 39
2.2.13 Images Differences for Change Detection - - - - - 41
2.2.14 Frequency Domain Techniques - - - - - - 42
2.2.15 Video and Image Compression - - - - - - 45
2.2.15.1 Huffman Coding- - - - - - -46
2.2.15.2 Arithmetic Coding- - - - - - 46

2.2.15.3 Substitution (Dictionary based) Coding- - - - 47

2.2.15.4 Sample/based Coding- - - - - - - - 47

2.2.15.5 Transform Domain Coding- - - - - - 48

2.2.15.6 Wavelet-based Coding- - - - - - - - 49

2.2.16 Discrete Cosine Transform - - - - - - - 51

2.2.17 Discrete Wavelet Transform - - - - - - - 54

2.2.18 Lifting Wavelet Transform - - - - - - - 55

2.2.19 Purpose of Image and Video Processing - - - - - 55

2.2.20 Application of Image Processing - - - - - - 56

2.2.21 Firefly Optimization Algorithm - - - - - - 56

2.2.22 Performance Metric - - - - - - - - 61

2.3 Review of Similar Works - - - - - - - 62

2.3.1 Review of Similar Works on Image Processing - - - - 62

CHAPTER THREE: MATERIALS AND METHODS

3.1 Introduction- - - - - - - - - - 84

3.2Materials used in this research work - - - - - 84

3.3Research Methodology- - - - - - - - 85

3.3.1 Pre-Processing- - - - - - - - - 85

3.3.2 Video Acquisition- - - - - - - - - 85

3.3.3Eliminate Hue and Saturation and Retain Luminance Intensity- - - 88

3.3.4 Noising and Filtering - - - - - - - - - 89

3.3.5 Brightness Enhancement- - - - - - - - 89

3.4 Enhanced Discrete Wavelet Transform- - - - - - 94

3.5Enhanced Discrete Cosine Transform- - - - - - 98

3.6Enhanced Lifting Wavelet Transform- - - - - - 101

3.7Compression Using Fire Fly Algorithm- - - - - 105

CHAPTER FOUR: RESULTS AND DISCUSSION

4.1 Introduction- - - - - - - - - - 111

4.2 Results- - - - - - - - - - 111

4.3 Performance Evaluation Using Peak Signal to Noise Ratio- - - - 121

4.4 Compression Analysis- - - - - - - - - 124

4.4.1 Comparison of Techniques before Enhancement- - - - 129

4.4.2 Comparison of Techniques after Enhancement- - - - - 129

4.5 Minimization Plots - - - - - - - - - 130

CHAPTER FIVE: SUMMARY, CONCLUSION AND RECOMMENDATIONS

5.1 Introduction- - - - - - - - - - 134

5.2 Summary of Findings - - - - - - - - - 134

5.3 Conclusion- - - - - - - - - - 135

5.4 Significant Contributions- - - - - - - - 137

5.5 Recommendations for Further Research- - - - - - - 139

REFERENCES- - - - - - - - - - 140

Appendix A
M-file implementation- - - - - - - - 153

Appendix B
M-File Implementation for Coloured Frames- 156

Appendix C
M-file implementation of Discrete WaveletTransform - 157

Appendix D:
M-File Implementation of Discrete Cosine- Transform 160

Appendix E
M-File Implementation of Lifting WaveletTransform - 162

Appendix F
M-Implementation of Fire Fly Algorithm 167

Appendix E2
M-file of objective function 171

Appendix F2
M-Implementation of Modified Fire Fly Algorithm (mFOA) 174

LIST OF FIGURES

Figure 2.1 The 2-D Cartesian coordinate Space of an M x N Digital Image 11

Figure 2.2 Digitization of a Continuous Image. 14

Figure 2.3 OriginalImage and Its Characteristics 20 Figure 2.4Histograms of Grey Level Images 27

Figure 2.5: Image and its Histogram 28

Figure 2.6 Original Image withoutand with Noise 32

Figure 2.7: Effect of Gaussian Noise with Zero Mean on an Image 32

Figure 2.8 Image with Poisson Noise 33

Figure2.9 Image with Speckle Noise 33

Figure2.10 Use of Mean Filter on Noise Images 36

Figure 2.11 Use of Median Filter on Noise Images (Salt & Pepper and Gaussian) 37

Figure 2.12Use of Median filter on Noise Images (Poisson and Speckle) 37

Figure2.13 Fundamental steps in Digital Image Processing 39

Figure 2.14Filtering with the Butterworth Filter 45

Figure 2.15 System with Typical Processes for Data Compression 47

Figure 2.16 Major Types of Probability Sampling 48

Figure 2.17 Discrete Cosine based Coding 49

Figure 2.18Block Diagram of the 1D Dyadic Wavelet Transform 50

Figure 3.1: Frames of Sampled Videos and Benchmark Videos. 86

Figure 3.2: Flowchart Implementation of Luminance Enhancement Technique 92

Figure 3.3: Block Diagram Representation of the Research Methods 94

Figure 3.4: Flowchart of MATLAB Implementation of Improved DWT Video Image 97

Figure 3.5: Flowchart of MATLAB Implementation of Improved DCT Video Image 100

Figure 3.6: Flowchart of MATLAB Implementation of Improved LWT Video Image 103

Figure 3.7: Block Diagram of the achieved result 104

Figure 3.8: Flow Chart of Compression Using Firefly Algorithm 107

Figure 3.9 Flow ChartImplementation of Modified FOA 110

Figure 4.1: Luminance Intensity of Black and White Video Frames 111

Figure 4.2: Luminance Intensity of Black and White Video Frames after Enhancement 112

Figure 4.3: Histograms of Original and Improvement NAERLS Sample Images 113

Figure4.4: Histograms of Original and Improvement NTA Sample Images 114

Figure 4.5: Histograms of Original and Improvement Benchmark Sample Images 115

Figure 4.6: Coloured Images and Histograms of Original and Improvement NAERLS1 Sample 116

Figure 4.7: Coloured Images and Histograms of Original and Improvement NAERLS2 Sample 117

Figure 4.8: Coloured Images and Histograms of Original and Improved NTA1 118

Figure 4.9: Coloured Images and Histograms of Original and Improved NTA2 Sample119

Figure 4.10: Coloured Images and Histograms of Original and ImprovedAkiyo Sample 120

Figure 4.11: Coloured Frame of Foreman before and After Enhancement 121

Figure 4.12: Brightness Enhancement Comparison on Grey scale and Coloured Frames 123

Figure 4.13: Percentage Reduction in Size 125

Figure 4.14: Minimization plot of Benchmark Video Frames 131

Figure 4.15: Minimization plot of NAERLS Video Frames 131

Figure 4.16: Minimization plot of Benchmark Video Frames 132

LIST OF TABLES

Table 2.1 Common values of digital image parameters 14

Table 2.2 The Peak Signal to Noise Ratio and its description 23

Table 3.1: Sample Video Data 87

Table 4.1: Simulation Result ofPerformance Evaluation Using PSNR on Hue and Saturation Free Frames 122

Table.4.2 Simulation Result ofPerformance Evaluation using PSNR on Coloured Frames. 122

Table 4.3: Simulation Result ofPerformance Comparison of the Sample Size after Compression 124

Table 4.4 Simulation Result of Compression Ratio Analysis 126

Table 4.5: Table 4.5: Simulation Results of Peak Signal-to-Noise Ratio (PSNR) of Various Techniques and after their Enhancement 128

Table 4.6: Percentage Improvement of mFOA over LWT and FOA for PSNR 128

Table 4.7: Percentage Improvement of Enhanced mFOA over Enhanced LWT and FOA for PSNR 128

ABBREVIATIONS AND ACRONYMS

FOA FireflyOptimization Algorithm

NAERLS: National Agricultural Extension Research Liaison Service

NTA: Nigerian Television Authority

PSNR: Peak Signal to Noise Ratio

DCT: Discrete Cosine Transform

DWT: Discrete Wavelet Transform

IDCT: Inverse Discrete Wavelet Transform

LWT: Lifting Wavelet Transform

dB: Decibel

MB: Mega Bite

R, G, B: Red, Green, Blue

CCD: Coupled Charged Device

2- D: Two Dimensional

FFT: Fourier Transform

SNR: Signal–to–Noise Ratio

MSE: Mean Square Error

AOD: Average Optical Density

Q(x): Cumulative Histogram Function

JPEG: Joint Picture Expert Group

ME: Motion Estimation

MC-P: Motion-Compensated Prediction

MPEG: Motion Picture Expert Group

DRLSE: Distance Regularized Level Set Evolution

HDTV: High Definition Television

DRLSE: Distance Regularized Level Set Evolution

LS: Lifting Scheme

HE: Histogram Equalization

EA: Evolutionary algorithms

MR: Magnetic Resonance

SSR & MSR: Modern techniques Retinex

JND: Just Noticeable Distortion

DPCM: Differential Pulse Code Modulation

RDOQ: Rate- Distortion Optimized Quantization

BBHE: Brightness Preserving Bi-Histogram Equalization

DSIHE: Dualistic Sub-Image Histogram Equalization

RMSHE: Recursive Mean-Separate HE Method

MMBEBHE: Minimum Mean Brightness Error Bi-HE Method

DHE: Dynamic Histogram Equalization

BPDHE: Brightness Preserving Dynamic Histogram Equalization

MMBEBHE: Minimum Mean Brightness Error Bi-HE Method

VSR: Video Super-Resolution

SD: Standard Dentition

HD: High Dentition

SR: Super-Resolution

DOG: Difference Of Gaussian

MICO: Multiplicative Intrinsic Component Optimization

RODE: Recursive Optimal distribution Estimation

GAC: Guided Active Contour

HSI: Hyper Spectral Imagery

LBP: Local Binary Patterns

RMSE: Root Mean Square Error

AVI: Video Interleaved file

SVD: Singular Value Decomposition

HVS: Human Visual System

MSE: Mean Square Error

ROI: Region of Interest

SPIHT: Set Partitioning in Hierarchical Trees

NPD: Normalized Pixel Difference

CRC: values (improved robustness of watermark).

PLSR:	Partial Least Squares Regression
SSIM:	Structural Similarity Index
UIQI:	Universal Image Quality Index
PLSR:	Partial Least Squares Regression
CRC:	Improved Robustness of Watermark
LSMC:	L Shaped Morpho Codec
bpp:	bits per pixel
VLC	block encode
NTSC	National Television System Committee
PAL	Phase Alternating Line
SECAM	Sequential Colour Memory

MATHEMATICAL SYMBOLS

$P(a)$: Probability Distribution Function

h[a]: Histogram

Λ: Number of pixels

ma: Average Brightness of a Region

$\Re$: Region

Sa: Standard Deviation

CV: Coefficient-of-Variation

CHAPTER ONE

INTRODUCTION

1.1 Background

Digital image processing has rapidly evolved over the decade, with increasing applications in the field of engineering and science. The image processing is a visual task, which involves image acquisition, enhancement and processing at the final stage. The image processing mostly involves taking an array of picture elements (pixels) as input and producing an array as output pixels which usually represent an improvement on the original picture. Image enhancements are employed in order to increase the contrast of the image, thereby enabling the visualization of the distinct features of the image. This usually augments the efficiency of image classification and interpretation. The digital image is considered as a large array of discrete dots, each of which has a brightness associated with it. These dots are called picture elements or simply pixels (McAndrew, 2004). Usually, captured images are often not a true reflection of the real object(s) the image is representing. For an efficient image processing, it is important to address some of the unwanted image background information (which cause a random variation in pixel intensity) by way of pre-processing. The main challenge for the pre-processing systems is that the captured images are often associated with low resolution. Although cameras on most mobile devices are capable of taking higher resolution images, the computation cost is still an issue nowadays (Optical character recognition) (Yu *et al* 2012). Image pre-processing is the term for operations carried out on images at the lowest level of abstraction. These operations do not increase image information content. The aim of pre-processing is an improvement of the image data that suppresses undesired distortions or enhances some image features such as edges and lane, etc relevant for further processing and analysis task. Image pre-processing used the redundancy in images. Neighbouring pixels corresponding to one real object have the same or similar brightness value. If a distorted pixel

is picked out from the image, it is restored as an average value of neighbouring pixels (Olgao, 2009). Image pre-processing methods are classified into categories according to the size of the pixel neighbourhood that is used for the calculation of new pixel brightness. Video data usually contain large amount of bits which makes it difficult for transportation and file shearing. This has prompted researchers in image processing to seek for ways of reducing this large amount of bits in the form of compression without necessarily damaging the image quality.

Compression refers to the process of reducing the number of bits required to represent the image and video *(*Vishnu*et al.* 2013*).* The main objective of video compression is to overcome the cost of transmission and required bandwidth. The size of video is also a factor that affects its transmission (Catania, 2008). Digital image compression comes in two forms namely lossless and lossy. The lossless compression is a process to reduce image or video data for storage and transmission while retaining the quality of original image (that is, the decoded image quality is required to be identical to image quality prior to encoding) *(*Vishnu*et al.* 2013*).* In lossy compression, on the other hand, some information present in the original image or video is discarded so that the original raw representation of image or video can only be approximately reconstructed from the compressed representation with high compression ratio *(*Vishnu *et al.* 2013*).* In other words, compression is a reversible conversion (encoding) of data that contains fewer bits. This allows more efficient storage and transmission of the data. The inverse process is called decompression (decoding). Software and hardware that can encode and decode are called encoders and decoders, respectively. In video, compression becomes necessary because the correlation between one pixel and its neighbouring pixels is high the values of one pixel and its adjacent pixels are very similar. This is called the intra-frame correlation in video compression because it is the correlation in a single frame. Once the correlation between the pixels is reduced, the storage quantity is

equally reduced (Djordje, 2006). The image compression method is also applied to video compression. However, there exists also temporal correlation. The video is composed of a large number of still images that are taken at short time distance of which any two neighbouring images are similar. Therefore, it is known that there exists high correlation between the images or frames in the time direction (Djordje, 2006). The correlation in the time direction is called the intra-frame correlation. If the intra-frame correlation can efficiently be reduced, then video compression can be achieved (Wei, 2010). This research focuses on developing an enhancement technique for efficient video compression using Firefly Optimization Algorithm (FOA), Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT), and Lifting Wavelet Transform (LWT).

Fireflies are amongst the most charismatic of all insects and their spectacular courtship displays have inspired poets and scientists alike. Fireflies are characterized by their flashing light produced by biochemical process known as bioluminescence. Such flashing light may serve as the primary courtship signals for mating. Besides attracting mating partners, the flashing light may also be used to serve as warning to potential predators *(*Fister *et al.*, 2013). These interestingcharacteristics of natural Fireflies inspired the development of FOA(Yang, 2008).

1.2 Aim and Objectives

The aim of this research work is to develop a video frame enhancement technique and modify Firefly Optimization Algorithm (mFOA) for improvedvideo compression.

In order to achieve this stated aim, the following objectives were employed:

1. To acquire video data and apply pre-processing technique in order to reduce the effect of background noise, as well as to develop a video brightness (luminance) enhancement modeland modify FOA for improving the visual quality of the video data using pixel intensity and histogram distribution.

2. To implementthe enhanced compressed sampled videos using Discrete Cosine Transforms (DCT), Discrete Wavelet Transform (DWT), and Lifting Wavelet Transform (LWT), FOA, as well as themFOA to obtain their respective coefficients.
3. To enhance the transformed coefficients developed in item2using the developed enhancement technique in item 1.
4. To estimate the quality of the final processed sampled video data using peak signal to noise ratio (dB), compression ratio, and size reduction efficiency(bytes) which are the fundamental metrics for measuring the performance of these compression techniques.

1.3 Statement of Problem

The three main challenges today in image and video enhancement are size of video window, frame rate and very much importantly quality of image or video. Digital video files are large, which makes them difficult for easy transportation and storage. Various quantization techniques and algorithms have been proposed in order to reduce the size of an image by reducing the number of colour content in a digital image(s) while preserving the significant information. However, high amount of quantization may lead to signal distortion between image regions and may not be acceptable to human visual perception. Image signal distortions are largely interpreted as a random variation in pixel intensity which results in an uneven distribution of image histogram, which is a problem in image compression. A lot of researchers have proposed various techniques for reducing large amount of size associated with video and enhanced efficient transportation and storage. Most ofthese methods deliver high reduction of image size in the form of compression. Usually, higher compression ratio is associated with lower size and quality degradation, which is also another problem associated with compression. Furthermore, the video signal encoding and decoding requires a high amount of computational resources and it is difficult for a real time application with low bandwidth requirement to compress a video with a computational expensive algorithm which

may take too long to encode and decode video data. This is also another limitation in image processing. Thus, this thesis developed a video frame enhancement technique based on pixel intensity, and histogram distribution for an improved video compression.

1.4 Scope and Limitations

i. The scope of this thesis is to produce a good quality video by developing and implementing brightness enhancement technique and Firefly Optimization Algorithm to achieve better video resolution output quality for the true analysis of the improvement rendered by these developed techniques compared to the standard ones. This research did not consider the following:

ii. The audio content of the video data since the focus is on enhancing the visual content and size reduction.

iii. The environmental changes and conditions of device used for capturing the video data.

iv. The practical hardware components required for implementation of the system. The limitations of this research are highlighted as follows:

v. The standard high resolution video cameras for capturing the video data are not available. Thus, the research made use ofthe cameras obtained from National Agricultural Extension Research Liaison Service (NAERLS) and Nigerian Television Authority (NTA).

vi. Due to lack of high speed computer and storage devices, the video data size was limited to less than 50MB.

1.5 Research Motivation

Video compression is considered one of the most important aspects of digital image processing. The analysis of video data is complicated due to its relative large size. This has

posed a challenge in video data transportation and loss of video data due to insufficient amount of storage devices available. Several research efforts have been directed toward reducing the large amount of size associated with these videos in the form of transformations and compression with a promising result. However, this approach is usually associated with loss of signal and reduction in quality of the video signal. These challenges necessitate the development of a luminance enhancement technique.

1.6 Hypothesis

The fundamental video processing problem is the relationship between the quality of the actual object in the video and the background object. The only explicitly known contents of the video are the hue, saturation and luminance. However, the intensity of the luminance is what determines the contrast level of the video. Thus, this thesis hypothesised that the contrast level or the brightness level of a video is dependent only on the luminance intensity of the video. Hence, an approach which focuses on improving the luminance intensity level of a video base pixel intensity and histogram distribution can present an efficient enhancement in video visual quality.

1.7 Research Methodology

The step by step procedures adopted in this research, which include the development of an image brightness enhancement and modifying FOA in order to achieve better image quality are highlighted as follows:

1. **Pre-processing:**
    i. Acquisition of video sample data and the online benchmark data base for analysis.
    ii. Elimination of hue and saturation in order to retainonly luminance intensityof the sample video data to pave way for luminance (grey level) enhancement.

iii. Noising and filtering of the video data such that its pixel is varied from its true value in order to separate the true picture from its background for easy of analysis.

iv. Applying the developed brightness enhancement techniqueto improve the luminance intensity of the video frames obtained from NEARLS and NTA to determine the improvement achieved in the image quality.

2. Video Compression:

i. Implementing the standard Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT), Lifting Wavelet Transform (LWT), FOA, and mFOAcompression techniques on images obtained from NEARLS and NTA to know the improvement achieved by the modified FOA.

ii. Applying the developed enhancement technique on both the standards and mFOA compressed images.

iii. Carrying out analysis of the results obtained in item 3 and validation of these results to ascertain the improvement of the modified FOA compression technique.

1.8 Thesis Organisation

The general background information relevant for the course of this research work has been presented in chapter one. The brief overview of the remaining chapters' layout is as follows: Chapter two presents detail fundamental existing works that border on the areas of investigation of this research. Studies cover major components that are addressed in this research. Chapter two provides the theoretical basis and the body of knowledge surrounding the investigation in this work. Chapter three presents the in-depth approach and relevant mathematical models necessary for the successful implementation of this research work. Chapter four presents the analysis, performance, and discussion of the results obtained during

simulation. Chapter five gives the summary of the main findings, conclusions of this research, its contributions, and suggestions for further research.

CHAPTER TWO

LITERATURE REVIEW

2.1 Introduction

This chapter is divided into two sub-sections. Subsection 2.2 describes the fundamental concepts relevant to this research work. Section 2.3 presents a summary of the critical review of similar research works in chorological order based on the previous researches available in the literature.

2.2 Review of Fundamental Concepts

In this subsection, detail concepts which are fundamental to the proposed research work are presented. Knowledge in this area aid the understanding of approaches, techniques, and tools used in research in this area of image processing by previous researchers. It also gives an indication of the effectiveness of these aspects.

2.2.1 Different Types of Digital Images

The human eye senses the variation of light taking place in its view field and forms an image of the scene on the retina. In other words, camera picks up a portion of the scene falling in its field of view and forms its image on a semiconductor plate (as in TV camera tube) (Katiyar, 2012).

Four types of digital image exist, namely (McAndrew, 2004):

i. Binary Image

ii. Grey scale Image

iii. True Colour (RGB) image and

iv. Indexed image

Binary image: In binary image, each pixel is either black or white and has two possible values (0, 1). Only one bit is needed to represent a pixel. Data for which a binary representation may be suitable include text and architectural plans (McAndrew, 2004).

Grey scale Image:The Grey scale image is an image in which each pixel is a shade of grey, that is, from 0 (black) to 255 (white). This range means that each pixel can be represented by 8 bits, or exactly one byte. The grey levels are suitable for representing any type of natural images. Other grey scale range exist (-255 to 255), such images representations are often used in medical imaging applications (McAndrew, 2004).

True colourImage: In the true colour image, each pixel has a particular colour which is described by the amount of Red, Green, and Blue (RGB) in it. With the image component range of 0 to 255, this gives a total $(256)^3$ **= 16, 777,216 different possible colours in the image. Since the number of bits required for representing each pixel is 24 bits, such images are referred to as 24-bits colour images (McAndrew, 2004).**

Indexed Image: In the index image type, the images only have a small subset of the $(256)^3$**possible colours. For ease of storage and data handling the image has an associated colour-map, colourpalette which is simply a list of all the colour components in the image. As opposed to the RGB image, each pixel of the indexed image has a value which does not give its colour but an index to the colour in the map (McAndrew, 2004).**

A digital image can be considered as a discrete representation of data possessing both spatial (layout) and intensity (colour) information. Image can also be treated as a multidimensional signal (Chris and Toby, 2010).

2.2.1.1 *Image layout*

The two-dimensional $2-D$ discrete digital image $I(\mathrm{m},\mathrm{n})$ represents the response of some sensors (or simply a value of some interest) at a series of fixed positions

$m = 1, 2, ..., M; n = 1, 2, ..., N$ in 2-D Cartesian coordinates. It is derived from the 2-D continuous spatial signal $I\ x, y$ through a sampling process naturally with certain types of image sensors (such as cameras CCD).These effect a local averaging of the continuous signal over some small region (typically square) in the receiving domain (Chris and Toby, 2010). The indices m and n respectively designate the rows and columns of the image. The individual picture elements or pixels of the image are thus referred to by their 2-D (m, n) index.

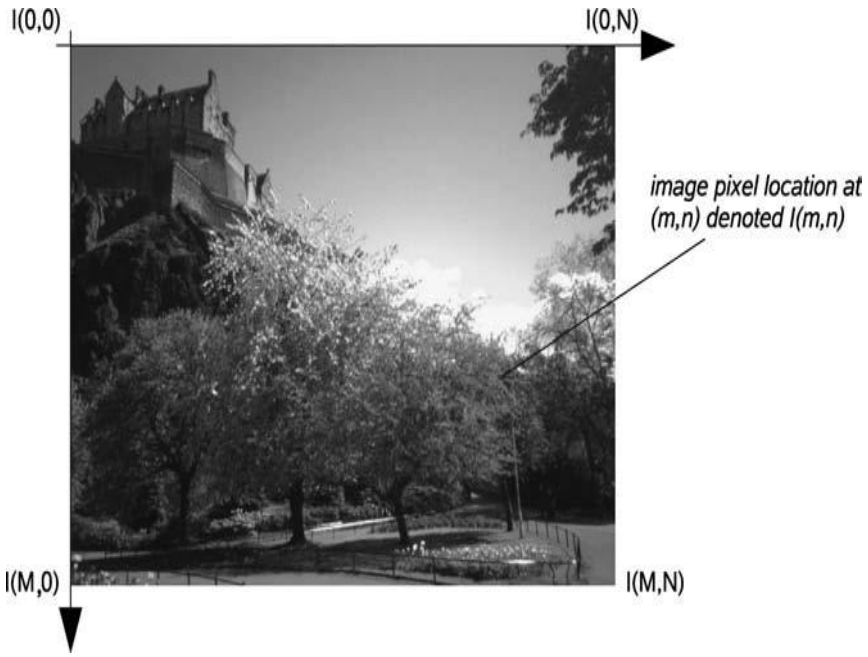


Figure 2.1: 2-D Cartesian Co-ordinate Space of an M x N Digital Image

Following the Matlab convention, I (m, n) denotes the response of the pixel located at the m^{th} row and n^{th} column starting from a top-left image origin in Figure 2.1. In other imaging systems, a column row convention may be used and the image origin in use may also vary (Chris and Toby, 2010).

2.2.1.2 *Image Colour*

An image contains one or more colour channels that define the intensity or colour at a particular pixel location I(m, n). In the simplest case, each pixel location only contains a

single numerical value representing the signal level at that point in the image. The conversion from this set of numbers to an actual (displayed) image is achieved through a colour map (Chris and Toby, 2010). A colour map assigns a specific shade of colour to each numerical level in the image to give a visual representation of the data. The most common colour map is the grey scale, which assigns all shades of grey from black (zero) to white (maximum) according to the signal level. The grey scale is particularly well suited to intensity images, namely images which express only the intensity of the signal as a single value at each point in the region (Chris and Toby, 2010).In addition to grey scale images where a single numerical value at each pixel location, there exist also the true colour images, where the full spectrum of colours can be represented as a typical triplet vector of RGBcomponents at each pixel location. Here, the colour is represented as a linear combination of base colours or values and the image may be considered consisting of three 2-D planes. Other representations of colour are also possible and used quite widely, such as the Hue, Saturation, and Intensity (HSI) or Hue, Saturation, and Value (HSV) (Chris and Toby, 2010).In these representations all three RGB components need to be of equal band width to generate any colour within the RGB colour cube(Keith, 2005). The result of this is a frame buffer that has the same pixel depth and display resolution for each RGB component. Also, processing an image in the RGB colour space is usually not the most efficient method(Keith, 2005). For example, to modify the intensity or colour of a given pixel, the RGB values must be read from the frame buffer, the intensity or colour calculated, the desired modifications performed, and the new RGB values calculated and written back to frame buffer (Keith, 2005).The YUV colour space is used by PAL (Phase Alternation Line), NTSC (National Television System Committee), and SECAM (Sequential Colour with Memory) composite colour video standards though other colour space exists like the YIQ and YCbCr (Keith, 2005). The black and white system used only luminance (Y) information; colour information chrominance (U andV) was added in

such a way that a black and whitereceiver would still display a normal black and white picture. Colour receivers decoded the additional colour information to display a colour picture (Keith, 2005). The basic luminance (brightness) equation is given as

$$Y = 0.299R + 0.587G + 0.114B \quad (2.1)$$

where R represents red, G for green and B for blue respectively.

2.2.2 Digital Image Definitions

The digital image is defined as a spatially distributed intensity signal $f(m,n)$, where f is the intensity of the pixel, and m and n define the position of pixel along a pair of orthogonal axes usually defined as horizontal and vertical. It is assumed that the image has M rows and N column and that the digital image has P quantized levels of intensity (grey levels) with values ranging from 0 to P-1 (Isaac, 2000).

A digital image *a* [*m*, *n*] described in a 2-D discrete space is derived from an analog image *a* (*x*, *y*) in a 2-D continuous space through a sampling process that is frequently referred to as digitization (Ian, 1995). The effect of digitization is shown in Figure 2.1. The 2-D continuous image *a* (*x*, *y*) is divided into N rows and M columns. The intersection of a row and a column is termed a pixel. The value assigned to the integer coordinates [*m*, *n*] with $m = 0,1,2,...,M-1$ and $n = 0,1,2,...,N-1$ is *a* [*m*, *n*]. These are considered to be the physical signal that impinges on the face of a 2-D sensor, which is actually a function of many variables including depth (*z*), colour (λ), and time (*t*). Unless otherwise stated, the case of 2-D, monochromatic, static images given in Figure 2.2 is considered here (Ian, 1995).

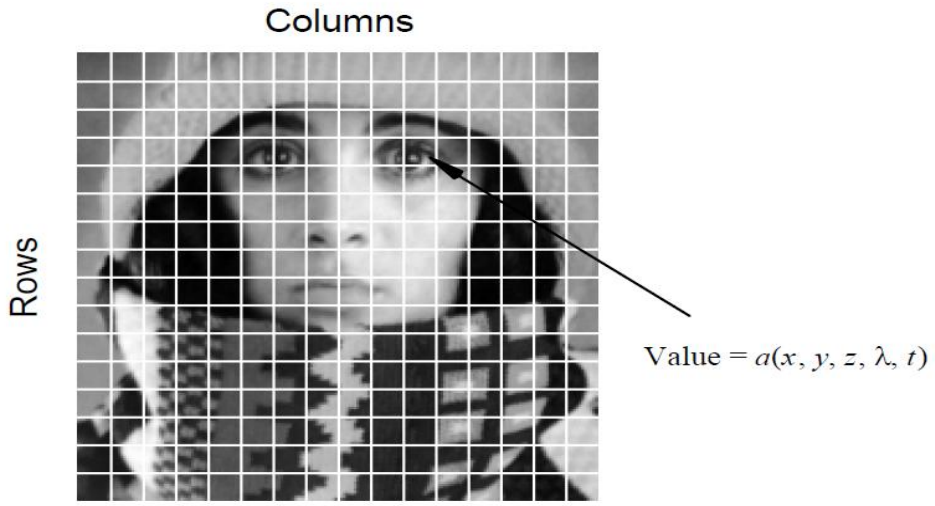


Figure 2.2: Digitization of a Continuous Image (Ian, 1995)

There are standard values for the various parameters encountered in digital image processing. These values are caused by video standards using algorithmic requirements, or by the desire to keep digital circuitry simple.

2.2.2.1 *Common Values*

The following table gives some commonly encountered values:

Table 2.1: Common Values of Digital Image Parameters (Ian, 1995)

Parameter	**Symbol**	**Typical values**
Rows	N	256,512,525,625,1024,1080
Columns	M	256,512,768,1024,1920
Grey Levels	L	2,64,256,1024,4096,16384

The Fourier transform is a representation of an image as a sum of complex exponentials of varying magnitudes, frequencies, and phases (Ian, 1995). The histogram of an image, commo nly used in image enhancement and image characterization is defined as a vector that contains the count of the number of pixels in the image at each grey level. The histogram $h(i)$, is defined as(Ian, 1995):

$h(i) = \sum_{m=0}^{M-1} \sum_{n=0}^{N} \delta(f(m,n) - i)$, i = 0, 1, ……., P-1, (2.2)

where

$$\delta(w) = \begin{cases} 1 & w = 0 \\ 0 & otherwise \end{cases}. \qquad (2.3)$$

A useful image enhancement operation is convolution using local operators, also known as kernels (Isaac, 2000). Considering a kernel $w(k,i)$ to be an array of (2K + 1 x 2L + 1) coefficients where the point (k,i)= (0, 0) is the centre of the kernel, convolution of the image with the kernel is defined by:

$$g(m,n) = w(k,i) * f(m,n)$$

$$= \sum_{k=-K}^{K} \sum_{l=-L}^{L} w(k,i).f(m-k,n-l), \qquad (2.4)$$

where $g(m,n)$ is the outcome of the convolution or output image (Isaac, 2000).

To convolve an image with a kernel, the kernel is centred on an image pixel (m, n), the point - to - point products of the kernel coefficients and corresponding image pixels are obtained, and the subsequent summation of these products is used as the pixel value of the output image at (m, n) (Isaac, 2000). The complete output image $g(m,n)$ is obtained by repeating the same operation on all pixels of the original image (Isaac, 2000). A convolution kernel can be applied to an image in order to effect a specific enhancement that may be desired. Attention is needed at the boundaries of the image where parts of the kernel extend beyond the input image. However, this approach can lead to artefacts at the boundaries of the output image (Isaac, 2000).In histogram equalization, the histogram of the input image is mapped to a new maximally flat histogram. The total number of pixels in the image M*N, is also the sum of all the values in $h(i)(\text{Isaac}, 2000)$. Thus, in order to distribute most uniformly the intensity profile of the image, each bin of the histogram should have a pixel count of (M*N)/ P, where P is the quantized level of intensity (grey levels) from 0 to P. It is possible to move a pixel with a given intensity to another intensity resulting in an increase in the pixels count in the new intensity bin. In order to achieve approximate uniformity, the average value of the pixel count over a number of pixel values can be made close to the uniform level (Isaac, 2000).

The Fourier transform $F(u,v)$ of an image $f(m,n)$ is defined as(Isaac, 2000):

$$F(u,v)= \frac{1}{MN}\sum_{M=0}^{M-1}\sum_{N=0}^{N-1} f(m,n)e^{-2\pi j(\frac{um}{M}+\frac{vn}{M})}, \qquad (2.5)$$

u = 0, 1, 2, ….., M – 1 v = 0, 1, 2, ……., N – 1, where u and v are the spatial frequency parameters. The Fourier transform provides the spectral representation of an image, which can be modified to enhance desired properties(Isaac, 2000). A spatial domain image can be obtained from a spectral domain image with the inverse Fourier transform given byIsaac, (2000):

$$f(m,n) = \sum_{u=0}^{M-1}\sum_{v=0}^{N-1} F(u,v)e^{2\pi j(\frac{um}{M}+\frac{vn}{M})} \qquad (2.6)$$

m = 0, 1, 2… M – 1, n = 0, 1, 2… N – 1.The forward or inverse Fourier transform of an N x N image computed directly with the preceding definitions requires a number of complex multiplications and additions proportional to N^2. By decomposing the expressions and eliminating redundancies, the Fast Fourier Transform (FFT) algorithm reduces the number of operations of the order of $N\log_2 N$. The computational advantage of the FTT is significant and increases with increasing N. When N = 64 the number of operations are reduced by an order of magnitude and when N =1024, by two orders of magnitude(Isaac, 2000).

The size of the 2-D pixel grid together with the data size stored for each individual image pixel determines the spatial resolution and colour quantization of the image.

2.2.3 Resolution and Quantization

The representational power (or size) of an image is defined by its resolution. The resolution of an image source (for example, a camera) can be specified in terms of three quantities (Chris and Toby, 2010) as follows:

i. **Spatial Resolution.** The column (C) by row (R) dimensions of the image defines the number of pixels used to cover the visual space captured by the image. This relates to the sampling of the image signal and is sometimes referred to as the pixel or digital resolution of the image. It is commonly quoted as C x R (for example, 640 x 480, 800 x 600, 1024 x 768, etc.) (Chris and Toby, 2010).

ii. **Temporal Resolution.** For a continuous capture system such as video, this is the number of images captured in a given time period. It is commonly quoted in frames per second (fps), where each individual image is referred to as a video frame, for example, commonly broadcast TV operates at 25 fps. 25–30 fps is suitable for most visual surveillance and higher frame-rate cameras are available for specialist science and engineering capture (Chris and Toby, 2010).

iii. **Bit Resolution.** This defines the number of possible intensity/colour values that a pixel may have and relates to the quantization of the image information. For instance a binary image has just two colours (black or white), a grey-scale image commonly has 256 different grey levels ranging from black to white whilst for a colour image it depends on the colour range in use (Chris and Toby, 2010). The bit resolution is commonly quoted as the number of binary bits required for storage at a given quantization level, for example, binary is 2 bits, grey-scale is 8 bits, and colour (most commonly) is 24 bits. The range of values a pixel may take is often referred to as the dynamic range of an image (Chris and Toby, 2010).

It is important to recognize that the bit resolution of an image does not necessarily correspond to the resolution of the originating imaging system. A common feature of many cameras is automatic gain, in which the minimum and maximum responses over the image field are sensed and this range is automatically divided into a convenient number of bits (that is, digitized into N levels) (Chris and Toby, 2010). In such a case, the bit resolution of the image

is typically less than that which is, achievable by the device. By contrast, the blind, unadjusted conversion of an analog signal into a given number of bits, for instance 2^{16} = 65536 discrete levels, does not, of course, imply that the true resolution of the imaging device as a whole is actually 16 bits. This is because the overall level of noise (random fluctuation) in the sensor and in the subsequent processing chain may be of a magnitude which easily exceeds a single digital level (Chris and Toby, 2010). The sensitivity of an imaging system is thus fundamentally determined by the noise and this makes noise a key factor in determining the number of quantization levels used for digitization. There is no point in digitizing an image to a high number of bits if the level of noise present in the image sensor does not warrant it (Chris and Toby, 2010).

2.2.4 Digital Video

Static images are derived from video cameras and frame grabbers. The standards that are associated with the three standard video schemes that are currently used worldwide are: NTSC, PAL, and SECAM (Ian *et al.,* 1995).

2.2.4.1 *Digital Image Statistics*

In image processing it is quite common to use simple statistical descriptions of images and sub images. The notion of a statistic is intimately connected to the concept of a probability distribution, generally the distribution of signal amplitudes. The probability distribution function of the brightness's in that region and the probability density function of the brightness's in that region can be defined. It is also assumed that the focus is on digitized image *a* [m,n] (Ian *et al.,* 1995).

i. **Probability Distribution Function of Brightness**

The probability distribution function, *P*(*a*), is the probability that a brightness chosen from the region is less than or equal to a given brightness value *a*. As *a* increases from

$-\infty$ to $+\infty$, $P(a)$ increases from 0 to 1. $P(a)$ is monotonic, non-decreasing in a and thus $dP/da \geq 0$ (Ian *et al.*, 1995).

ii. **Probability Density Function of Brightness**

The probability that a brightness in a region falls between a and $a+\Delta a$ is given by the probability distribution function $P(a)$ expressed as $p(a)\Delta a$, where $p(a)$ is the probability density function (Ian, 1995).

$$p(a)\, a = f(x) = \left(\frac{dp(a)}{da}\right)\Delta a \qquad (2.7)$$

Because of the monotonic, non-decreasing character of $P(a)$, it implies that:

$$p(a) \geq 0 \text{ and } \int_{-\infty}^{+\infty} p(a)\,da = 1 \qquad (2.8)$$

For an image with quantized (integer) brightness amplitudes, the interpretation of Δa is the width of a brightness interval (Ian *et al.*, 1995). It is assumed to be a constant width interval. The brightness probability density function is frequently estimated by counting the number of times brightness occurs in the region to generate a histogram, h[a] (Ian *et al.*, 1995). The histogram can then be normalized so that the total area under the histogram is 1, as inequation (2.9). In other words, the P[a] for a region is the normalized count of the number of pixels, Λ, in a region that have quantized brightness a: as shown in (2.9) (Ian *et al.*, 1995).

$$P[a] = \frac{1}{\Lambda}h[a]. \qquad (2.9)$$

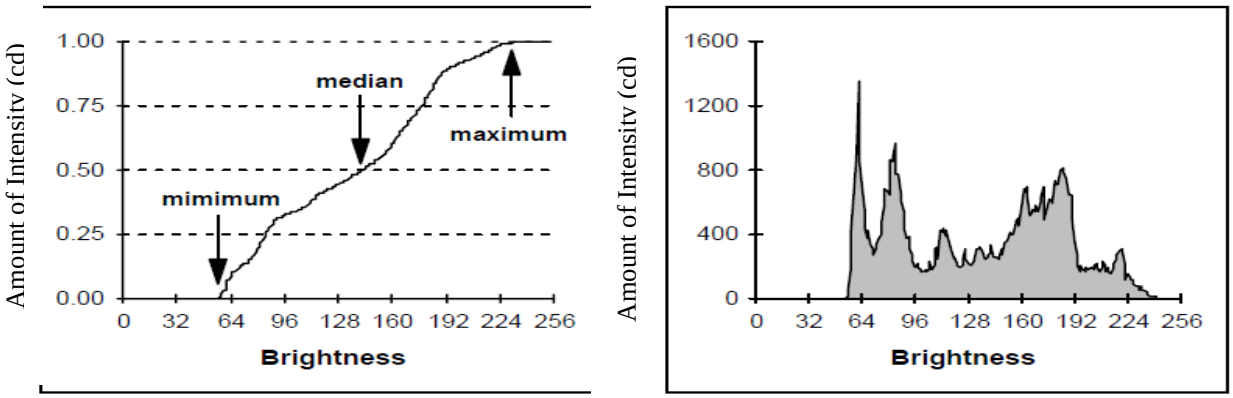
with,

$$\Lambda = \sum_a h[a] \qquad (2.10)$$

The brightness probability distribution function for the image $f(x)$ shown in Figure 2.2 is shown in Figure 2.3.



(a) Image



(b) Image Brightness Distribution Function(c) Image Brightness Histogram

Figure 2.3: Original Image and Its Characteristics(Ian et al., 1995)

Both the distribution function and the histogram as measured from a region are a statistical description of that region. It must be emphasized that both $P[a]$ and $p[a]$ should be viewed as estimates of true distributions when they are computed from a specific region (Ian et al., 1995). This implies that viewing an image at a specific region as one realization of the various random processes involved in the formation of that image and that region. In the same context, the statistics defined in (2.14) through (2.16) must be viewed as estimates of the underlying parameters (Ian et al., 1995).

i. Average

The average brightness of a region is defined as the sample mean of the pixel brightness within that region. The average, *ma*, of the brightness over the Λ pixels within a region (ℜ) is given by (Ian et al., 1995):

$$ma = \frac{1}{\Lambda}\sum_{(m,n)\in\Re} a[m,n] \qquad (2.11)$$

Alternatively, a formulation based upon the (un normalized) brightness histogram is used as:

$$h(a) = \Lambda \bullet p(a) \qquad (2.12)$$

with discrete brightness values *a* given as (Ian *et al.*, 1995):

$$ma = \frac{1}{\Lambda}\sum_{a} a \,.\, h[a] \qquad (2.13)$$

The average brightness, *ma*, is an estimate of the mean brightness, μ*a*, of the underlying brightness probability distribution (Ian *et al.*, 1995).

ii. Standard Deviation

The unbiased estimate of the standard deviation, *sa*, of the brightness within a region (ℜ) with Λ pixels is called the sample standard deviation and is given by (Ian, 1995):

$$Sa = \sqrt{(\frac{1}{\Lambda-1})\sum_{(m,n)\in\Re}(a[m,n]-ma)^2}$$

$$= \sqrt{\frac{\sum_{(m,n)\in\Re} a^2[m,n]-\Lambda ma^2}{\Lambda-1}} \qquad (2.14)$$

Using the histogram formulation, equation (2.13) becomes:

$$Sa = \sqrt{\frac{(\sum_{a} a^2.h[a])-\Lambda ma^2}{\Lambda-1}} \qquad (2.15)$$

The standard deviation, *Sa*, is an estimate of σ*a* of the underlying brightness probability distribution (Ian et al., 1995).

iii. Coefficient-of-Variation

The dimensionless coefficient–of–variation, CV, is defined as (Ian *et al.,* 1995):

$$CV = \frac{S_a}{ma} \text{x } 100\% \quad (2.16)$$

iv. **Signal–to–Noise Ratio**

The Signal–to–Noise Ratio(SNR) can have several definitions. The noise is characterized by its standard deviation, s_n. The characterization of the signal can differ. If the signal is known to lie between two boundaries, $a_{min} \leq a \leq a_{max}$, then the SNRis defined as (Ian *et al.,* 1995):

$$\text{Bounded signal SNR} = 20 \log_{10}\left(\frac{a_{max}-a_{min}}{s_n}\right) dB \quad (2.17)$$

If the signal is not bounded but has a statistical distribution then two other definitions are known:

In a stochastic signal S and N are inter-dependent and related as (Ian *et al.,* 1995):

$$\text{SNR} = 20 \log_{10}\left(\frac{ma}{s_n}\right) dB \quad (2.18)$$

2.2.5 Digital Video Processing

Digital video communication is found today in many application backgrounds such as broadcast services over satellite and terrestrial channels, digital video storage, wires and wireless conversational services, etc. The data quantity of digital video is very large for the memory of the storage devices and the finite bandwidth of the transmission channel. This makes it impracticable to store or transmit the full digital video without processing. Thus, several video compression algorithms have been developed to reduce the data quantity and provide an acceptable quality possible (Wei, 2010).

2.2.5.1 Video Quality Measure

For lossy coding, the rate-distortion theory was developed. Its main goal summarized the Rate-Distortion optimization criterion: The most popular distortion measure is the Mean

Square Error (MSE) (Catania, 2008). In order to evaluate the performance of video compression coding, it is necessary to define a measure to compare the original video and the video after compression. Most video compression techniques are designed to minimize the MSE between two video sequences Ψ_1 and Ψ_2 which is defined as (Wei, 2010):

$$\text{MSE} = \sigma_e^2 = \frac{1}{N}\sum_t \sum_{x,y}[\Psi_{1(x,y,t)} - \Psi_{2(x,y,t)}]^2 \qquad (2.19)$$

where N is the total number of frames in either video sequences.

Instead of the MSE, the Peak-Signal-to-Noise Ratio (PSNR)in decibel (dB) is more often used as a quality measure in video coding. The higher the PSNR, the better the degraded image is reconstructed to match the original image and also the better the constructive algorithm which is defined as(Zhengying *et al* 2014):

$$\text{PSNR} = 20\log_{10}\frac{255}{MSE} \qquad (2.20)$$

The PSNR is more commonly used than the RMSE, because people tend to associate the quality of an image with a certain range of PSNR. Table 2.2 illustrates the PSNR values and its indication (Mohamed*et al.*, 2015).

Table 2.2 The Peak Signal to Noise Ratio and its description(Mohamed et al., 2015)

PSNR	Description
Over 40 dB	Excellent image (i.e., being very close to the original)
Between 30 to 40 dB	Good image (i.e., the distortion is visible but acceptable)
Between 20 and 30 dB	Acceptable.
Lower than 20 dB	Unacceptable.

The MSE does not always reflect the real distortion perceived by human visual system. For practical purposes the PSNR is used (Catania, 2008).

If really MSE is to be used, then it should be computed between corresponding frames, then take the average of the resulting MSE values over all frames, and then convert the MSE value to PSNR (Wei, 2010).

2.2.6 Pixel Operations

In this section, it is intended to only consider image processing that depends only on the pixel grey level and do not take into account the pixel neighbourhood or whole image characteristics (Isaac, 2000).

2.2.6.1. *Intensity Scaling*

The intensity scaling is a method of image enhancement that can be used when the dynamic range of the acquired image data significantly exceeds the characteristics of the display system, or vice versa (Isaac, 2000). It may also be the case that image information is present in specific narrow intensity bands that may be of special interest to the observer. Intensity scaling allows the observer to focus on specific intensity bands in the image by modifying the image such that the intensity band of interest spans the dynamic range of the display (Isaac, 2000). For example, if f_1 and f_2 are known to define the intensity band of interest, a scaling transformation may be defined as

$$e = \begin{cases} f & f_1 \le f \le f_2 \\ 0 & otherwise \end{cases} \tag{2.21}$$

$$g = \left\{ \frac{e - f_1}{f_2 - f_1} \right\}. \ f_{\max} \tag{2.22}$$

where e is an intermediate image, g is the output image, and $f_{\max}$ is the maximum intensity of the display.

The image, however, is displayed such that all grey levels in the range 0 to 255 are seen (Isaac, 2000).

2.2.6.2 *Image Histogram Equalization*

The basic tool that is used in designing point operations on digital images (and many other operations as well) is the image histogram. The histogram H_f of the digital image f is a plot

orgraph of the frequency of occurrence of each grey level in f (Jerry, 2000). Hence, H_f is a one-dimensional function with domain (0, . . . ,K - 1) and its possible range extending from 0to the number of pixels in the image, NM. The histogram is given explicitly by

$$H_f(k) = J \tag{2.23}$$

It contains exactly J occurrences of grey level k, for each k= 0, . . . ,K - 1**.** Thus, an algorithm to compute the image histogram involves a simple counting of grey levels, which can be accomplished even as the image is scanned. Every image processing development environment and software library contain basic histogram computation, manipulation, and display routines.

Since the histogram represents a reduction of dimensionality relative to the original image$f$, information is lost – the image f cannot be deduced from the histogram H_f except in trivial cases (when the image is constant value). In fact, the number of images that share the same arbitrary histogram H_f is astronomical. Given an image f with a particular histogram, H_f every image that is a spatial shuffling of the grey levels of $f$has the same histogram H_f**.** The histogram H_f contains no spatial information about f and it describes the frequency of the grey levels in f and nothing more. However, this information is very rich, and many useful image processing operations can be derived from its histogram. Indeed, a simple visual display of H_f reveals much about the image (Jerry, 2000). By examining the appearance of a histogram, it is possible to ascertain whether the grey levels are distributed primarily at lower (darker) grey levels, or vice versa. Although this can be ascertained to some degree by visual examination of the image itself, the human eye has a tremendous ability to adapt to overall changes in luminance, which may obscure shifts in the grey-level distribution. The histogram supplies an absolute method of determining an image's grey-level distribution (Jerry, 2000).

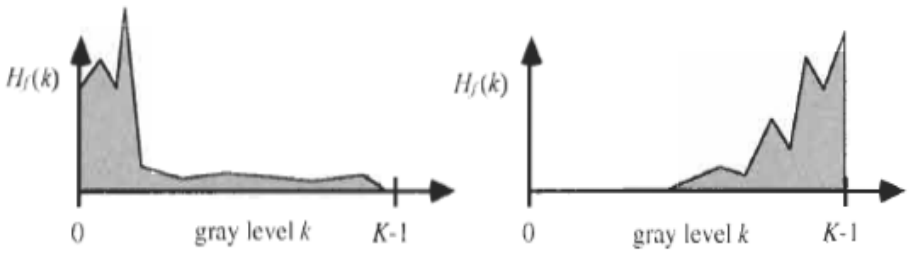
For example, the Average Optical Density (AOD) is the basic measure of an image's overall average brightness or grey level. It can be computed directly from the image as:

$$AOD\ f = \frac{1}{NM}\sum_{n_1=0}^{N-1}\sum_{m_1}^{M-1} f\ n_1, m_1 \qquad (2.24)$$

or it can be computed from the image histogram:

$$AOD\ f = \frac{1}{NM}\sum_{K=0}^{K-1} KH_f\ f \qquad (2.25)$$

The AOD is a useful and simple meter for assessing the centre of an image's grey-level distribution. A target value for the AOD might be specified when designing a point operation to change the overall grey-level distribution of an image. Figure 2.5 depicts two proposed image histograms. Figure 2.4(a) has a heavier distribution of grey levels close to zero andhas allowAOD**,** while Figure 2.4(b) is skewed toward the right with a high AOD. Since image grey levels are usually displayed with lower numbers' indicating darker pixels, the image of Figure 2.4(a) corresponds to a mainly dark image. This may occur if the image *f* was originally underexposed prior to digitization, or if it was taken under poor lighting levels or perhaps the process of digitization was performed improperly (Jerry, 2000). A skewed histogram often indicates a problem in grey-level allocation. The image of Figure 2.4(b) may have been overexposed or taken in very bright light.



(a) Left Skewed Distribution (b) Right Skewed Distribution

Figure2.4: Histograms of Grey Level Images

2.2.6.3 *Histogram Shaping*

The process of histogram shaping generalizes histogram equalization, which is the special case in which the target shape is flat. Histogram shaping can be applied when multiple images of the same scene, taken under mildly different lighting conditions, are to be compared; this extends the idea of AOD equalization (Jerry, 2000). When the histograms are shaped to match, the comparison may exclude minor lighting effects. Alternately, it may be that the histogram of one image is shaped to match that of another for the purpose of comparison. Or it might simply be that a certain histogram shape, such as a Gaussian form produces visually agreeable results for a certain class of images (Jerry, 2000). Histogram shaping is also accomplished by a nonlinear point operation defined in terms of the empirical image probabilities or histogram functions. Again, exact results are obtained in the hypothetical continuous-scale case. Suppose that the target (continuous) cumulative histogram function is Q(x), and that Q^{-1} exists (Jerry, 2000). Then let

$$g = Q^{-1}\left[P_f(f)\right] \tag{2.26}$$

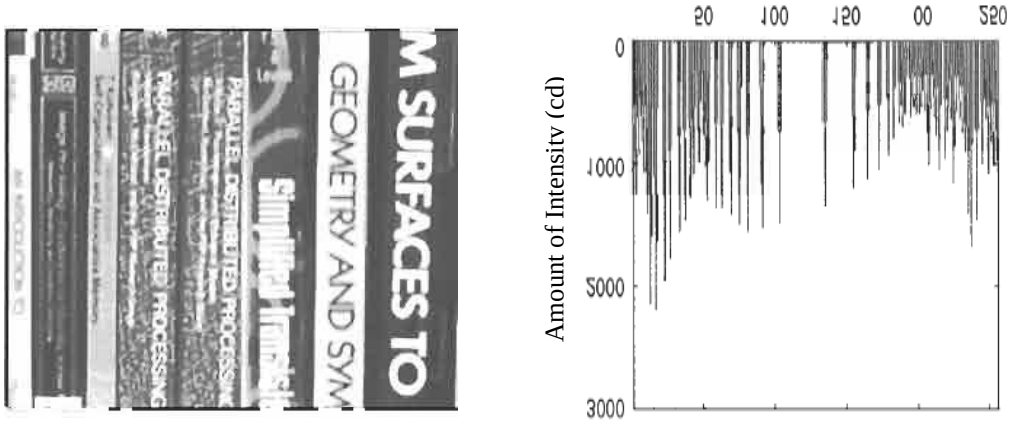
where both functions in the composition are applied on a pixel wise basis. The cumulative histogram of $\boldsymbol{g}$ is then,

$$\begin{aligned} P_g\ x &= \mathrm{P_r}\ g \le x \\ &= \mathrm{P_r}\ Q^{-1}\left[P_f\ f\ \right] \le x \\ &= \mathrm{P_r}\ P_f\ f\ \le Q\ x \\ &= \mathrm{P_r}\ f \le P_f^{-1}\left[Q\ x\ \right] \\ &= P_f\ P_f^{-1}\left[Q\ x\ \right] = Q\ x \end{aligned} \tag{2.27}$$

In such cases, the specified target cumulative histogram function Q(k) is discrete and some convention for defining Q^{-1} should be adopted, particularly if Q is computed from a target image and is unknown in advance (Jerry, 2000). One common convention is to define Q^{-1} as:

$$Q^{-1}\ k\ = \min\ s : Q\ s\ \geq k \qquad (2.28)$$

Asan example, Figure2.5represents the result of shaping the histogram of "books" to match the shape of an inverted "V" centred at the middle grey level and extending across the entire grey scale. Again, a perfect V is not produced, although an image of very high contrast is still produced. Instead, the histogram shape of the results is a crude estimate to the target (Jerry, 2000).



(a)Book Image (b) V-Shape Histogram

Figure 2.5: Image and its Histogram

2.2.7 Arithmetic Operations between Images

The types of operations that can be applied to digital images to transform an input image $a[m,n]$ into an output image $b[m,n]$ (or another representation) is classified into three categories (Young, 2007).

i. **Point**– the output value at a specific coordinate is dependent only on the input value at that same coordinate.

ii. **Local**– the output value at a specific coordinate is dependent on the input values in the neighbourhood of that same coordinate.

iii. **Global**– the output value at a specific coordinate is dependent on all the values in the input image (Young, 2007).

Consider arithmetic operations defined on multiple images. The basic operations are point-wise image addition, subtraction and point-wise image multiplication and division. Since digital images are defined as arrays of numbers, these operations have to be defined carefully (Jerry, 2000). Suppose n images of dimensions NxM_{f_1}, $f_1 ..... f_n.t$, it is important that they be of the same dimensions since operations between corresponding array elements are going to be defined (having the same indices) (Jerry, 2000). The sum of ***n*** images is given by:

$$f_1 + f_2 + ... + f_n = \sum_{m=1}^{n} f_m \qquad (2.29)$$

While for any two imagesf_r, $f_s$the image difference I_{diff} is:

$$I_{diff} = f_r - f_s \qquad (2.30)$$

The point-wise product of the ***n*** images f_1,..., f_n is denoted by:

$$f_1 \otimes f_2 \otimes ... \otimes f_n = \prod_{m=1}^{n} f_m \qquad (2.31)$$

The product is defined on a point-wise basis. Hence:

$$g = f_1 \otimes f_2 \otimes, ..., \otimes f_n \qquad (2.32)$$

if and only if for every n

$$g\ n = f_1\ n\ \ f_2\ n\ , ..., f_n\ n \qquad (2.33)$$

In order to clarify the distinction between matrix product and point-wise array product, a special notation $\otimes$ was introduced to denote the point-wise product (Jerry, 2000). Given two imagesf_r, f_s the point-wise image quotientis denoted as follows:

$$g = f_r \Delta f_s \qquad (2.34)$$

If for every nth is true that $f_s\ \ n\ \neq 0$ and

$$g\ \ n\ = f_r\ \ n \,/\, f_s\ \ n \qquad (2.35)$$

The point-wise matrix product and quotient are mainly useful when Fourier transforms of images are manipulated (Jerry, 2000).

2.2.8 Image Noise

Noise is a random variation of image Intensity and visible as grains in the image. It may arise in the image as effects of basic physics-like photon nature of light or thermal energy of heat inside the image sensors. It may produce at the time of capturing the image on the time line or image transmission. Noise means, the pixels in the image show different intensity values instead of true pixel values (Verma and Ali, 2013). Noise removal algorithm is the process of removing or reducing the noise from the image. The noise removal algorithms reduce or remove the visibility of noise by smoothing the entire image leaving areas near contrast boundaries in order to avoid affecting the intensity of theluminance. But these methods can obscure fine, low contrast details. Different noises have their own characteristics which make them distinguishable from others (Verma and Ali, 2013).

2.2.8.1 *Sources of Noise in Images*

Noise is introduced in the image at the time of image acquisition or transmission. Different factors may be responsible for the introduction of noise in the image. The number of pixels corrupted in the image will decide the quantification of the noise (Verma and Ali, 2013). The principal sources of noise in the digital image are:

i. The image sensor may be affected by environmental conditions during image acquisition.
ii. Insufficient Light levels and sensor temperature may introduce the noise in the image.
iii. Interference in the transmission channel may also corrupt the image.
iv. If dust particles are present on the scanner screen, they can also introduce noise in the image (Verma and Ali, 2013).

2.2.8.2 *Different Noise Types*

Noise is the undesirable effects produced in the image. During image acquisition or transmission, several factors are responsible for introducing noise in the image. Depending on the type of disturbance, the noise can affect the image to different extent. Image noise can be classified as Impulse noise (Salt-and-pepper noise), Amplifier noise (Gaussian noise), Shot noise (Poisson noise), Quantization noise (uniform noise), Film grain, on-isotropic noise, Multiplicative noise (Speckle noise) and Periodic noise (Verma and Ali, 2013).

Amplifier Noise (Gaussian Noise): The standard model of amplifier noise is additive, Gaussian, independent at each pixel and independent of the signal intensity. In colour cameras where more amplification is used in the blue colour channel than in the green or red channel, there can be more noise in the blue channel. Amplifier noise is a major part of the noise of an image sensor, that is, of the constant noise level in dark areas of the image (Patidar*et al.*, 2010). Dust particles in the image acquisition source or over heated faulty components can cause this type of noise. Image is corrupted to a small extent due to noise. This noise arises in the image because of sharp and sudden changes of image signal. Figure 2.6 shows the effect of noise on the original image.



(a) Image without Noise (b) Image with 30% Salt & PepperNoise

Figure 2.6: Original Image without and with Noise (Verma and Ali, 2013)

Figure 2.7 shows the effect of the addition of Gaussian noise onto images of Figure 2.6 with zero mean (Verma and Ali, 2013).

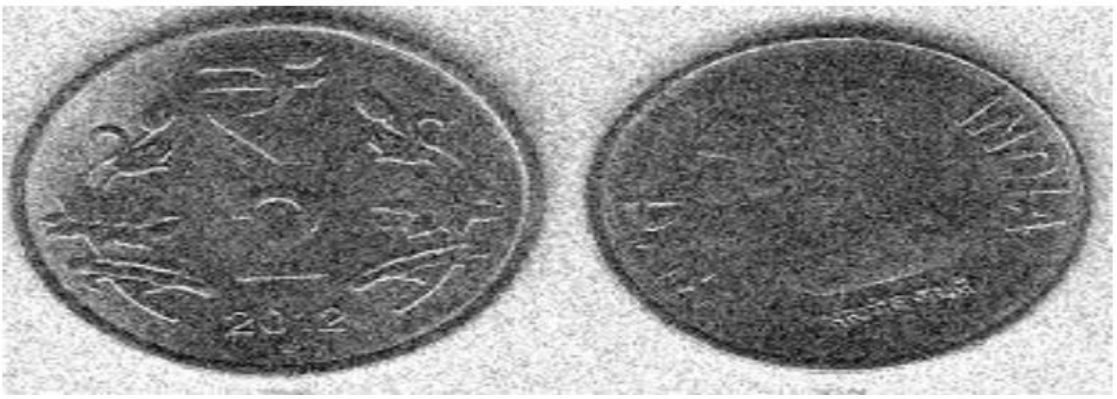
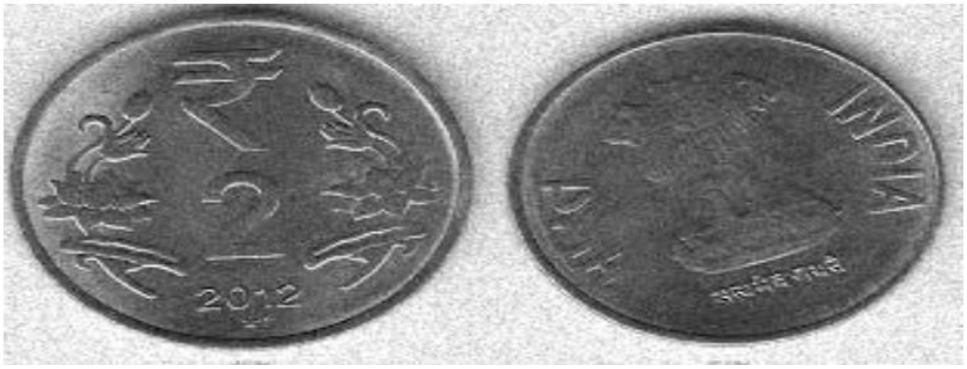


Figure 2.7: Effect of Gaussian Noise with Zero Mean on an Image (Verma and Ali, 2013)

Salt-and-Pepper Noise: An image noise will have dark pixels in bright regions and also the brightness contain salt-and-pepper pixels in dark regions(Patidar et al., 2010). This type of noise can be caused by dead pixels, analog-to-digital converter errors, bit errors in transmission, etc. This can be eliminated in large part by using dark frame subtraction and by interpolating around dark/bright pixels (Patidar at al., 2010).

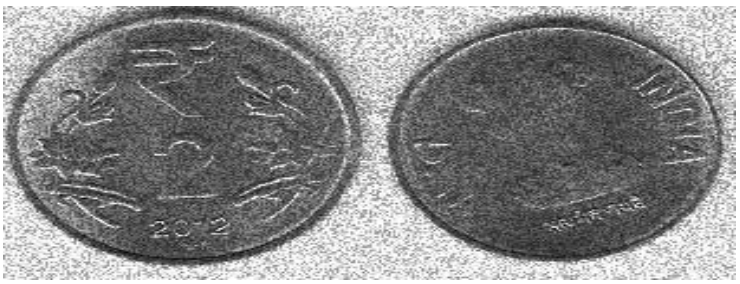
Poisson Noise: Poisson noise or shot noise is a type of electronic noise that occurs when the finite number of particles that carry energy, such as electrons in an electronic circuit or photons in an optical device, is small enough to give rise to detectable statistical fluctuations in a measurement (Patidar*at al.,* 2010).This noise has root mean square value proportional to square root intensity of the image. Different pixelsare affected by independent noise values. At practical grounds the photon noise and other sensor based noise corrupt the signal at different proportions (Verma and Ali, 2013). Figure 2.8 `shows the result of adding Poisson noise.



(a) Front of Coin (b) back of Coin

Figure 2.8: Image with Poisson Noise (Verma and Ali, 2013)

Speckle Noise: Speckle noise is a granular noise that inherently exists in and degrades the quality of the active radar and Synthetic Aperture Radar (SAR) images. Speckle noise in conventional radar results from random fluctuations in the return signal from an object that is not bigger than a single image-processing element. It increases the mean grey level of a local area. Speckle noise in SAR is generally more serious, causing difficulties for image interpretation. It is caused by coherent processing of backscattered signals from multiple distributed targets. In SAR oceanography, for example, speckle noise is caused by signals from elementary scatters, the gravity-capillary ripples, and manifests as a pedestal image, beneath the image of the sea waves (Patidar*at al.,* 2010). Figure 2.9 illustrates the effect of adding speckle noise.



(a)Front of Coin (b) back of Coin

Figure2.9: Image with Speckle Noise (Verma and Ali, 2013)

2.2.9 Image De-noising

Image de-noising is very important task in image processing for the analysis of images. Ample image de-noising algorithms are available, but the best one should remove the noise

completely from the image, while preserving the details (Verma and Ali, 2013). Filtering in an image processing is the basic function that is used to achieve many tasks such as noise reduction, interpolation, and re-sampling (Kamboj and Rani, 2013). Filtering image data is a standard process used in almost all image processing systems. The choice of filter is determined by the nature of the task performed by the filter and its behaviour and the type of the data (Kamboj and Rani, 2013).

De-noising methods can be linear as well as non-linear. Where linear methods are fast enough, they do not preserve the details of the images, whereas the non- linear methods preserve the details of the images (Verma and Ali, 2013) but they are slow. Image filtering is a technique employed to eliminate or reduce the effect of noise on an image. Filters can be described as follows:

i. **Filtering without Detection**: In this filtering technique there is a window mask which is moved across the observed image. When the mask starts moving from left top corner to the right bottom corner of the image, it performs some arithmetic operations without discriminating any pixel of image (Kamboj and Rani 2013).

ii. **Detection followed by Filtering**: This filtering mechanism involves two steps. In the first step it identifies the noisy pixels of an image and in second step it filters out those pixels off image leaving clean. In this filtering also there is a mask which is moved across the image. It performs some arithmetic operations to detect the noisy pixels of image. Then the filtering operation is performed only on those pixels of image which are found to be noisy in the first step, keeping the non-noisy pixel of image intact (Kamboj and Rani 2013).

iii. **Processes Combination Filtering**:In this type ofhybrid filtering scheme, two or more filters are used to filter a corrupted location of a noisy image(Kamboj and Rani 2013).

2.2.10Linear Filters and Non-Linear Filters

Linear filtering of a signal can be seen as a controlled scaling of signal components in the frequency domain. Reducing the components in the centre of the frequency domain that is low frequencies, gives high frequency components an increase relative importance, and thus high pass filtering is performed. Gaussian or averaging filters are suitable for this purpose. These filters also tend to blur the sharp edges, destroy the lines, and other fine details of the image.

In recent years, a variety of non-linear median type filters such as rank conditioned, weighted median, relaxed median, rank selection median have been developed to overcome the shortcoming of linear filters (Kamboj and Rani 2013) by preserving the image details after filtering.

2.2.10.1 *Different Type of Linear and Non-Linear Filters*

The mean linear filter is a simple spatial filter. It is a sliding-window filter that replaces the centre value in the window (Kamboj and Rani 2013). Also, an example of a non-linear filter is the median filter which isused for reducing the amount of intensity variation between onepixel and the other pixel.

Mean Filter: The mean filter is a simple spatial filter .It is a sliding-window filter that replaces the centre value in the window. It replaces with the average mean of all the pixel values in the kernel or window.(Kamboj and Rani, 2013). Here, the filter computes the average value of the corrupted image in a pre-decided area to achieve image smoothening. Then the centre pixel intensity value is replaced by that average value (Verma and Ali 2013) for compensation purpose. This process is repeated for all pixel values in the image.

Figure2.10 (a) and (b) show the effect of using mean filter of size 5x5 on different types of noise (Verma and Ali 2013).



(a) Salt & Pepper Noise Image (b) Poisson Noise Image

Figure2.10:Use of Mean Filter on Noise Images (Verma and Ali, 2013)

Some of the well-known advantages of the mean filter are easy implementation and its ability to remove impulse noise. Its main disadvantage is that itdoes not preserve the whole details of the image; some are removed during its usage (Kamboj and Rani, 2013)

Median Filter: Median filter is a simple and powerful non-linear filter which is based on statistical arrangement. It is easy to implement as the method of smoothing images. Median filter is used for reducing the amount of intensity variation between one pixel and the other pixel (Kamboj and Rani, 2013). In this filter, the pixel value of image is not replaced with the mean of all neighbouring pixel values; it is replaced with the median value. Then the median is calculated by first sorting all the pixel values into ascending order and then replace the pixel being calculated with the middle pixel value. If the neighbouring pixels of image which are to be considered, contain an even numbers of pixels, than the average of the two middle pixel values is used. The median filter gives best result when the impulse noise percentage is less than 0.1 %. When the quantity of impulse noise is increased the median filter does not gives the best result (Kamboj and Rani, 2013). Median filter is good for salt and pepper noise. These filters are widely used as smoothers for image processing, as well as in signal processing. A major advantage of the median filter over linear filters is that the median filter can eliminate the effect of input noise values with extremely large magnitudes (Verma and

Ali, 2013). Here the centre value of the pixel is replaced by the median of the pixel values under the noise. Figure 2.11 show the effect of median filter on different types of noise (Salt & Pepper and Gaussian)(Verma and Ali, 2013)



(a) Salt & Pepper Noise Image (b) Gaussian Noise Image

Figure 2.11:Use of Median Filter on Noise Images (Salt & Pepper and Gaussian) (Verma and Ali, 2013)

Figure 2.12 shows the effect of median filter on different types of noise (Poisson and Speckle).



(a) Poisson Noise Image (b) Speckle Noise Image

Figure2.12:Use of Median filter on Noise Images (Poisson and Speckle) (Verma and Ali, 2013)

2.2.11 Fundamental Steps in Image Processing

Below are some of the fundamental steps of image processing

i. **Image Acquisition:** The face image could be acquired through a high-resolution still digital camera and compressed to an image file (Frank, 2010).

ii. **Image Pre-processing:** The acquired image may be enhanced by improving contrast, sharpness, colour, and so on. (Frank, 2010).

iii. **Image Segmentation:** The image may first be cropped to only the facial area. Then, the face may be segmented into eyes, mouth, nose, chin, and so on (Frank, 2010).

iv. **Image Representation and Description:** In this step, each of the segmented areas may be characterized by statistical data, for example, principal components analysis, texture, aspect ratios of eyes and nose, or the colour of eyes (Frank, 2010).

v. **Matching Recognition and Interpretation:** This step may involve using the characteristics derived in the previous step to match each individually segmented area based on specific recognition algorithms. For example, eyes may be processed to determine, based on its features, what class of eye it is. Then, all of these interpretations are used to create a composite description of the "ensemble," perhaps in the form of a feature vector for the subject (Frank, 2010).

vi. **Knowledge Base:** Finally, the feature vector above may be fed to a knowledgebase of all known subjects to associate it with one of the subjects in the database, thus returning perhaps the individual's social security number or perhaps confidence score of the match (Frank, 2010).

Figure 2.13 shows the fundamental steps in digital image processing.

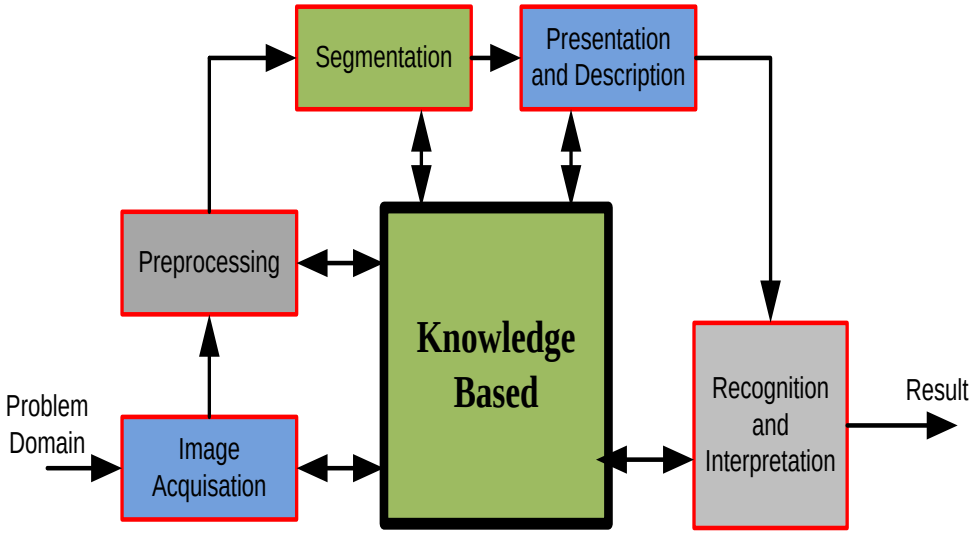


Figure 2.13: Fundamental Steps in Digital Image Processing (Chang, 2007)

2.2.12Methods of Noise Suppression

i. **Mean Filter Technique**: Mean filtering can be realized by convolving the image with a (2K + 1x2L + 1) kernel where each coefficient has a value equal to the reciprocal of the numbers of coefficients in the kernel (Isaac, 2000). Naturally, this type of smoothing reduces noise in the image, but at the expense of the sharpness of edges. The amount of noise suppression is related to the size of the kernel, with greater suppression realized by larger kernels (Isaac, 2000).

ii. **Median Filter Technique**: Median filtering is a common nonlinear method for noise suppression that has unique characteristics. It does not use convolution to process the image with a kernel of coefficients, but in each position of the kernel frame, a pixel of the input image contained in the frame is selected to become the output pixel situated at the coordinates of the kernel center (Isaac, 2000). The frame is centered on each pixel (m, n) of the original image, and the median value of the pixels within the kernel frame is computed. The pixel at the coordinates (m, n) of the output image is set to this median value. Generally, median filters do not have the same smoothing characteristics as the mean filter (Isaac, 2000). Features that are smaller than half the

size of the median filter kernel are completely removed by the filter. Large breaks such as edges and large changes in image intensity are not affected in terms of grey level intensity by the median filter, though their locations may be shifted by a few pixels. The nonlinear operation of the median filter allows significant reduction of specific types of noise. For example "short noise" may be removed completely from an image without reduction of significant edges or image characteristic. (Isaac, 2000)

iii. **Image Average Technique: Noise suppression using image averaging relies on three basic assumptions: (1) that a relatively large number of input images are available, (2) that each input image has been corrupted by the same type of additive noise, and (3) that the additive noise is random with zero mean value and independent of the image (Isaac, 2000). When these assumptions hold, it may be advantageous to acquire multiple images with the specific purpose of using image averaging since with this approach even severely corrupted images can be significantly enhanced and each of the noisy images a_i(m, n) can be represented by (Isaac, 2000):**

$$a_i(m,n) = f(m,n) + d_i(m,n) \tag{2.36}$$

where f(m, n) is the underlying noise free image, and d_i(m, n) is the additive noise in the image. If a total of Q images are available, the average image is(Isaac, 2000):

$$g(m,n) = \frac{1}{Q}\sum_{i=1}^{Q} a_i(m,n) \tag{2.37}$$

Such that

$$E(g, m,n) = f(m,n) \tag{2.38}$$

And

$$\sigma = \frac{\sigma_d}{\sqrt{Q}} \tag{2.39}$$

where E E . is the expected value operator, σ_g is the standard deviation of g (m, n), and σ_d is that of the noise.

Noise suppression is more effective for larger value of Q (Isaac, 2000). The amount of noise reduction can be quite significant, with expected reduction in the noise variance by a factor n**.** However, this is subject to inaccuracies in the model, for example, if there is any change in the scene itself, or if there are any dependencies between the noise images (in an extreme case, the noise images might be identical), then the reduction in the noise will be limited (Jerry, 2000).

2.2.13 Images Differences for Change Detection

Often it is of interest to detect changes that occur in images taken of the same scene but at different times. If the time instants are closely placed, for example, adjacent frames in a video sequence, then the goal of change detection amounts to image motion detection (Jerry, 2000). There are many applications of motion detection and analysis. For example, in video compression algorithms, compression performance is improved by exploiting redundancies that are tracked along the motion trajectories of image objects that are in motion. Detected motion is also useful for tracking targets, for recognizing objects by their motion, and for computing three-dimensional scene information from two-dimensional motion. If the time separation between frames is not small, then change detection can involve the discovery of gross scene changes (Jerry, 2000). This can be useful for security or surveillance cameras, or in automated visual inspection systems. In either case, the basic technique for change detection is the image difference. Suppose that f_1 and f_2 are images to be compared, then the absolute difference image will embody those changes or differences that have occurred betwe en the images.

$$g = |f_1 - f_2| \tag{2.40}$$

At coordinate n, where there has been little change, g (n) will be small. Where change has occurred, g(n) can be quite large. For the difference image, large changes are displayed as brighter intensity values. Since significant change has occurred, there are many bright intensity values (Jerry, 2000). This difference image could be processed by an automatic change detection algorithm. A simple series of steps that might be taken would be to use the binaries of the difference image, thus separating change from non-change, using a threshold counting the number of high-change pixels and finally deciding whether the change is significant enough to take some action (Jerry, 2000).

2.2.14 Frequency Domain Techniques

Linear filters used for enhancement can also be implemented in the frequency domain by modifying the Fourier transform of the original image and taking the inverse Fourier transform.

When an image g(m, n) is obtained by convolving an original image f(m, n) with a kernel

w (m, n),

$$g\ m,n\ = w\ m,n\ * f\ \ m,n \qquad (2.41)$$

the convolution theorem states that G(u, v), the Fourier transform g(u, v), is given by

$$G\ u,v\ = W\ u,v\ F\ u,v \qquad (2.42)$$

Where W (u, v) and F(u, v) are the Fourier transforms of the kernel and the image, respectively. Therefore, enhancement can be achieved directly in the frequency domain by multiplying F (u, v), pixel- by –pixel, by an appropriate W (u, v) and forming the enhanced image with the inverse Fourier transform of the product (Isaac, 2000). Noise suppression or image smoothing can be obtained by eliminating the high frequency components. Since the

spectral filtering process depends on a selection frequency parameters as high or low, each pair (u, v) is quantified with a measure of distance from the origin of the frequency plane,

$$D\ u,v\ = \sqrt{u^2 + v^2} \tag{2.43}$$

Which can be compared to a threshold D_T to determine if (u, v) is high or low (Isaac, 2000). The simplest approach to image smoothing is the ideal low- pass filter W_L(u, v), defined to be 1 when D (u, v) $\geq D_T$ and 0 otherwise. Similarly, the ideal high- pass filter W_H(u, v) can be defined to be 1 when D (u, v) $\geq D_T$ and 0 otherwise. However, these filters are not typically used in practice, because images that they produce generally have spurious structures that appear as intensity ripples, known as ringing (Isaac, 2000). The inverse Fourier transform of the rectangular window W_L(u, v) or W_H(u, v) has oscillations, and its convolution with the spatial- domain image produces the ringing.

Because ringing is associated with the abrupt 1 to 0 discontinuity of the ideal filters, a filter that imparts a smooth transition between the desired frequencies and the attenuated ones is used to avoid ringing (Isaac, 2000). The commonly used Butterworth low-pass and high- pass filters are defined respectively as

$$B_L\ u,v\ = \frac{1}{1+c\ D(\mathrm{u,v})/\mathrm{D_T}^{\ 2n}} \tag{2.44}$$

and

$$B_H\ u,v\ = \frac{1}{1+c\ \mathrm{D_T}/D(\mathrm{u,v})^{\ 2n}} \tag{2.45}$$

Where c is a coefficient that adjusts the position of the transition and n determines its steepness, if c = 1, these two functions take the value 0.5 when D (u, v) = D_T. Another

common choice for c is$\frac{1}{\sqrt{2}}-1$which yields 0.707 (-3 dB) at the cut-off D_T. The most common choice of n is 1; higher value yield steeper transitions (Isaac, 2000).

The threshold D_T is generally set up by considering the power of the image that will be contained in the preserved frequencies. The set S of frequency parameters (u, v) that belong to the preserved region, i.e., D (u, v) $\leq D_T$ for high- pass determines the amount of retained image power. The percentage of total power that the retained power constitutes is given by

$$\beta = \frac{\sum_{u,v \in s} \left|F(u,v)\right|^2}{\sum_{\forall u,v} \left|F(u,v)\right|^2} \times 100 \qquad (2.46)$$

and is used generally to guide the selection of the cut-off threshold. In Figure 2.14 a, circles with radii r_β thatcorrespond to five different β values are shown on the Fourier transform on an original MRI image in Figure 2.14 e. The u = v = 0 point of the transform is in the centre of the image in Figure 2.14 a. The Butterworth low-pass filter obtained by setting D_T equal to r_β for β = 90 %, with c = 1 and n = 1, is shown in Figure 2.14 b, where bright point indicate high values of the function (Isaac, 2000). The corresponding filtered image in Figure 2.14 f shows the effects of smoothing. A high-pass Butterworth filter with D_T set at the 95% level is shown in Figure 2.14 d, and its output in Figure 2.14 h highlights the highest frequency components that form 5% of the image power (Isaac, 2000). Figure 2.14 c shows a band – pass filter formed by the conjunction of a low- pass filter at 95% and a high-pass filter at 75%, while the output image of this band –pass filters in Figure 2.14 g.

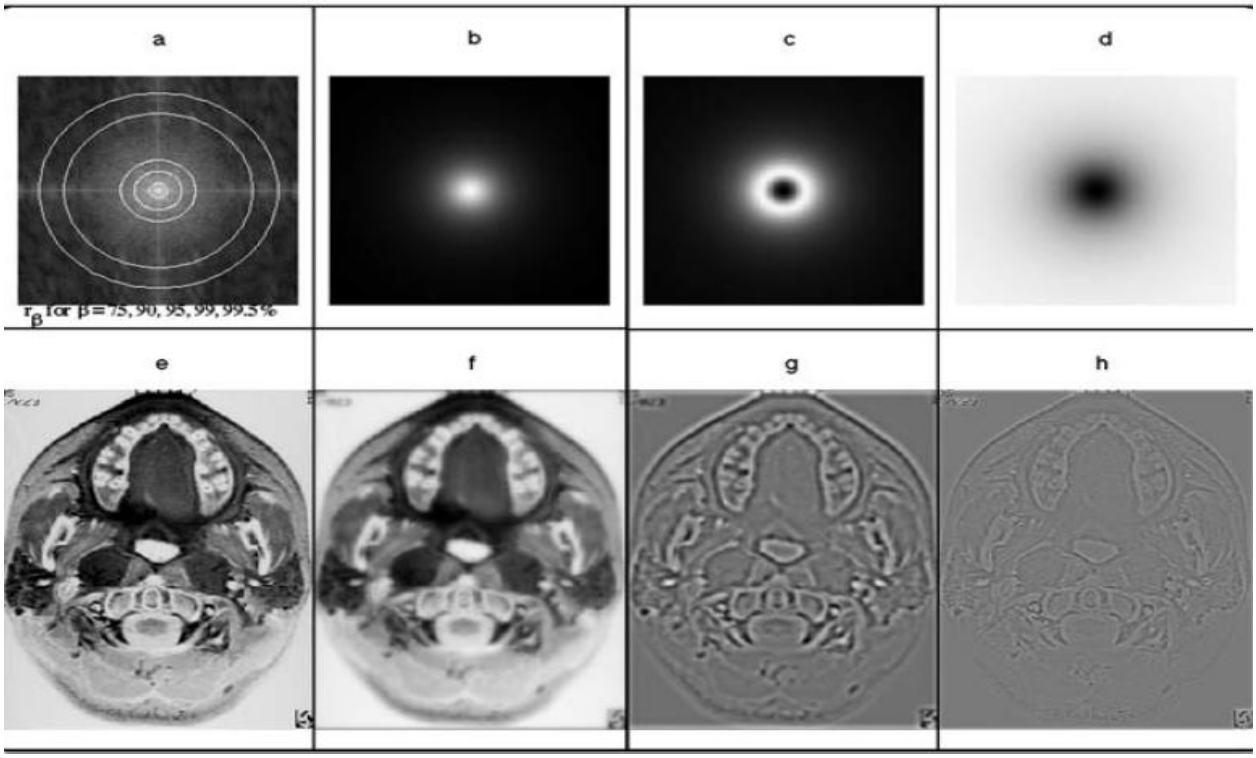


Figure 2.14: Filtering with the Butterworth Filter

Fourier transforms of MRI image in (e); the five circles correspond to the β values 75, 90, 95, 99, and 99.5%. (b) Fourier transform of low-pass with β = 90% which provides the output image in (f). (c) Band-pass filter with band β = 75% to β = 90% whose output is in (g). (d) High-pass filter with β = 95%, which yields the image in (h) (Isaac, 2000).

2.2.15 Video and Image Compression

Video or image compression is aimed at reducing the file size of an image or video data while degrading image quality as little as possible (Pascal *et al.*, 2015). Most lossy or lossless compression like the discrete cosine transform tries to eliminate the spatial redundancies. Most, compression technique are based on the principles of coding. Some of the coding techniques available are highlighted as follows (Catania, 2008):

i. Huffman coding;

ii. Arithmetic coding;

iii. Substitutional (Dictionary based) coding;

iv. Sample/based coding;

v. Transform Domain coding;

vi. Wavelet based coding

2.2.15.1 Huffman Coding

The probability model and symbol-to-code word are combined together to form a sequence of symbols. The sequences of the symbols are represented in the following order (Catania, 2008):

i. Ordering of the symbols according to their probabilities.

ii. Applying a contraction to the two symbols with the smaller probabilities.

iii. Repeating the previous step until the final set has only one member.

Construction of a binary tree involved the code word for each symbol which is obtained by traversing the binary tree from its root to the leaf corresponding to the symbol (Catania, 2008).

2.2.15.2 Arithmetic Coding

This process depends on the data type, and the blocks in Figure 2.15 is in different order or combined. Numerical processing, like predictive coding and linear transforms, were normally used for waveform signals, like images and audio. Logical processing which entails changing the data to a form a better compression, like run-lengths, zero-trees, set-partitioning information, and dictionary entries (Amir, 2004). The next stage, source modelling, is used to account for variations in the statistical properties of the data. It is responsible for gathering statisticaldata and identifying contexts that make the source models more accurate and reliable (Amir, 2004).

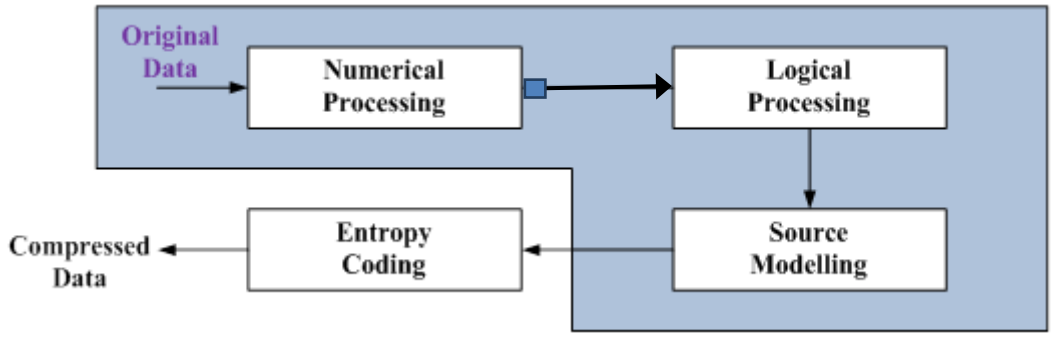


Figure 2.15: Typical Data Compression Processes (Amir, 2004).

Arithmetic coding is normally the final stage, and the other stages can be modelled as a single data source (Amir, 2004).

2.2.15.3*Substitution (Dictionary based) Coding*

Dictionary-based algorithms do not encode single symbols as variable-length bit strings. Instead, they encode variable-length strings of symbols as single tokens. The tokens form an index into a phrase dictionary. If the tokens are smaller than the phrases they replace the sequence, and compression occurs. Dictionary-based compression is easier to understand because it uses a strategy that programmers are familiar with. Implies that, using indexes data bases, information can be retrieved from large amounts of storage (Mark and Jean-loup, 2002).

2.2.15.5Sample/based Coding

There are four major types of probability sample designs: simple random sampling, stratified sampling, systematic sampling, and cluster sampling as shown in Figure 2.16. Simple random sampling is the most recognized probability sampling procedure. Stratified sampling offers significant improvement to simple random sampling. Systematic sampling is probably the easiest one to use because the cluster sampling is most practical for large national surveys (Mark and Jean-loup, 2002).

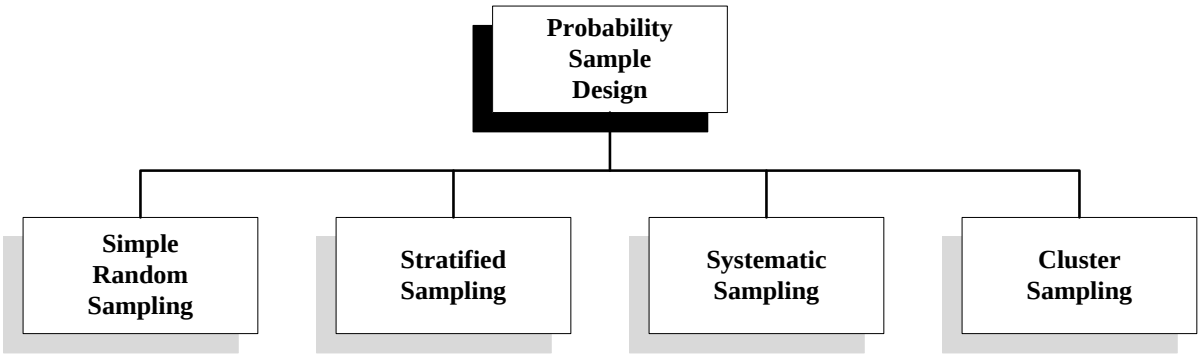


Figure 2.16: Major Types of Probability Sampling (Mark and Jean-loup, 2002)

2.2.15.5 Transform Domain Coding

The rationales behind transform coding are as follows (Li and Drew, 2003):

i. If Y is the result of a linear transform T of the input vector X in such a way that the components of Y are much less correlated, then Y can be coded more efficiently than X.

ii. If most information is accurately described by the first few components of a transformed vector, then the remaining components can be coarsely quantized, or even set to zero, with little signal distortion.

iii. The Discrete Cosine Transform (DCT) formalizes this notion with a measure of how much the image contents change in correspondence to the number of cycles of a cosine wave per block. The role of the DCT is to decompose the original signal into its DC and AC components; the role of the IDCT is to reconstruct (re-compose) the signal.

iv. The spatial frequency indicates how many times pixel values change across an image block.

The coding of DCT is shown in Figure 2.17.

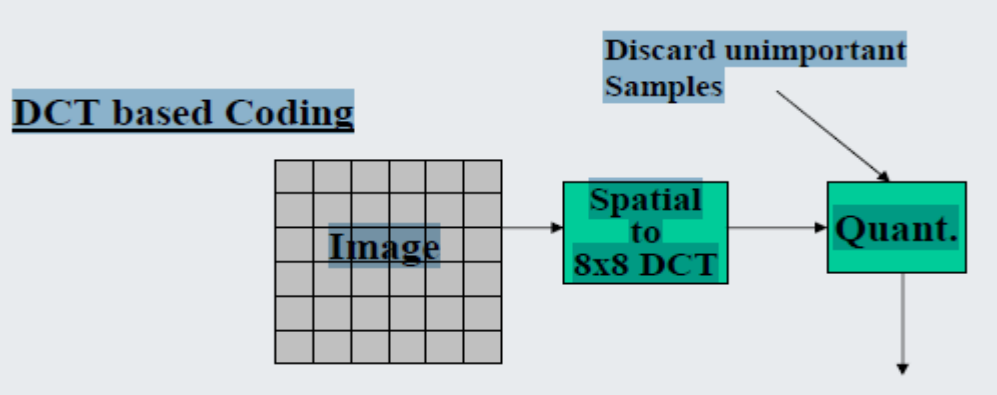


Figure 2.17: Discrete Cosine based Coding (Catania, 2008)

2.2.15.6 *Wavelet-based Coding*

The objective of the wavelet transform is to decompose the input signal into components that are easier to deal with, have special interpretations, or have some components that can be thresholded away for compression purposes (Li and Drew, 2003). The aim here is to approximately reconstruct the original signal contained in these components. The basic functions of the wavelet transform are localized in both time and frequency domains. There are two types of wavelet transforms:

i. Continuous wavelets transform (CWT) (it is popular among physicists
ii. Discrete Wavelets Transform (DWT)(is more common in numerical analysis, signal- and image-processing).

The block of diagram of 1D Dyadic Wavelet Transform is shown in Figure 2.18.

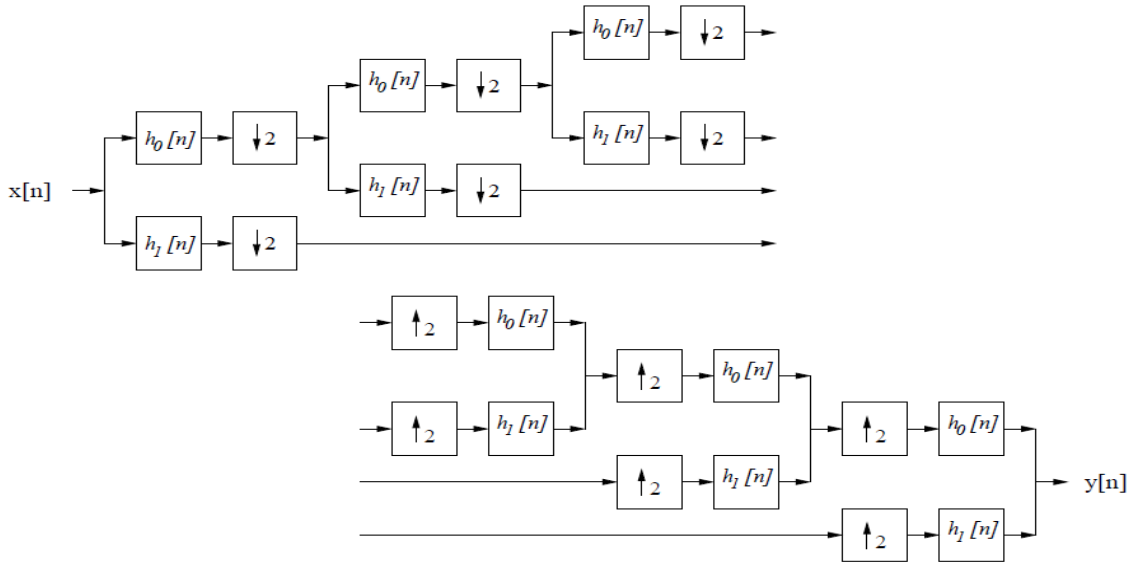


Figure 2.18: Block Diagram of the 1D Dyadic Wavelet Transform (Li and Drew, 2003)

In the usual dyadic wavelet decomposition, only the low-pass filtered sub-band is recursively decomposed and thus can be represented by a logarithmic tree structure (Li and Drew, 2003). Wavelet packet decomposition allows the decomposition to be represented by any pruned sub tree of the full tree topology. The wavelet packet decomposition is very flexible since a best wavelet basis in the sense of some cost metric can be found within a large library of permissible bases. The computational requirement for wavelet packet decomposition is relatively low asdecompositioniscomputed in the order of $N \log N$ using faster bands (Li and Drew, 2003).

As stated earlier, the research implemented threecompression techniques (Enhanced Discrete Cosine Transform, Enhanced Discrete Wavelet Transform and Enhanced Lifting Wavelet Transform) and the developed enhancement technique is applied to enhance the output of each compression by improving the luminance intensity of the output.

The transform coding techniques used a reversible, linear mathematical transform to map the pixel values onto a set of coefficients, which are then quantized and encoded. Transform coding relies on the premise that pixels in an image exhibit a certain level of correlation with

their neighbouring pixels (Preet and Geetu, 2012). Consequently, these correlations can be exploited to predict the value of a pixel from its respective neighbours. Different mathematical transforms, such as DCT, DWT and LWT have been considered for the task and their description is as follows (Preet and Geetu, 2012).

2.2.16 Discrete Cosine Transform

Image compression is considered just as the joint photographic expert group (JPEG) standard, which was designed to achieve the spatial and colour redundancy that existed in a single still image. Adjacent pixels in an image are often highly similar, and natural images often have most of their energies concentrated in the low frequencies (John, *et al* 2002). JPEG achieved these features by partitioning an image into 8x8 pixel blocks and computing the 2-D Discrete Cosine Transform (DCT) for each block. The drive for splitting an image into small blocks is that the pixels within a small block are generally more similar to each other than the pixels within a larger block. The DCT compacts most of the signal energy in the block into only a small fraction of the DCT coefficients, where this small fraction of the coefficients are sufficient to reconstruct an accurate version of the image (John, *et al* 2002). Each 8x8 block of DCT coefficients is then quantized and processed using a number of techniques known as zigzag scanning, run length coding, and Huffman coding to produce a compressed bit stream (Wallace, 1991). In the case of a colour image, a colour space conversion is first applied to convert the RGB image into luminance and chrominance colour space where the different human visual perception for the luminance (intensity) and chrominance characteristics of the image can be better exploited (Wallace, 1991).A video sequence consists of a sequence of video frames or images. Each frame may be coded as a separate image, for example by independently applying JPEG-like coding to each frame. However, since neighbouring video frames are typically very similar much higher compression can be achieved by exploiting the similarity between frames (Djordje, 2006). Currently, the most effective approach to exploit

the similarity between frames is by coding a given frame by first predicting it based on a previously coded frame, and then coding the error in this prediction. Consecutive video frames typically contain the same imagery, however possibly at different spatial locations because of motion (Djordje, 2006). Therefore, to increase the predictability it is important to evaluate the motion between the frames and then to form an appropriate prediction that compensates for the motion. The process of estimating the motion between frames is known as motion estimation (ME), and the process of forming a prediction while compensating for the relative motion between two frames is referred to as motion-compensated prediction (MC-P). Block based ME and MC-prediction are currently the most popular form of ME and MC-prediction: the current frame to be coded is partitioned into 16x16-pixel blocks, and for each block a prediction is formed by finding the best matching block in the previously coded reference frame (Djordje, 2006). The relative motion for the best-matching block is referred to as the motion vector. Three basic common types of coded frames are in existence: intra-coded frames, or I-frames, where the frames are coded independently of all other frames, predictively coded, or P-frames, where the frame is coded based on a previously coded frame, and bi-directionally predicted frames, or B frames, where the frame is coded using both previous and future coded frames (Djordje, 2006).

Most video and image compression algorithms apply the discrete cosine transform (DCT) to transform an image to frequency domain and achieve quantization for data compression. One of the advantages of the DCT is its excellent energy compaction property; that is, the signal energy is concentrated on a few components while most other components are zero or negligibly small. This helps separate an image into parts (or spectral sub bands) of hierarchical importance (with respect to the image's visual quality). A well-known JPEG compression technology uses the DCT to compress an image (Frank, 2010). A discrete cosine transform (DCT) expresses a sequence of finitely many data points in terms of a sum of

cosine functions oscillating at different frequencies. DCTs are important to numerous applications in science and engineering, from lossy compression of images (where small high-frequency components can be discarded) (Andrew, 1994). Discrete Cosine Transform is a technique for converting a signal into elementary frequency components, widely used in image compression. The rapid growth of digital imaging applications, including desktop publishing, multimedia, teleconferencing, and high-definition television (HDTV) has increased the need for effective and standardized image compression techniques (Gupta and Garg, 2012). Among the emerging standards are JPEG, for compression of still images, MPEG, for compression of motion video and CCITT H.261 (also known as Px64), for compression of video telephony and teleconferencing (Gupta and Garg, 2012). According to the DCT properties, a DC is transformed to discrete delta-function at zero frequency. Hence, the transform image contains only the DC component. To transform an image into 8 x 8 subsets by applying DCT in 2 dimensions. Also, a subset of DCT co-efficient has been prepared in order to perform inverse DCT to get the reconstructed image. The DCT exploits inter pixel redundancies to render excellent de-correlation for most natural images. Thus, all (uncorrelated) transform coefficients can be encoded independently without compromising coding efficiency. In addition, the DCT packs energy in the low frequency regions. Therefore, some of the high frequency content can be discarded without significant quality degradation. Such a (course) quantization scheme causes further reduction in the entropy (or average number of bits per pixel). Lastly, it is concluded that successive frames in a video transmission exhibit high temporal correlation (mutual information). This correlation can be employed to improve coding efficiency (Gupta and Garg, 2012). The work to be done is to perform the inverse transform of the transformed image and also to generate the error image in order to give the results in terms of MSE (Mean Square Error), as MSE increases, the

image quality degrades and as the MSE would decrease, image quality would be enhanced with the help of changing the co-efficient for DCT Blocks (Gupta and Garg, 2012).

2.2.17 Discrete Wavelet Transform

Industrial standards for compressing still images (e.g., JPEG) and motion pictures (For example, MPEG) have been based on the DCT. Both standards have produced good results, but have limitations at high compression ratios (Frank, 2010). At low data rates, the DCT based transforms suffer from a "blocking effect" due to the unnatural block partition that is required in the computation. Other drawbacks include mosquito noise (i.e., a distortion that appears as random aliasing occurs close to object's edges) and aliasing distortions (Frank, 2010). Furthermore, the DCT does not improve the performance as well as the complexities of motion compensation and estimation in video coding. Due to the shortcomings of DCT, discrete wavelet transform (DWT) has become increasingly important. The main advantage of DWT is that it provides space–frequency decomposition of images, overcoming the DCT and Fourier transform that only provide frequency decomposition (Frank, 2010). By providing space–frequency decomposition, the DWT allows energy compaction at the low-frequency sub bands and the space localization of edges at the high-frequency sub bands. Furthermore, the DWT does not present a blocking effect at the low data rates (Frank, 2010).

A wavelet is waveform of limited duration that has an average value of zero. Wavelets are localized waves and they extend not from -∞ to +∞ but only for finite time duration (Amara, 1995). The basis of Discrete Cosine Transform (DCT) is cosine functions while the basis of Discrete Wavelet Transform (DWT) is wavelet function that satisfies requirement of multi-resolution analysis (Christian *et al.,* 2009). DWT represents image on different resolution levels i.e., it possesses the property of Multi-resolution. DWT Converts an input image coefficients series x_0, x_1, x_m into one high-pass wavelet coefficient series and one low-pass

wavelet coefficient series. In practice, such transformation will be applied recursively on the low-pass series until the desired number of iterations is reached (of length n/2 each) (Priyanka, *et al.,* 2011).

2.2.18 Lifting Wavelet Transform

The lifting scheme (LS) has been introduced for the efficient computation of DWT. For image compression, it is very necessary that the selection of transform should reduce the size of the resultant data as compared to the original data set. So this new lossless image compression method is used. Wavelet using the lifting scheme significantly reduces the computation time and speeds up the computation process. The lifting transform even at its highest level is very simple. It performsthree operations: split, predict, and update (Chesta, *et al.,* 2011).

The image processing applications have the following key properties(Syed, 2003):

(i) **De-Correlation**: The principle advantage of image transformation is the removal of redundancy between neighboring pixels. This leads to uncorrelated transform coefficients which can be encoded independently

(ii) **Energy Compaction**: The efficiency of a transformation scheme can be directly gauged by its ability to pack input data into as few coefficients as possible. This allows the quantize to discard coefficients with relatively small amplitudes without introducing visual distortion in the reconstructed image.

(iii) **Orthogonally**: The basic functions of these transforms are orthogonal. Thus, the inverse transformation matrix of A is equal to its transpose,that is, $invA = AT$ (Priyanka, *et al.*, 2011).

2.2.19Purpose of Image and Video Processing

The following are some of the reasons why image and video are processed:

i. Improvement of pictorial information for human interpretation.

ii. Compression of image and video data for storage and transmission.

iii. Pre-processing to enable object detection, classification, and tracking (Yao Wang, 2016).

2.2.20 Application of Image Processing

In today modern engineering and technology world, image processing has been extensively applied in virtually all areas of possible industrial application. Some of these areas are listed as follows (Yao Wang, 2016):

i. Television Signal Processing.

ii. Satellite Image Processing.

iii. Medical Image Processing.

iv. Robotics.

v. Visual Communications.

vi. Law Enforcement.

2.2.21 Firefly Optimization Algorithm

The Firefly algorithm (FA) is an optimization tool which was inspired by the flash pattern and characteristic behaviour of fireflies. The (FOA) is a meta-heuristic, nature-inspired, optimization algorithm which is based on the social (flashing) behaviour of fireflies. The flashing light behaviour in natural fire flies can be associated with an objective function (minimization or maximization) of an optimization problem. This prompted Yang,(2010) to formulate new optimization algorithm of FOA based on the behaviours of natural fireflies. The basic idealized principles or rules governing the behaviours of FA are highlighted as follows(Yang, 2010):

1. “All fireflies are unisex so that one firefly will be attracted to other fireflies regardless of their sex.”
2. “Attractiveness is proportional to their brightness, thus for any two flashing fireflies, the less bright one will move towards the brighter one. The attractiveness is proportional to the brightness and they both decrease as their distance increases. If there is no brighter one then a particular firefly moves randomly,”
3. “Brightness of a firefly is affected or determined by the landscape of the objective function.”

The following are the pseudo code implementation of the standard FOA:

```
%Start
Initialize Parameters
Defined the objective function f(x). x=(x1,...,d)
Randomly generate initial population xi(i=1,2,...,n)
Evaluate light intensity Ii at xi is determined by f(xi)
Defined light absorption coefficient
while iteration
    for i=1:n
        for j=1:n
            if Ii<Ij
                move firefly i towards j
            end
            vary attractiveness with distance r
            Evaluate new solution and update light intensity
        end%for j
    end%for i
    Rank the fireflies and find the current global best
end%while
Evaluate result
```

For a maximization problem, the brightness can simply be proportional to the value of the objective function. Other forms of brightness can be defined in a similar way to the fitness function of other met heuristic algorithm such as, Genetic Algorithm, Particle Swarm Optimization, Bacterial Foraging Algorithm etc. In the simplest case of maximization problems, the brightness ‘I’of a firefly at a particular location ‘x’ can be chosen as $I(x) \propto f(x)$. However, the attractiveness ‘β’ is relative, so it should be seen in the eyes of

the beholder or judged by the other fireflies. Thus, it will vary with the distance 'r_{ij}between firefly 'i' and firefly '*j*'.

In addition, light intensity decreases with the distance from its source, and light is also absorbed in the media, so the attractiveness is allowed to vary with the degree of absorption. In the simplest form, the light intensity *'I'* varies according to the inverse square law (Yang, 2010):

$$I = \frac{I_s}{r^2} \tag{2.47}$$

Where;

I_s,is the intensity at the source, and

r,is the distance between two fireflies

For a given medium with a fixed light absorption coefficient γ, the light intensity *'I'* varies with the distance 'r'. That is:

$$I = I_o e^{-\gamma r} \tag{2.48}$$

Where;

I_o is original light intensity,

r is the distance between two fireflies and,

In order to avoid singularity at r = 0 in equation (2.47), the combined effect of both the inverse square law and absorption coefficient can be approximated as the follows (Yang, 2010):

$$I(r) = I_o e^{-\gamma r^2} \tag{2.49}$$

As the firefly's attractiveness is proportional to the light intensity seen by adjacent fireflies, we can now define the attractiveness *'* of a firefly by:

$$\beta = \beta_o e^{-\gamma r^2} \tag{2.50}$$

Where;

β_o is the attractiveness at r equals zero (0).

As it is often faster to calculate $\frac{1}{(1+r^2)}$ than an exponential function, the above function, if necessary, can conveniently be approximated as:

$$\beta = \frac{\beta_o}{1+\gamma r^2} \tag{2.51}$$

Equations (2.50) and (2.51) define a characteristic distance $\Gamma = \frac{1}{\sqrt{\gamma}}$ over which the attractiveness changes significantly for β to $\beta_o e^{-1}$ for equation (2.50) or $\frac{\beta_o}{2}$ for equation (2.51).

In the actual implementation, the attractiveness of β(r) can be any monotonically decreasing function such as the following generalised form:

$$\beta(r) = \beta_o e^{-\gamma r^m}, \quad m \geq 1 \tag{2.52}$$

For a fixed γ, the characteristic length becomes:

$$\Gamma = \gamma^{-\frac{1}{m}} \to 1, m \to \infty \tag{2.53}$$

Conversely, for a given length scale 'Γ' in an optimization problem; the parameter 'γ' can be used as a typical initial value. That is:

$$\gamma = \frac{1}{\Gamma^m} \tag{2.54}$$

The distance between any two fireflies 'i' and 'j' at 'x_i' and 'x_j' respectively is the Cartesian distance:

$$r_{ij} = \left\| x_i - x_j \right\| = \sqrt{\sum_{k=1}^{d} (x_{i,k} - x_{j,k})^2} \tag{2.55}$$

Where;

$x_{i,k}$ is the k^{th} component of the spatial coordinate x_i of the i^{th} firefly.

In 2D case, we have:

$$r_{ij} = \sqrt{(x_i - x_j)^2 + (y_i - y_j)^2} \tag{2.56}$$

The movement of firefly 'i' is attracted to another more attractive (brighter) firefly 'j' is determined by:

$$x_i^* = x_i + \beta_o e^{-\gamma r_{ij}^2} (x_j - x_i) + \alpha \in_i \tag{2.57}$$

Where;

$\beta_o e^{-\gamma r_{ij}^2} (x_j - x_i)$ is due to attractiveness,

$\alpha \in_i$ is a randomization parameter

α is the randomization parameter, and

$\in_i$ is a vector of random numbers drawn from a Gaussian or uniform distribution.

It is worth pointing out that equation (2.55) is a random walk biased towards the brighter fireflies. If β_o=0, it becomes a simple random walk (Yang, 2010). Based on the described optimization behaviour of the FOA, this research proposed an image compression technique.

Even though research has reported the efficient performance of FOA on several engineering problems, several challenges of FOA have also been reported. For example, Firefly algorithm

is powerful in local search but sometimes it may get trap into several local optimums as a result it cannot search global solutions efficiently. In order to address these challenges, researchers have proposed a modified and hybrid firefly algorithm with an improved diversification, better precision and convergence capability. Therefore, this research adopts the standard FOA algorithm and proposed a modified FOA capable of addressing the problem of imbalance between exploration and exploitation for video compression.

2.2.22 Performance Metric

The quality of a compression method with respect to the Discrete Cosine Transform (DCT), Discrete Wavelet Transform (DWT) and the Lifting Wavelet Transform (LWT) could be measured by the traditional distortion measures such as the Peak Signal to-Noise Ratio (PSNR), Signal to Noise Ratio (SNR), Mean Square Error (MSE) and Execution Time (ET). The phrase Mean Square Error, often abbreviated MSE (also called PSNR, Peak Signal to Noise Ratio) is an engineering term for the ratio between the maximum possible power of a signal and the power of corrupting noise that affects the fidelity of its representation (Gupta and Garg, 2012). Because many signals have a very wide dynamic range, MSE is usually expressed in terms of the logarithmic decibel scale. The MSE / PSNR is most commonly used as a measure of quality of reconstruction of lossy compression codecs (e.g., for image compression). The signal in this case is the original data, and the noise is the error introduced by compression (Gupta and Garg, 2012). When comparing compression codecs it is used as an approximation to human perception of reconstruction quality, therefore in some cases one reconstruction may appear to be closer to the original than another, even though it has a lower PSNR (a higher PSNR would normally indicate that the reconstruction is of higher quality) (Gupta and Garg, 2012). It is most easily defined via the mean squared error (MSE) which for

two $m \times n$ monochrome images I and K where one of the images is considered a noisy approximation of the other is defined as:

$$MSE = \frac{1}{mn} \sum_{i=0}^{m-1} \sum_{j=0}^{n-2} \left[I\ i, j\ - K\ i, j\ \right]^2 \quad (2.58)$$

The PSNR is defined as:

$$PSNR = 10 \log_{10} \left(\frac{MAX_1^2}{MSE} \right) = 20 \log_{10} \left(\frac{MAX_1^2}{\sqrt{MSE}} \right) \quad (2.59)$$

Where, MAX1 is the maximum possible pixel value of the image (Gupta and Garg, 2012). When the pixels are represented using 8 bits per sample, this is 255. More generally, when samples are represented using linear PCM with B bits per sample, MAX1 is $2^B - 1$. For colour images with three RGB values per pixel, the definition of PSNR is the same except the MSE is the sum over all squared value differences divided by image size and by three (Gupta and Garg, 2012).

2.3 Review of Similar Works

In order to have adequate knowledge of what is obtainable presently in the image and video signals research work, critical review of some of the related research predictions in this area are presented as follows:

2.3.1 Review of Similar Works on Image Processing

Li *et al.*, (2010) presented a new Distance Regularized Level Set Evolution (DRLSE). This active contour model in DRLSE formulation allowed the use of relatively large time steps to significantly reduce iteration numbers and computation time.It maintained sufficient numerical accuracy in both full domain and narrow band implementations due to the intrinsic distance regularization embedded in the level set evolution. However, edge based active

contour method is not considered the best, as the edges are not sharp due to fading of link or degradation which prevents the gradient value at the edges from being a high value.

Dhanasekar and Ramamoorthy, (2010) came up with restoration of blurred images for surface roughness evaluation using machine vision. The surface roughness of uniformly moving machined surface (grinding, milling) using machine vision technique was evaluated. However, in the case of moving surfaces the images are likely to blur due to the relative motion between the CCD camera and the object to be captured, which leads to the unwanted degradation of the image.

Fan *et al.*, (2010)proposed a new method for image contrast enrichment which was suitable for multiple-peak images. In this study, test images with poor contrast are used toverify the practicability and the stability of an automatic adaptive segmentationhistogram enhancement. Also, the comparisons for some of the previous studies wereprovided to demonstrate the robustness, visual quality, and effectiveness of the proposed algorithm.The research performed better than others on the aspects of simplicity and adaptability. However, the research work is only peculiar to image contrast enhancement only, but failed to address resolution and colour related issues.

Vasicek and Sekanina, (2010) applied Evolutionary Algorithms (EA) in image filtering. The filter was designed for parameter tuning to improve performance of the existing filters and designing a new structure filter by EA to achieve an optimal filter circuit. Their developed filter possessed noise cancellation technique and blurriness of imagesfeatures. Noise cancellation was performed only on the noise candidates and noise free pixels were not changed. Thus, more image edge detail was preserved and computation effort was reduced. However, the research work was influenced by errors such as roughness in the image quality and this could have resulted to poor resolution in visual image.

Kwok and Ha (2010)designed a strategy for local sector improvement by histogram equalization. Intermediate images were provoked recursively by making use of this approach and a resultant image obtained by a weighted-sum aggregation on the basis of an intensity gradient measure. Local sectors with higher contrast dominated, thus achieving overall global contrast improvement. An enhanced image was then produced where the intermediate images were repeatedly averaged using a weight-sum strategy. However, contrast is not the only component that constitutes a video signal enhancement, others includehue, saturation, and resolution of the sampled videothat were not considered and this made the improvement incomplete.

Boon *et al.*, (2011) proposed a single frame super-resolution algorithm which was aimed at increasing the resolution of mobile broadcast video, effectively forming a high quality conduit between mobile devices and a high-resolution displayed for broadcast video applications. Their results on video sequences showed that the algorithm could render high-resolution frames with good subjective quality, especially around image edges. The algorithm also provided significant objective Peak Signal to Noise Ratio (PSNR) improvement around such regions. However, in the case of moving surface the image are likely to blur due to the relative motion between the CCD camera and the object to be captured, hence the degraded image had to be restored by removing distortion, which is not an easy process.

Mathew and Shiba (2011)used a wavelet based technique for super resolution image reconstruction. This technique identified local features of low resolution image and then enhanced its resolution appropriately, thus overcoming the blurring problem that existed in interpolation techniques. The result of the research work showed that the technique had taken care of the problems of blur and checker board effect; the edges are well preserved and the originality of images were also preserved. However, a closely related problem observed was that of image restoration, which utilized a prior knowledge of the scene to recover missing

detail from a single image. Also, wavelet based techniques suffered from high frequency oscillation and low frequency oscillation which are associated with the energy content of the image.

Shechtman *et al.*, (2011)modelled a high-resolution image as Markov random field and used maximum posterior estimate as the final solution using graph cut optimization technique . The technique has an issue because the geometric formulation method is more meaningful than the labelling interpretation commonly proposed by graph-cut optimization technique. Also, the researchers focus was only limited to high resolution images. Furthermore, the geometric formulation method is accompanied by excess luminance whose end result is noise. These were some the limitations of this work.

Lu and Zhang, (2011) presented a video coding scheme which applied the technique of visual saliency computation to adjust image fidelity before compression. The algorithm effectively preserved visually important information while gradually degrading the unimportant regions. Instead of partitioning the whole picture into macro blocks, the algorithm was applied in a pixel-based process. It did not need any segmentation, which was often difficult to do in practice. Since the bilateral filter changes gradually, transitions between areas of different fidelity are smooth and region boundaries remain sharp. However, Colour space transformation between YUV and CIE-Lab was not lossless, and might have causedcolour distortion in some video frames and bi-lateral filter could not be used for removing mixed noise.Furthermore, bi-lateral filter in its form can introduce several types of image artefacts (noise).

Li *et al.*, (2011)made use of a variation level set framework for segmentation and bias correction of images with intensity in homogeneities. The research work was much more robust to initialization than the piecewise smooth model. Experimental results of this method

demonstrated superior performance in terms of accuracy, efficiency, and robustness. As an application, the method had been applied to Magnetic Resonance (MR) image segmentation and bias correction with promising result. However, during image segmentations, not all images were recovered. Also, the image components generated at high frequencies are considered to be noise.

Siddavatam *et al.*, (2011) employed a much improved impulse noise removal algorithm based on the lifting filter that gave acceptable results for image restoration even at 95% degradation by noise. This algorithm also worked well for binary images corrupted with impulse noise. The proposed algorithm yielded better results at 10%, 20%, 50% and 80% noise densities. The employed algorithm restored the image with only 4 iterations. The increment in the PSNR values with the median filters quantified the improvement in the algorithm. It was also proven that the algorithm computationally took very less time in comparison to other methods. However, median filtering performance is not satisfactory in case of signal dependent noise and the researchers failed to go beyond 4 alterations to see if a much better result could be achieved.

Rajib *et al.*, (2012)suggested an enhancement technique using scaling of internal noise of a dark image in discrete cosine transform domain. The mechanism of improvement was attributed to noise-induced transition of discrete cosine transform coefficients from a poor state to an improved state. The given approach had adopted a limited adaptive processing and significantly improved the image contrast and colour information while ascertaining good perceptual quality. When compared with the existing improvement approaches such as adaptive histogram equalization, gamma correction, etc, the given approach had showed extraordinary performance in terms of relative contrast enhancement, colourfulness, and visual quality of improved image. However, the researchers weresilent on other key components such as visual quality and resolution of the image.

Anila and Devarajan, (2012) presented the pre-processing method of solving one of the common problems in face images due to a real capture system of lighting variations. The different stages of this image processing included gamma correction, Difference of Gaussian (DoG) filtering and contrast equalization. But gamma correction changes the overall brightness of the image in RGB because the lighter the image when gamma level is raised and the darker the image when it is decreased, which does not allow the achievement good processed image.

Liu, (2012) presented a prediction error pre-processor based on the Just Noticeable Distortion (JND) for the colour image compression scheme. In this research work a prediction error pre-processor was presented with the goal of reducing the dynamic range of the prediction error signals of the colour image to be compressed. The pre-processor was further applied to the input colour image of the JPEG and JPEG2000 coder for better performance. Their model showedloss in terms of PSNR when JND driven by Rate Distortion Optimized Quantization (RDOQ) as a solution was used.

The researchers,**Garg and Kumar, (2012)** reviewed the Bayesian estimation process for statistical signal processing. Different noise models including additive and multiplicative types were used. They included Gaussian noise, salt and pepper noise, speckle noise, and Poisson noise. The results showed that Bayesian estimator optimized the Poisson noise removal as its Signal to Noise Ratio (SNR) was maximum and least Mean Square Error (MSE) minimum. The SNR for Gaussian noise was minimum and also the MSE was maximum. However, reducing the bit rate in Bayesian Estimator normally affects image quality adversely, which was the limitation in this work.

Padmavathi *et al.*, (2012) presented a hierarchical method utilizing the advantage of the exemplary based method by handling the structures of the image separately through wavelets. Selection of the decomposition level depended on the mask size. Although this method

produced visual quality better than other methods, it however, changed the overall brightness and contrast of the image as the number of levels increased.

Kaur *et al.*, (2013) reviewed image enhancement based on Histogram Equalization (HE) technique. Minimum Mean Brightness Error Bi-HE (MMBEBHE) method was the extension of BBHE method that provided maximum brightness preservation. Though these methods could perform good contrast enhancement, they also caused more irritating side effects depending on the variation of grey level distribution in the histogram. Brightness Preserving Dynamic Histogram Equalization (BPDHE) could preserve the mean brightness better than BBHE, DSIHE, MMBEBHE, RMSHE, MBPHE, and DHE. However, in all their review and analysis, the researchers limited their research work to enhancement of contrast and brightness, but failed to include hue and saturation, except the variation of grey level which was peculiar to contrast, brightness in black and white images only.

Premnivas *et al.*, (2013)suggested a 3-D de blurring method to reduce motion blur from a single motion-blurred video to produce a high resolution video in both space and time. The researchers up-scaled the input video in space and time without explicit estimate of local motions and then performed 3-D de-blurring to obtain the restored sequence. However, the researchers only assumed that exposure time was known in this work, but failed to consider where the exposure time is unknown.

Keller *et al.*, (2013) proposed an energy-based algorithm for motion-compensated Video Super-Resolution (VSR) targeted on up-scaling of Standard Definition (SD) video to High Definition (HD) video. Since the motion (flow field) of the image sequence was generally unknown, they introduced a formulation for the joint estimation of a super-resolution sequence and its flow field. But the registrations of video still became a complex task of

computing optical flow (motion estimation) in this work, which made this research work inefficient in terms of correct high frequency image content.

Ghodke and Ganorkar, (2013) presented comparison between conventional image processing techniques and image enhancement using fuzzy intensification factor. The approaches applied to modify the images were grouped into two categories, one was the spatial domain approach and the other the fuzzy domain approach. They considered contrast enhancement for the fuzzy based enhancement algorithm. It was observed that sharper images were obtained using fuzzy intensification techniques. It was also observed that entropy and index of fuzziness of fuzzy intensified images were much less than that obtained using conventional filters. Fuzzy intensification was more effective than conventional filters for image enhancement, but its enhanced contrast of the images lacked brightness preservation.

Kamboj and Rani, (2013) described various types of noise models and filtering techniquessuch aslinear and non-linear techniques. In their hybrid filtering scheme, there were two or more filters which were recommended to filter a corrupted location. The decision to apply each particular filter was based on the different noise level at the different test pixel location. However, the hybrid filtering schemes do not preserve details of image because some details of the image are lost.

Verma and Ali, (2013) discussed different types of noise corruptions in images during image acquisition or transmission. Experimental results presented, concluded that BM3D and median filters performed well. Whereas averaging and minimum filters performance recorded the worst results. BM3D was the best choice for removing the salt and pepper noise. Whereas in other cases median filters were more suitable. However, the researchers might have obtained a good result with respect to BD3M, but median filter tended to remove image

details while reducing noise at thin lines and corners. Again median filtering performance is not satisfactory in case of signal dependent noise.

Deepa *et al.*, (2014) combined blur estimation for super resolution process to improve the quality of blur estimates by enhancing strong soft edges toward step edges, while filtering out weak structures. This method also accommodated arbitrary scaling factors, providing state-of-the-art results in terms of Peak Signal to Noise Ratio (PSNR) as well as other quantitative visual quality metrics like MSE. However, analytical results showed that the reconstruction constraints provided less useful information as the magnification factor increased, which implied that for large enough magnification factors, would lead to overly smooth results.

Gwanggil, (2014) proposed a luminosity conserving and contrast enhancing histogram equalization method for colour images. The histogram equalization was one of the ordinary methods employed for enhancing contrast in television and images for consumer electronics where unwanted subjective deterioration is frequently occurring. The final result showed that HSV colour space yielded favourable results in MSE by giving the luminosity conserving ability. It is always established that HSV colour space is characterized with a lot of drawback such as freezing of colour as a result of oscillation created by chips in the camera whose end result is noise.

Singh and Sharma, (2014) surveyed image enhancement techniques focusing on the different image. Image enhancement is found to be one of the most important vision applications because it has the ability to enhance the visibility of images. Their research work established that the nonlinear image enhancement could be used to improve the quality of a blurred image by using the concept of the light source refinement. It was found that the available methods employed in their techniques did not provide better results in multiple light sources because no modification was done on the hue and saturation. The image enhancement

technique could be improved by modifying the hue and saturation, which provides better results than theirused techniques.

Ballabeni *et al.*, (2015) evaluated how the radiometric pre-processing of image datasets could help in improving the performances of state-of-the-art automated image processing tools. An efficient pipeline based on colour enhancement, image de-noising, RGB to grey conversion and image content enrichment was presented. The performed tests demonstrate an effective image pre-processing, which considered the entire dataset analysis that could improve the automated orientation procedure and dense 3D point cloud reconstruction, even in case of poor texture scenarios. The employed tools for automated image orientation did not provide a unique standardize type of output.

Zhang and Liang, (2015) developed a scheme to estimate the distortion of the synthesized view when transmission error is introduced into the encoded reference texture and depth images. The method was capable of modelling the warping competition effect such that the necessary depth and texture distributions in the synthesized views could be accurately obtained. Experimental results demonstrated the accuracy of the proposed scheme. The researchers' method lacked adaptive parameter selection for both the deformable and the coupling model, which was a serious limitation.

Brochier *et al.*, (2015) presented a comparative study of thirteen methods of segmentation applied to a problem of extraction of tree leaves algorithm in smartphone images. The researchers first highlighted the performance obtained by the guided active contour approach, developed in the research by Cerutti et al. By improving the accuracy and quality of segmentation, their approach suffered defects related to the under-segmentation of image.

Pandey *et al.*, (2015)employed a new image enhancement process using neuro fuzzy method. They used Mamdani based fuzzy inference system, where a noisy image was selected.The

image was then enhanced using normal histogram image enhancement method, fuzzy image enhancement method, spatial domain method, and proposed Mamdani based neuro fuzzy image enhancement method. The implementation was carried out in MATLAB7.8.0 (R2009a) and Neucom. The result from their combined methods was far better in contrast enhancement than individual results of these methods. The evaluation was done on the basis of Root Mean Square Error (RMSE). The larger the RMSE value the better the image quality. However, contrast is not the only component of image enhancement but other components are involved such hue and saturation which were not considered in their enhancement. Also, their image enhancement technique was complex and difficult for human interpretation.

2.3.2 Review of Similar Works on Compression

Tomas (2010) proposed a video compression method based on the multi-dimensional discrete cosine transform. In the experimental results, substantial effects of this part on the compression ratio and particularly on picture quality were shown. The last part of the encoder was the run-length coding followed by lossless entropy coding and Huffman coding. However, Huffman coding was always accompanied with some degree of distortion because of the quantization process (frequency coefficient modifications).

Narwaria and Weisi(2010) suggested a new approach to addressing image compression problem, with the use of singular vectors out of singular value decomposition (SVD) as features for quantifying major structural information in images and then, support vector regression (SVR) for automatic prediction of image quality. Experiments conducted with three independent databases confirmed the effectiveness of the proposed system in predicting image quality with better alignment with the HVS's perception than the relevant existing work. However, though the researchers tried in their work, but for a lossy compression the final result cannot be devoid of some degree of distortions in the form of artefacts.

Rajkumar and V Latte (2011) proposed an efficient Region of Interest (ROI) encoding scheme with diverse resolution. The renowned wavelet based image encoding scheme SPIHT were used by the proposed encoding scheme. The ROI coding started with the selection of ROI and its corresponding resolution by the user. The diverse ROIs are encoded with diverse resolution (bpp) by applying lifting wavelet transform and Set Partitioning in Hierarchical Trees (SPIHT). The experimental results illustrated that using lifting wavelet transform and SPIHT, the proposed ROI encoding scheme provides high compression ratio and quality ROI. However, Setting Partitioning in Hierarchical Trees (SPIHT) and with the attainment of high compression may not be devoid of little degradation in terms of quality.

Faghih and Moghaddam (2011)presented a method called Neural Grey Edge. The method employed a neural network to model the Grey-Edge assumption based on image statistics. In other words, a Grey-Edge act as a global search that finds the neighbourhoods of the scene illuminate vector and then, the neural network acts as a local search and compensates the Grey-Edge error. Experiments result in a large dataset of 11000 images showed that the proposed approach outperformed the current state of the art algorithms. However, the research work were only limited to colour, no any consideration to luminance intensity or contras which played a prominent role in image processing.

Lima*et al.* (2011)proposed a new method to optimally define quality layers using Integer Linear Programming and distortion models. Scalable Video Coding offers the possibility to adapting the content following the "quality layer" abstraction. The performances of the proposed approach are comparable with the state-of-the-art methods, however, they are obtained with strong complexity reduction and augmented flexibility.

Takahashi*et al.* (2011)developed a new numerical model that combined a frequency-domain SR model and a rate-distortion theory for lossy image compression. The researchers

considered several factors that affected the reconstruction quality, and revealed that SR decoding performed better in low bitrates. They also conducted real-image simulations and confirmed that both the numerical analysis and real-image simulations exhibit quite similar tendencies, which supported the effectiveness of their numerical model. However, the final result cannot be completely devoid of noise because the lossy compression is never a distortion free compression.

Mauro *et al.* (2011) proposed a new method to optimally define quality layers using Integer Linear Programming and distortion models. The performances of the proposed approach are comparable with the state-of-the-art methods. Scalable Video Coding offers the possibility to adapt the content following the "quality layer" abstraction. The performances of the proposed approach were comparable with the state-of-the-art methods. However, they are obtained with strong complexity reduction and augmented flexibility.

Naemura and Tanaka (2011) proposed a new numerical model that combines a frequency-domain SR model and a rate-distortion theory for lossy image video compression. The researchers considered several factors that affect the reconstruction quality, and revealed that super resolution (SR) decoding performed better in low bitrates. They also conducted real-image simulations and confirmed that both the numerical analysis and real-image simulations exhibit quite similar tendencies, which supported the effectiveness of their numerical model. However, the research work was not completely free from artefacts (distortion). The distortion level can only be minimized through the numerical analysis and real image simulation because the techniques used was for a lossy compression.

Bisjerdi and Behrad (2012) came up with a new method for motion estimation and compensation and its application for video compression. This algorithm was based on the mesh energy minimization with novel sets of energy functions. The proposed energy

functions had an appropriate features, which improved the accuracy of motion estimation and compensation algorithm. They employed the motion estimation algorithm in two different manners for video compression. In the first approach, the proposed algorithm was employed for motion estimation of consecutive frames. In the second approach, the algorithm was applied for motion estimation and compensation in the wavelet sub-bands.

Kaur and lalit (2012) proposed acomparative analysis of DCT, DWT and LWT for Image compression. In this paper, comparison of various transforms based image compression method was described. The DCT showed its best results in terms of energy compaction but execution time required and MSE that is the error between original and recovered image was not acceptable. So to speed up the process and to improve the MSE, DWT based compression can be done. But in DWT results SNR reduces, so to improve further SNR & ET, lifting based decomposition process with a comparable performance in the compression ratio and peak signal to noise ratio (PSNR) and reconstructed image quality. Wavelet based compression was best suitable for the applications in which the speed was critical factor that is software based video conferencing and real time image compression systems. However, the speed up mechanism can be more improved by using the lifting scheme being a lossless compression method.

Augustin and Senthil (2012) employed a new three dimensional discrete cosine transform (3D-DCT) based video compression algorithm that will select the optimal cube size based on the motion content of the video sequence. It was determined by finding normalized pixel difference (NPD) values, and by categorizing the cubes as “low” or “high” motion cube suitable cube size of dimension either [16×16×8] or[8×8×8] were chosen instead of fixed cube algorithm. The researchers used the standard variable length coding method that was stated for 2D-DCT based video compression technique. However, variable length coding was a lossy compression method, and was also accompanied with some degree of distortion.

.**Er. Ramandeep and Navneet (2012)** proposed a new hybrid scheme combining the DWT and the DCT algorithms under high compression ratio constraint. The algorithm performed the DCT on the lowest level DWT coefficient. DCT yielded an advantage of redundancies of the data by grouping pixels with similar frequencies. The paper covered some background of wavelet analysis, data compression and how DCT and DWT can be used for image compression and the researchers proposed a hybrid DWT-DCT algorithm for image compression and reconstruction taking benefit from the advantages of both algorithms. The analysis showed that for a fixed level of distortion, the number of bits required to transmit the hybrid coefficients. However, little distortions were still evident due to compression.

Horn, (2012) presented a vector quantization base firefly algorithm for image compression in order to address the local optimal codebook usually generated by the Lide-Buzo-Grey algorithm. This paper employed the firefly algorithm to construct the codebook of vector quantization and used the Lide-Buzo-Grey method to initialize the firefly algorithm. Simulation results were compared with similar methods and results showed that the technique proposed in this research is efficient with a faster convergence. However, the work did not consider the efficient of the firefly algorithm when applied directly on image compression; moreover his research work was on a still picture of code book not a moving image.

Zhiyuanet *et al.* (2012) proposed a No-reference video quality assessment metric based on H.264/AVC bit stream through extracting features from the H.264/AVC encoded bitstream. After the extraction of the features which were very important for video quality assessment, the researchers used Partial Least Squares Regression (PLSR) to calculate the weights of them. Then a quality prediction model was also proposed. During the experiments, the results showed that their NR metric had low computing complexity. Finally, compared to subjective assessment, which showed that there was a high correlation between quality prediction and

the actual quality by 0.95. Though, the researchers came up with good computation, however, the final result was not completely devoid of noise.

Abdullah and Rao (2013) in this paper, the compression performance of classical and lifting based wavelet transforms were presented. Haar wavelet, Daubechie wavelet, Coiflet wavelet, Demeyer wavelet, and Symlet wavelets were considered under classical wavelet category. Lifting based wavelets were considered under Lifting based wavelet transform category. All the Classical wavelets produced less PSNR around 30dB and less compression ratio around 2bpp. But, Coiflet wavelet produced high PSNR around 47dB, but at low compression ratio in the ranges of only 2bpp. The lifting based wavelet transforms produced high PSNR and compression ratio. The PSNR was above 40dB and compression ratio was around 8bpp signifying an excellent result. From the analysis, it wasestablished that the lifting based wavelets outperformed the classical wavelets.

Kejgir and Kokare (2012) proposed a novel robust digital image watermarking method based on LWT-SVD. To investigate the robustness of the proposed algorithm, five different images and eighteen different intentional and unintentional attacks were employed. PSNR values were estimated for all the five images. Comparing these PSNR values with that obtained by earlier approach based on DWT-SVD. The researchers concluded that the fidelity of the watermarked image can be considered as an improvement with the proposed method. Similarly, the correlation coefficient between retrieved watermark and original watermark were estimated for all the images. From these coefficients it was observed that the watermark retrieval was highly efficient with proposed method. Here it can be concluded that the proposed algorithm has two-fold advantages, firstly there were improvement in PSNR values (with no degradation of watermarked image quality) and secondly there was simultaneous improvement in the CRC values (improved robustness of watermark).

Chandan and Sukadev (2013) proposed a combination of DCT and fractal image compression techniques. DCT was employed to compress the colour image while the fractal image compression was employed to evade the repetitive compressions of analogous blocks. Analogous blocks were found by using the Euclidean distance measure. Comparative analysis were performed to prove that their system was competent to compress the images in terms of Peak Signal to Noise Ratio (PSNR), Structural Similarity Index (SSIM) and Universal Image Quality Index (UIQI) measurements. However, for any compression techniques it is expected to be accompanied by some element of degradation in terms of quality of the image especially when DCT is involved.

Staelens*et al.* (2013) proposed a novel no-reference bit stream-based objective video quality metric that was constructed by genetic programming-based symbolic regression. The benefit of this approach was that it calculated reliable white-box models that allowed the researchers to determine the importance of the parameters. In addition, these models can provide human insight into the underlying principles of subjective video quality assessment. However, numerical results showed that perceived quality can be modelled with high accuracy using only parameters extracted from the received video bit stream.

Hassan and Pouriya (2013) suggested a method in which due to compression of the obtained bits by VLC block (obtained from difference between bits made by VLC block encoder and bits made using Huffman method) and replacing the bits created by the secondary channel encoder, the researchers could increase quality of the received video frames in comparison with the method indicated by (Farooq *et al*. 2009). The proposed method through increasing the channel coding rate and therefore more protection of transmitted bits and maintaining overall transmission rate constant, could improve the quality of the reconstructed video frames. This was achieved by using intelligent neural network and Huffman coding in VLC blocks used in the MPEG standard to compress transmitted data

significantly. However, Huffman coding used in MPEG compression is accompanied by very little error during motion compensated prediction.

Raghavendra and Anita (2013) suggested a DWT based compression technique algorithm which provided a very effective way to compress with minimal loss in quality. Although the actual implementation of the JPEG2000 algorithm was more difficult than other image formats, and the actual compression of images were considered expensive computationally.The high compression ratios that can be routinely attained using the JPEG2000 algorithm easily compensated for the amount of time spent implementing the algorithm and compressing an image. However, the DWT is a lossy compression and cannot be completely devoid of noise.

Hazarathaiah and Rao (2014) developed a new lifting based wavelets method of calculating lifting coefficients. In this method, new basis functions are utilized to ease new orthogonal traditional wavelets. Then by using the decomposing poly-phase matrix the lifting steps are calculated using a simplified method. It was found that the compression ratio (CR) and Peak Signal to Noise Ratio (PSNR) were far ahead of that was obtained with the popular traditional wavelets as well as the successful 5/3 and 9/7 lifting based wavelets. Set Partitioning in Hierarchical Trees (SPIHT) was used to incorporate compression. However, Setting Partitioning in Hierarchical Trees (SPIHT) and with the attainment of high compression may not be devoid of little degradation in terms of quality.

Hamsavahini *et al.* (2014) presented a survey of the existing lifting based implementations of 1-dimensional and 2-dimensional Discrete Wavelet Transform. The researchers briefly described the principles behind the lifting scheme in order to better understand the different implementation styles and structures. They presented several architectures of different flavours ranging from highly parallel ones to highly folded ones to programmable ones. They provided a systematic derivation of the architecture and evaluated them with respect to their

hardware and timing requirements. However, their research was not completely devoid of artefacts in the form of noise at the final stage.

Patel and Pathak (2014) proposed the architecture of Discrete Wavelet Transform (DWT) using lifting scheme for image compression applications in Xilinx and MATLAB software and then compare the results. The DWT played a major role in the field of signal analysis, computer vision, object recognition, image compression and video compression standard. Wavelet based techniques such as JPEG2000 for image compression had a lot more to offer than conventional methods in terms of compression ratio. However, storage of image requires higher bandwidth on the smaller devices; also image transmission requires higher bandwidth.

Deepak *et al.* (2014) developed, a new lossless image compression scheme based on the DCT. This method caused a significant reduction in entropy, thus making it possible to achieve compression using a traditional entropy coder. The method performed well when compared to the popular lossless. However, the end result is accompanied with distortion.

Chengging *et al.* (2014) proposed a discrete cosine transform (DCT)-based no-reference video quality prediction model, where the compressed natural video was statistically analysed (neural network) to take care of the compression discursion (artefact). The model has two stages: 1) distortion measurement and 2) nonlinear mapping. In the first stage, an unsigned ac band, three frequency bands, and two orientation bands are generated from the DCT coefficients of each decoded frame in a video sequence. Detailed experimental results demonstrated that the results of the proposed method were highly correlated with the subjective assessments; however the final result was not completely free from blurredness and noise manifested during segmentation of the DCT.

Xiuqi and Borko (2014) came up with an innovative image compression algorithm that utilizes three-dimensional discrete cosine transformation. The algorithm first divides an

image into 8x8 blocks. Then eight adjacent blocks are taken sequentially and repeatedly to form three dimensional data cubes, which were required by the 3D DCT transformation. Next, a three-dimensional discrete cosine transformation was performed on each data cube. The DCT coefficients in the same 3D data cube are then quantized and Huffman encoded. The experimental results have shown that the new algorithm is better than JPEG for some classes of images, however, the final output video was accompanied with some degree of distortion as a result of compression. Lossy compressions like the DCT are not devoid of noise or distortion,

Deepak *et al.* (2014) developed a new lossless image compression scheme based on the DCT. This method caused a significant reduction in entropy, thus making it possible to achieve compression using a traditional entropy coder. The method performed well when compared to the popular lossless. However, the end result of a DCT is accompanied with insignificant distortion in the lower frequencies which end result is noise.

Babu and Alamelu (2015) proposed a Novel Morpho Image Codec called L Shaped Morpho Codec (LSMC) based on Lifting Wavelet Transform (LWT). LWT was used for decomposing medical image into various sub-bands. Significant pixels of sub-band were tracked by LSMC in particular order. Morphological dilation immediately applied using L shaped structuring element if significant pixel are found. Experimental results showed that the proposed LSMC outperforms standard codec's such as Set Partitioning in Hierarchical Trees (SPIHT) and Set Partitioned Embedded block coder (SPECK) for Lossy and Lossless Compression for 512 x 512 & 1024 x 1024 images. The average bits per pixel (bpp) required for lossless compression by LSMC was less over by SPECK. However, even though the LWT is a lossless transform, the 512 x 512 image for block coder (SPECK) cannot be devoid of some element of noiseembedded.

Rabiul Islam *et al.*, (2015) suggested image compression based on compressive sensing using wavelet lifting scheme framework that addressed the best-compressed image components and preserved of high-frequency details in medical images. The proposed method was also compared with three different matrixes, which were Gaussian, Bernoulli and random orthogonal measurement matrix and image reconstructed by convex optimization technique for reconstruction of the image. From the above review and comparison, it was established that different compression techniques exist in literatures, among which are DCT, DWT, Lifting wavelet etc. Lifting wavelet and DWT are better compression techniques than the DCT (Preet and Geetu, 2012). The best compression technique among the three establishe d was Lifting wavelet Transform. Statistical and intelligent techniques were most often used to further reduce the level of distortion in video compression; prominent among them is the neural network which with series of training and re- training of the sample compressed video, it is possible to regain the skipped frames at the coding stage and the sample can favourably be compared with the original sample in terms of PSNR and MSE.

Chen *et al.*, (2016) presented a multi-level image segmentation using an improved firefly optimization algorithm. The firefly algorithm was applied to enhance the efficiency of the multilevel image segmentation. A diversity enhancement strategy with Cauchy mutation and neighbourhood strategies was introduced in order to improve the performance of the firefly algorithm. The proposed method was compared with three benchmark optimal algorithms, which were, Darwinian particle swarm optimization, hybrid differential evolution optimization, and firefly algorithm. The experimental results showed that the proposed method can efficiently segment multilevel images and obtained better performance than the other three methods. However, the research does not consider the image representation as a minimization problem. Thus the pixel information in the image is not entirely represented.

It is obvious from the literatures that; several research efforts have been directed towards video (dynamic image) enhancement. However, an acceptable compromise level of de-noising and contrast enhancement is still a challenging problem faced by researchers in the area of image processing. There are usually three basic problems with digital videos. These include:

i. Size of video window

ii. The frame rate of the video

iii. Quality of the video images

High amount of quantization distortion between image regions may not be acceptable to human visual perception. Image distortions are largely caused by random variation in pixel intensity which results in an uneven distribution of image histogram. For an efficient image enhancement technique, effort needs to be directed towards extracting the pixel information that constitutes the real object(s) in the image. This will help in focusing the enhancement technique only on the real objects while reducing the influence of the noisy content of the background pixel. This is the idea on which the enhancement technique presented in this report is based, with the aim of addressing some of the challenges highlighted in the reviewed literature.

CHAPTER THREE

MATERIALS AND METHODS

3.1 Introduction

In this section, the step by step procedures adopted in carrying out this research work, which include the development of an image brightness enhancement and modifying FOA in order to achieve better image quality are highlighted as follows:

3.2 Materials used in this research work

The materials used in carrying out this research work are listed as follows:

1. Digital HD Video Camera Recorder (HVR-Z1U/Z1N) and its tripod is used for shooting the four sample images from NAERLS and NTA.

2. Sample data acquired from NAERLS (NAERLS1.avi and NAERLS2.avi), NTA (NTA1.av i and NTA 2.avi) and those acquired from the online standard database (Akiyo.avi and Forman.avi). Other materials used include:

3. MATHLAB Image Processing Tools box: The image processing toolbox is a group of functions that cover the capability of the MATHLAB numeric computing environment. The toolbox functions are MATHLAB M-files which are series of MATHLAB statements that carries out specialised image processing algorithms. The toolbox provides anextensive range of image processing operations, including:

i. Geometric operations
ii. Image analysis and enhancement
iii. Transforms

iv. Linear filtering and filter design

v. Region of interest operations

3.3Research Methodology

The details of the method, and the steps adopted in achieving this research work are listed below.

3.3.1 Pre-processing:

i. Acquisition of video sample data and the online benchmark data base for analysis.

ii. Elimination of hue and saturation in order to retainonly luminance intensity of the sample video data to pave way for luminance (grey level) enhancement.

iii. Noising and filtering of the video data such that its pixel is varied from its true value in order to separate the true picture from its background for easy of analysis.

iv. Applying the developed brightness enhancement technique to improve the luminance intensity of the video frames obtained from NEARLS and NTA to determine the improvement achieved in the image quality.

3.3.2 Video Acquisition

A total of four different video data images were acquired from National Agricultural Extension Research Liaison Service (NAERLS) and Nigerian Television Authority (NTA)which were compared with the benchmark video data images i.e Akiyo avi and Foreman avi in order to meet the required features for image processing. These video images are given in Figure 3.1.



(a) NAERLS1.avi (b) NAERLS2.avi



(c) NTA1.avi (d) NTA2.avi



(e) Akiyo.avi (f) Forman.avi

Figure 3.1: Frames of Sampled Videos and Benchmark Videos

From Figure 3.1, the first two sample videos from NAERLS (NAERLS1.avi and NAERLS2.a vi) were obtained using the video camera system of NAERLS. Similarly, the two sample videos from NTA (NTA1.avi and NTA2.avi) were obtained using the video camera system of the NTA, Abuja Broadcasting Station and the last two benchmark videos (Akiyo.

avi and Forman avi) were also obtained from the standard image database (VintaSoftImaging. Net). MATLAB R2015a image processing toolbox was used on individual video images

obtained and represented in Figure 3.1 to achieve the information on each videoas shown in Table 3.1.

Table 3.1: Simulation Result Sample Video Data

SN	File Names	File Size (*.avi)	File Frames
1	NAERLS1.avi	18.1Mb	157
2	NAERLS2.avi	10.3Mb	155
3	NTA1.avi	9.6Mb	152
4	NTA2.avi	11.2Mb	200
5	Akiyo.avi	11Mb	300
6	Foreman.avi	7.25Mb	100

It is noteworthy that, videos are frames of dynamically changing images and images are usually derived from video cameras in the form video frames. Therefore, the video data given in Table 3.1 were initially converted into frame of static images and cropped for easy processing and analysis as in Appendix A. These are done as follows:

i. Video Conversion

The video frames conversion into images is done using MATLAB image processing toolbox *frame2im* command.The snippet of the conversion process is given as follows:

```
%Convert Video Frames to Images
VIDEO=aviread('Enter_Video_Name_Here.avi');
IMAGE=frame2im(VIDEO(1,df));
figure
imshow(IMAGE);
```

In these MATLAB scripts, the *avi read* reads the input video in *.avi* format. The *df* in the image is used to specify a particular frame to be converted into an image.

ii. Image Cropping

The acquired image data contain irrelevant parts which are not necessary in the interpretation of the image. Therefore, these irrelevant parts are cropped out such that further processing focuses only on prominent parts of concernand thereby improving the image frame, aspect ratio, and reducing computational cost. The image cropping is done using the following MATLAB command.

```
Column=imrop(IMAGE,[a,b,c,d]);
```

3.3.3 Eliminate Hue and Saturation and Retain Luminance Intensity

The next stage in the pre-processing was to extract the luminance information from the true colour image by removing the hue and saturation. In order to achieve this, conversion formula was employed as follows (Keith, 2005):

$$Intensity = a \times (red) + b \times (green) + c \times (blue) \tag{3.1}$$

where a, b, and c are the hue weighted average of the imagesRed , Green, and Blue (RGB) channels. Refer to the luminance equation (2.1) in the fundamental concept.

These individual values of channels are usually greater than zero, depending on the colour content of the original image. These coefficients were adjusted based on sensitivity to colours until appropriate luminance intensity was obtained.The constituent colours of the image were extracted and the image intensity given in equation (3.1) was transformed into equation (3.2): (Keith, 2005)

$$Intensity = a \times (1,:,:) + b \times (:,2,:) + c \times (:,:,3) \tag{3.2}$$

From equation(3.2), the first matrix is with only the coefficient multiplying the Red, coefficient of both Blue and Green represented in hidden does not exist. The same procedure as in the second and third matrix where the Blue and Green exist.

3.3.4 Noising and Filtering

Noise was applied to the intensity image given in equation (3.2). Four different type of noise (Gaussian, Poisson, salt& pepper, and speckle) were implemented. The essence of this noise was to randomly vary the intensity of the image such that the pixel values showed different values from its true values. By so doing, the actual image was separated from its background. Since the image intensity ranges from 0 to 1, an additionof parameter (*d*) wasused to control the level of the noise added to the image. This was implemented using the inbuilt image processing toolbox in MATLAB R2015a and its snippet is given as follows:

```
IM=IMAGE;
IN1=imnoise(IM,'salt & pepper',d);
IN2=imnoise(IM,'poison',d);
IN3=imnoise(IM,'gaussian',d);
IN4=imnoise(IM,'speckle',d);
```

The Noisy Image (*IN*) fromthis operation was smoothed using the median filter and hybrid median filter which are two of the most widely used filtering technique. The median filtering was done by selecting the median value from the neighbourhood as the output of each pixel. The inbuilt image processing toolboxin MATLAB R2015a was used for its implementation as follows:

```
IMF1=medfilt2 (IN, [K,L]);
IMF2=hmf (IN,[K,L]);
```

where, K and L are the selected neighbourhood operation of the image see Appendix A.

3.3.5 Brightness Enhancement

Although the standard colour transformation is relatively easy to implement and many researchers have actually adopted it for coloured video frames, it still has great challenges,especially when the objects in a particular video frame have very similar grey scale values. This research used a different enhancement technique based on the following approach:

i.) At the first stage, the pixel values throughout the image was extracted and stored in a buffer (B). The dimensional sizes of the image were first determined and stored as X number of rows and Y number of columns. Therefore, from MATLAB toolbox R2015a, each pixel value extraction formula is given as:

$$B(k+1) = \sum_{k=0}^{N} \left(\overline{\frac{I(k)}{X \times Y}} \right) \tag{3.3}$$

where:

N is the total number of possible intensity levels, which is usually 256 for 8-bit grey scale image.

I is the input image,

Equation (3.3) only extracts part of the image that constitute the pixel information

ii.) At the next stage, the pixel information from equation (3.3) is converted into a pixel matrix as follows using MATLAB code as in appendix A:

$$W = \sigma \times 1, (N+1) \tag{3.4}$$

Therefore, substituting equation (3.4) into equation (3.3) gives the following:

where σ is zero (variable, but kept at zero) and $B\ l+1$ is the updated picture information.

$$B(l+1) = \sum_{m=0}^{N+1} B(m) + \sum_{l=0}^{N+1} w(l) \qquad (3.5)$$

iii.) In this stage, the elements of the update equation (3.5) were converted to integer valuesfor further processing as follows:

$$R = \lfloor B(l+N) \times (N+1) + 0.5 \rfloor \qquad (3.6)$$

The addition of 0.5 in equation (3.6) was to ensure that the element of the image stored as R was always rounded up to 1because this is when the maximum intensity of the image is achieved.If recalled,the probability density function of brightness is rounded to 1 as in equation (2.7). The values of equation (3.3) that correspond to any of the values of Rwere determined and stored as *P* using the following:

$$P = \sum_{i}^{N+1} sum(B(\ find(R = i)) \qquad (3.7)$$

where sum and find are operators in the MATLAB code as shown in appendix A. Equation (3.7) is the improved brightness or luminance of the employed method that enhanced the level of luminance intensity.

In the final stage, the pixel values of the original image were compared with R and the best from both images was kept. This was achieved using the following formula:

$$F \quad find \quad I = i \quad = \sum_{i=0}^{N} R(i+1) \qquad (3.8)$$

The use of *find* in equation (3.8) was to determine the linear indices corresponding to the nonzero element of the image.

The output from equation (3.8) was the improved brightness or luminance of the used image brightness enhancement technique. This technique is robust to any kind of image whether coloured or grey scale images these was evident in the result of the enhancement model.

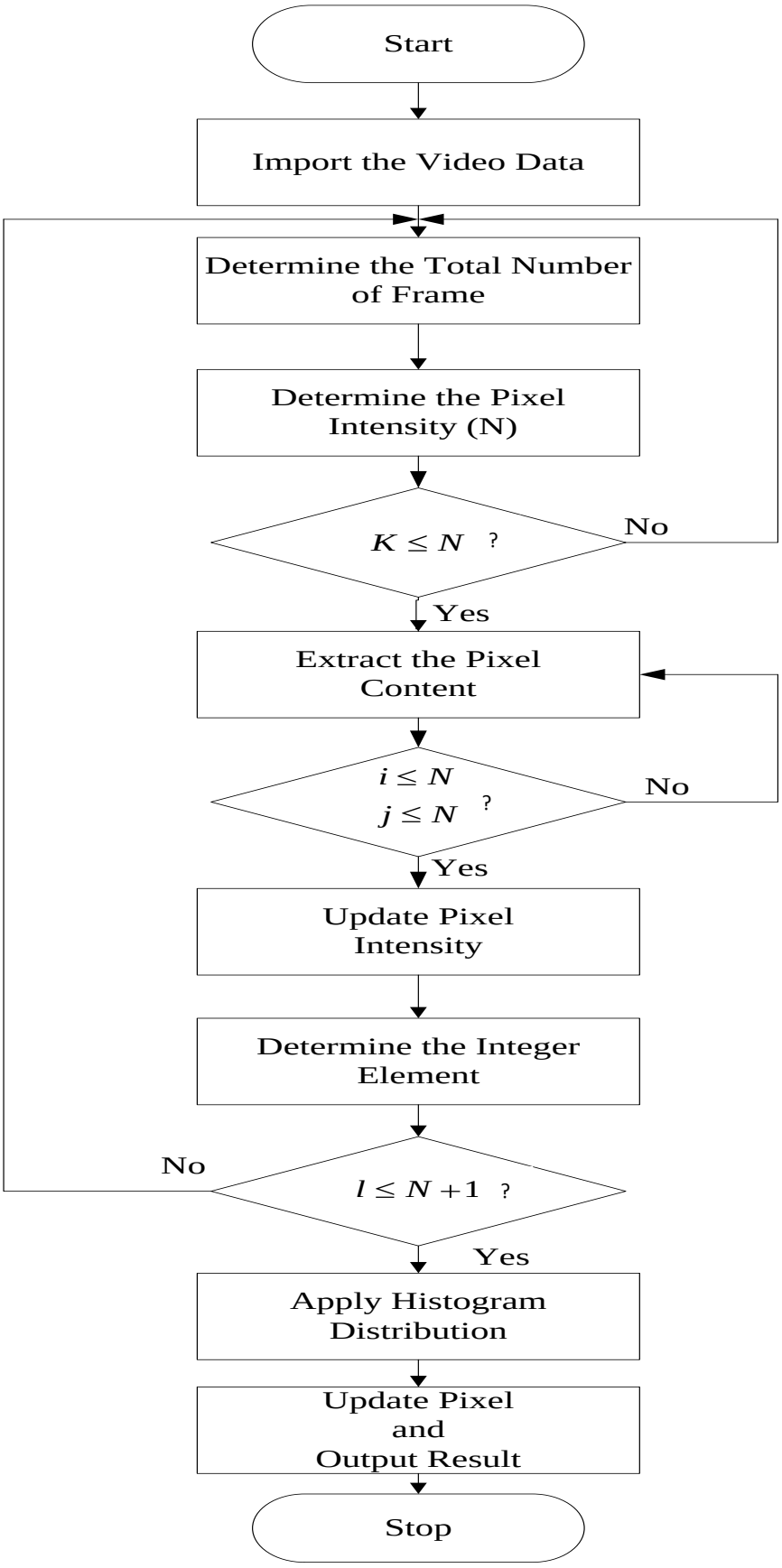


Figure 3.2: Flowchart Implementation of Luminance Enhancement Technique

Figure 3.2 is the flowchart implementation of luminance enhancement model, where video is read from the first block after start. The process involves importing the video into the MATLAB simulation environment and determining the total number of frames. At this point the video sample is converted into frames in the first part of Appendix A. Then the pixel intensity is determined. At this point, the read is generated, the new video is read, and its number of frames, size, and amount of pixels intensity are determined. The first decision box initialises K when N is 256 pixels if Kstarts from 0. But when K starts from 1, then N becomes 255 + 1. These is just like initialising the counter over the entire number of images. If every time K is less than or equal to 1 the system continues to count the number of pixels in an image to be enhanced. Note, it is the pixels that are subjected to enhancement. But if K is greater than 1, the process gets stocked and repeated until that condition is achieved. The initialised counter now goes through each of the 256 pixels between 0 and 256, then it moves to extract the pixels content from the image. In the next decision box, i and j are initialised and they must always be less than or equal to N. If they are less than 1, the initializationprocess starts from 0, if not the process goes back to start the initialization again. If the condition is achieved then the pixel intensity is updated. The next stage is to determine the integer element of the box before the next decision box used in initialization of an integer value l expressed in equation (3.6) and appendix A. Hence, a counter is needed to count from 0 to N +1 so that this can go through all the pixel of the image. At this point the histogram distribution is applied as in appendix A and equation (2.4). The output result and updated pixels are achieved and the process comes to a final stop.

Figure 3.3 research block diagram shows the different phases of the work.

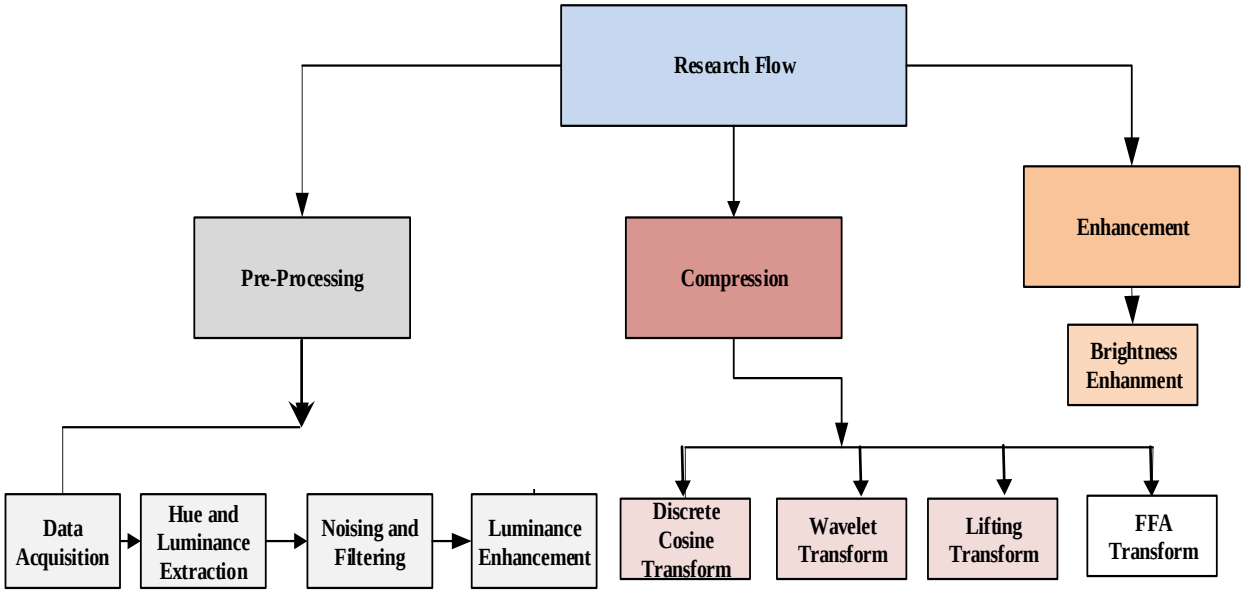


Figure 3.3: Block Diagram Representation of the Research Methods

From Figure 3.3, it is observed that, this section covers the first part (pre-processing) of the research methodology. This included the video data acquisition, hue and luminance extraction, noising and filtering and finally brightness enhancement. This was implemented in MATLAB R2014b as presented in Appendix A

In order to evaluate the performance of the brightness enhancement technique, this research implemented three video compressions transforms on the output of the enhancement video. These compressions transforms (DCT, DWT, and LWT) were also implemented on the original video data in order to evaluate and determine the compression between its enhanced and its original forms.

3.4 Enhanced Discrete Wavelet Transform

The Discrete Wavelet Transform (DWT) has an input sample image signal expanded with respect to its basis called the mother wavelet. The process of DWT employed in this research is made of two stages:

Stage one: In this stage, a vertical subsampling is applied to obtain the low pass sub band L and the high pass sub-band H.

Stage Two: In this second stage, a horizontal subsampling is applied to the first stage to obtain the LL,LH, HL, and HH sub-bands, respectively.

At every level of the wavelet transform*(Li and Drew, 2003)*, four output images (approximate, vertical, horizontal, and diagonal) details are obtained. In this research, the 2D wavelet transform was implemented by multiplying the wavelet function by the scaling function as follows*(Li and Drew, 2003)*:

$$F(x,y) = \psi(x,y) \times \varphi(x,y) \tag{3.9}$$

After which the four details of the image compression were determined as follows*(Li and Drew, 2003)*:

Approximate detail

$$F(x,y) = \varphi(x)\varphi(\mathrm{y}) \tag{3.10}$$

Horizontal detail

$$F(x,y) = \psi(x)\varphi(y) \tag{3.11}$$

Vertical Detail

$$F(\mathrm{x},\mathrm{y}) = \psi(\mathrm{x})\varphi(x) \tag{3.12}$$

Diagonal Detail

$$F\ \ x,y\ \ = \psi(x)\psi\ \ y \tag{3.13}$$

The approximate detail is repeatedly passed through a low pass (L) and a high pass (H) filter bank until an appropriate level of compression is achieved as (Li and Drew, 2003):

$$L = \frac{1}{\sqrt{2}}(1,1) \tag{3.14}$$

$$H = \frac{1}{\sqrt{2}}(1,-1) \tag{3.15}$$

The flowchart implementation of the DWT is shown in Figure 3.3.

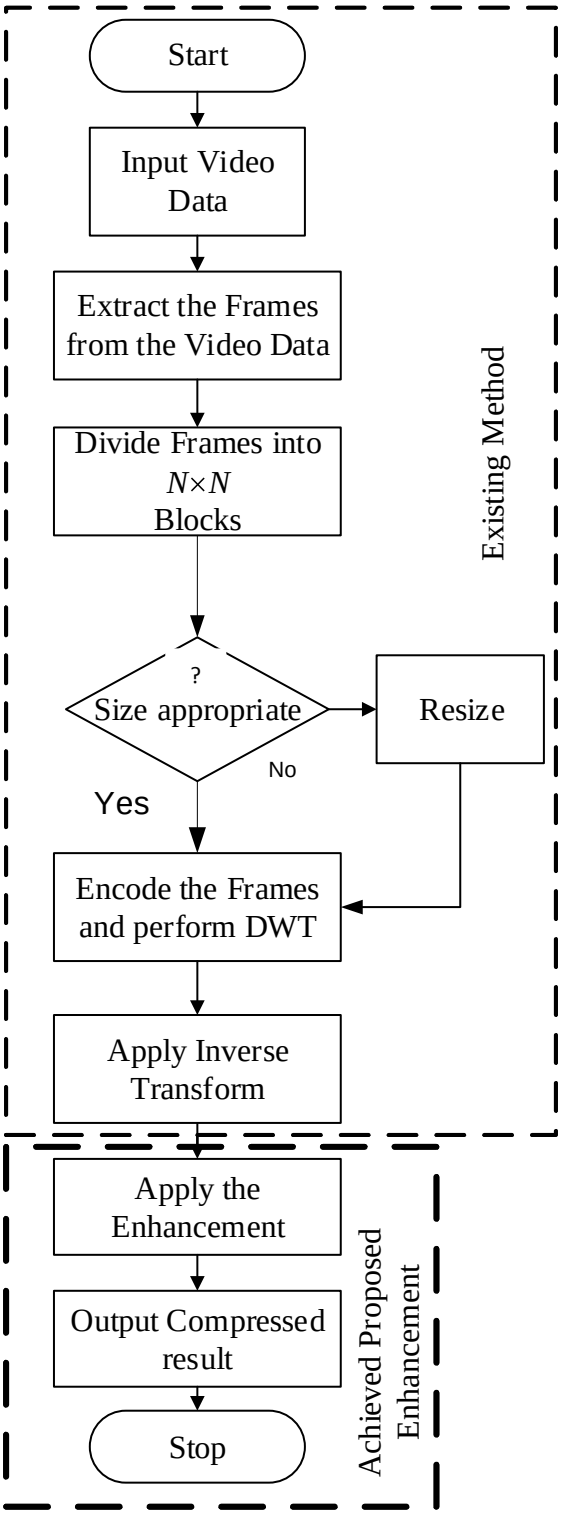


Figure 3.4: Flowchart of MATLAB Implementation of Improved DWT Video Image

Figure 3.4 is the flowchart implementation of*Improved DWT Video Image*where video is read from the first block after start. The process involves importing the video into the MATLAB simulation environment and determining the total number of frames. At this point the video

sample is converted into frames in the first part of Appendix C and the converted frames are extracted.The frames are then divided into segmented blocks of N x N. The decision box decides if the segmented blocks are divided into appropriate sizes, if “No” then the blocks must be resized and if “Yes”the process goes ahead to encode the frames and perform the DWT.The inverse transform is then applied to regain the compressed image. The enhancement model is then applied as in appendix A to achieve the enhanced compressed image.

However, this processis applied to each of the standard compression techniques (DCT, DWT, and LWT) to obtain their respective results. The only difference observed in each of these techniques is in the encoding process where respective transform operation is performed.

3.5 Enhanced Discrete Cosine Transform

Discrete Cosine Transform(DCT) is an orthogonal transform method which is based on the Fourier transform technique. In this research, it was assumed that the image frames of the entire sample video datum contained both real and even functions: Therefore, the DCT is derived from the Discrete Fourier Transform (DFT) of the enhanced output as follows:

$$F\ u,v\ = \frac{2}{N}\sum_{n_2=0}^{N-1}\sum_{n_1=0}^{N-1} c\ u\ c\ v\ f\ n_1,n_2\ \cos\left[\frac{2n_1+1\ u\pi}{2N}\right]\cos\left[\frac{2n_2+1\ v\pi}{2N}\right] \quad (3.16)$$

where:

$n_1, n_2, u, v, = 0,1,..., N-1$, f n_1, n_2 represent coefficients data in the original enhanced image block.

$F\ u,v$ represents the output coefficient in the block after DCT.

$c\ u,v$ is a $N \times N$ cosine matrix which is defined as follows:

$$c\ u\ = c\ v\ = \begin{cases} \sqrt{(1/\mathrm{N})} & if\ u = 0,\ 0 \le v \le N-1 \\ \sqrt{(2/N)\cos\left[\ 2v+1\ \right]/2N} & if\ 0 \le u \le N-1,\ 0 \le v \le N-1 \end{cases} \quad (3.17)$$

The inverse DCT of the enhanced input image is given by:

$$f\ \ n_1, n_2\ = \frac{2}{N}\sum_{v=0}^{N-1}\sum_{u=0}^{N-1} c\ u\ \ c\ \ v\ \ F\ \ u,v\ \cos\left[\frac{2n_1+1\ u\pi}{2N}\right]\cos\left[\frac{2n_2+1\ v\pi}{2N}\right] \quad (3.18)$$

After the DCT, the output block is divided into two sub-bands:

iii. **Low frequency sub-band**:This is used to store the important objects in the image.

iv. **High frequency sub-band**: This is also used to store the other details and texture of the image.

Emphasis was only on the low frequency sub band because it contains the actual object in the image.The coefficient of the high frequency sub-band is usually very close to or equal to zero in some instances, therefore, it was removed entirely from the algorithm for the purpose of efficient compression. The flowchart for the implementation of the DCT is given in Figure 3.5.

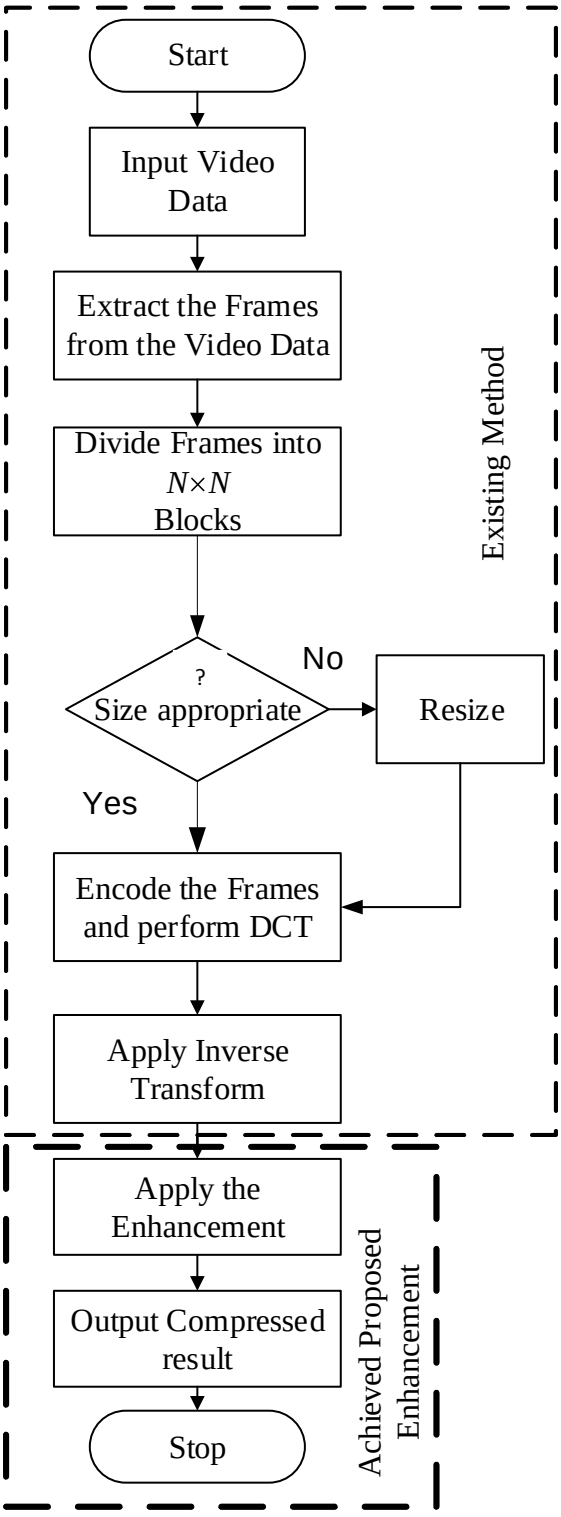


Figure 3.5: Flowchart of MATLAB Implementation of Improved DCT Video Image

3.6 Enhanced Lifting Wavelet Transform

The Lifting Wavelet Transform (LWT) is a computational efficient transform method developed to address some of the challenges of DWT. Thus the LWT implementation is similar to the DWT, except thatthe total number of samples at each stage is the same as the initial set of samples. In this research, the input image sample from the processed sample video was split into even and odd sets of samples for the efficient lifting filter to ensure appropriate approximation and detail extraction.

The step by step procedural approach for the implementation of the enhanced LWT based compression is highlighted as follows:

Step One

Input the sampled enhanced frames,that is, $F(i, j)$, where, $i = 1, 2, ..., N$ and j = 1, 2, …M+1.

The three stages (split, prediction, and update) of LWT on the image were then performed as:

i. The input image signal was split into even, f_e and odd, f_o samples as follows:

$$f_e\ m,n\ = F(m, 2n) \tag{3.19}$$

$$f_o\ m,n\ = F(m, 2n+1) \tag{3.20}$$

ii. The integer positions of the odd samples from the neighbouring even samples were predicted as follows:

$$h(m,n) = f_o(m,n) - p_e(m,n) \tag{3.21}$$

where, $h(m,n)$ is the resulting prediction residuals or high sub-band coefficients.

Assuming the sample pixels have a strong correlation in the angle θ_v and the integer pixels are marked by "O", the half pixels by "+", and the quarter pixels by "×". The prediction of $f\ m, 2n+1$ is taken as a linear combination of the even samples as follows:

$$P_e\ m,n\ = \sum_i \alpha_i x_e\ m + sign(i-1)\tan\theta_v,\ n+1 \tag{3.22}$$

where

$$sign(\mathrm{f}) = \begin{cases} 1 & f \geq 0 \\ -1 & otherwise \end{cases} \tag{3.23}$$

The weighting factor α_i is given by the filter coefficients.

i. In the updating step, the even samples are replaced using the following equation

$$l\ m,n\ = f_e(m,n) + u_h(m,n) \tag{3.24}$$

The values of $l\ m,n$ are always located at an integer position which is one of the characteristics of the LWT.

The brightness (luminance) enhancement technique was applied to the input at the transform stage, where the LWT is implemented for compression. Subsequently, the output reconstructed to enhance the video signals makes the LWT different from the standard LWT used by other researchers. The flowchart for the implementation of the enhanced LWT is given in Figure 3.6.

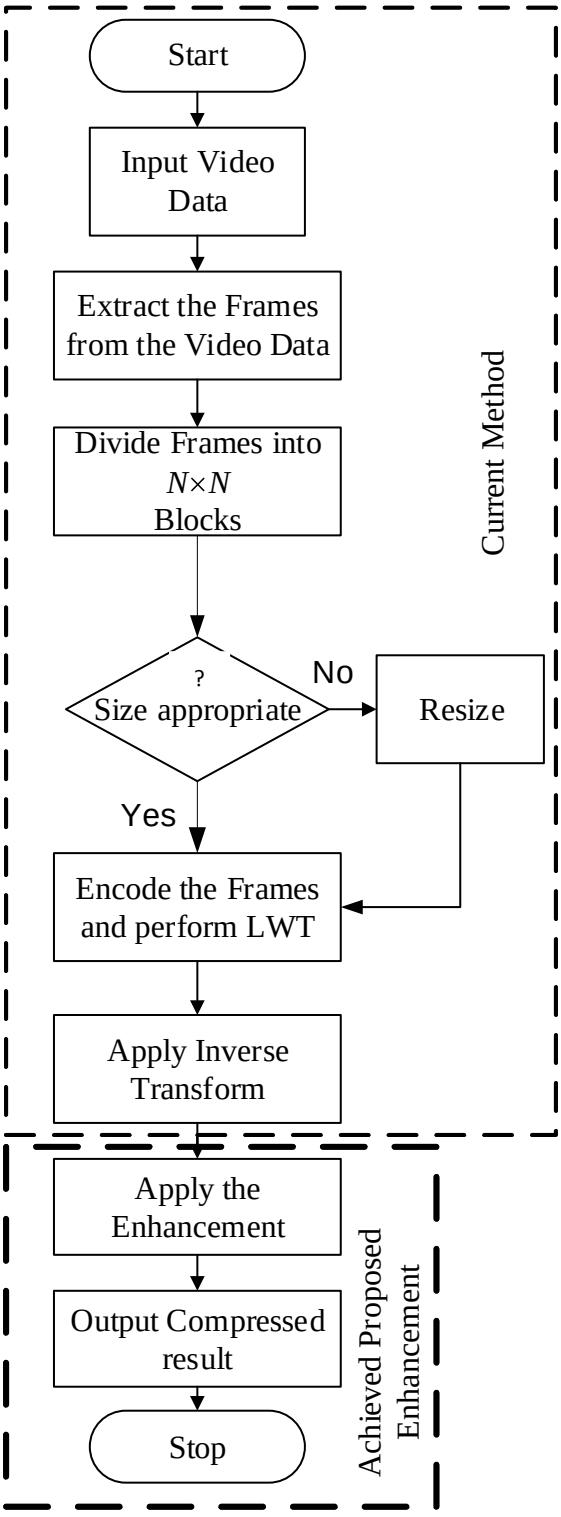


Figure 3.6: Flowchart of MATLAB Implementation of Improved LWT Video Image

In the LWT, a large number of memories are saved since the number of samples to be stored is the same as the input to each stage. The number of computations required is also reduced, since the approximation coefficients at one level can be derived from the detail already

computed from some of the input samples. The process of implementing the LWT is as in DWT and DCT only at the encoding stage where a particular type of compression technique is applied.

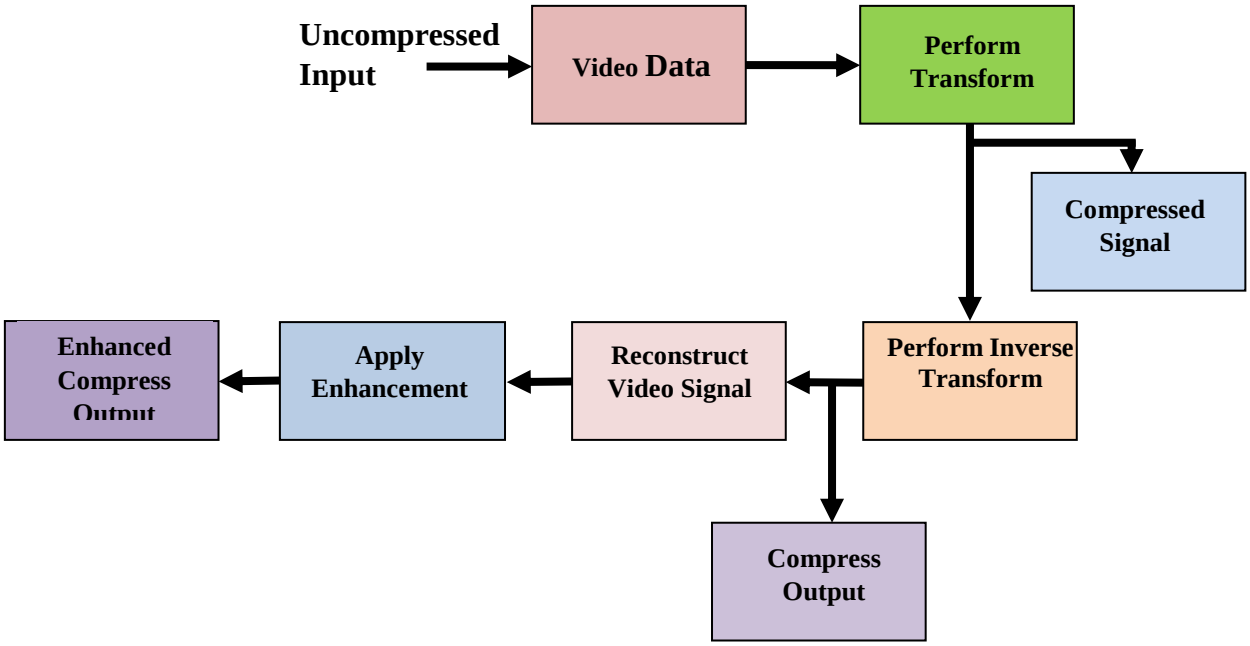


Figure 3.7: Block Diagram of the achieved result

In their implementation as shown inFigure 3.7, the sampled uncompressed video data served as input into the algorithm at the transformation stage of the respective DWT, DCT, andLWT to achieve video compression. The inverse transform was then implementedto regain back the signal and the enhanced video output was reconstructed to achieve efficient compression.

For analysis, the compression ratio metric was used to determine the performance of each technique implemented in Figure 3.6. This metric is expressed as (Wei, 2010):

$$Cr = \frac{size \;\; F(x,y)}{\sum_{i=1}^{N} Pixels} \tag{3.25}$$

The numerator of equation (3.25) is the total size of the image frame, the denominator is the total number of pixels and *Cr* is the compression ratio.

3.7 Compression Using Firefly Algorithm

The Firefly Optimization Algorithm (FOA) is inspired by the bioluminescence flash pattern characteristics of biological fireflies. One of the major characteristic behaviours of firefly which inspired its choice for this research is its attractiveness proportional to their brightness. In this research, the pixel content of all video frames represents potential brightness content of the fireflies, which was the value of the objective function (this is the video frame compressed or quantized). Two major issues of FOA were the variations of light intensity and formulation of the attractiveness. To address these challenges, this research assumed that the attractiveness of a firefly (pixels) was determined by the brightness (pixel intensity), which in turn was encoded in the form of an objective function. Since the FOA is a met heuristic optimization problem, therefore, the algorithm had to be specified as fitness function for which it should optimize. The research employed the sum of intra pixel distances (represented as an absolute value in equation (3.26)) as (Chang, 2007):

$$f(\mathrm{C}_1, \mathrm{C}_2, ..., \mathrm{C}_\mathrm{K}) = \sum_{i=1}^{K} \left(\sum_{X_j \in C_i} \left\| Y_i - X_j \right\| \right) \tag{3.26}$$

Equation (3.26) is used as the objective (fitness) function to be optimized. Since the compression involves minimization of the luminance content of the video frames, equation (3.26) is transformed into a minimization objective function as follows:

$$f(\mathrm{C}_1, \mathrm{C}_2, ..., \mathrm{C}_\mathrm{K}) = \min \left(\sum_{i=1}^{K} \left(\sum_{X_j \in C_i} \left\| Y_i - X_j \right\| \right) \right) \tag{3.27}$$

where:

$$\left\| Y_i - X_j \right\| = \sqrt{ Y_1 - X_1 \ ^2 + ... + \ Y_i - X_j \ ^2 } \tag{3.28}$$

is the Euclidean distance between each pixel vector in the frame clusters and thecentresof the clusters.

N_p is the population of fireflies (which denotes the total number of pixels in the video frame)

K denotes the number of clusters (which represents the colours in the colour map)

Y_i denotes the coordinates of pixel p in total pixel.

X_j denotes the centroid of cluster K (representing one colour triple in the colour map)

This is calculated and summed up. The constraint of the frame clusters is formulated as $C_i \ 1 \le i \le K$ such that each N pixel vectors $X_j \ 1 \le j \le N$ are clustered on the basis of distance from each of these cluster centres $Z_i \ 1 \le i \le K$. Based on these constraints, the FOA is used to determine the cluster centres which minimized equation (3.27). The simplified equation (3.28) shows that equation (3.27) determines the differences between original frame pixel colours and decoded video frames. Thus, in this research, each artificial firefly represents a K cluster centre. Therefore, the FOA represents a K dimensional vector, which includes one of the colours to be replaced with some better similar colours.

The pseudo code implementation of FOA based video frame compression is given as follows:

```
start
Initialize parameters of the FFA
Randomly generate the initial population of the FFA (based on
the frame pixel)
for each firefly(i)
    evaluate fitness using equation (3.27)
    for each pixel(j)
        calculate the Euclidean function
        Determine the cluster centres
        evaluate fitness using equation (3.27)
    end
end
determine best fitness
update solution
```

Figure 3.8 is the flowchart implementation of this pseudo code.

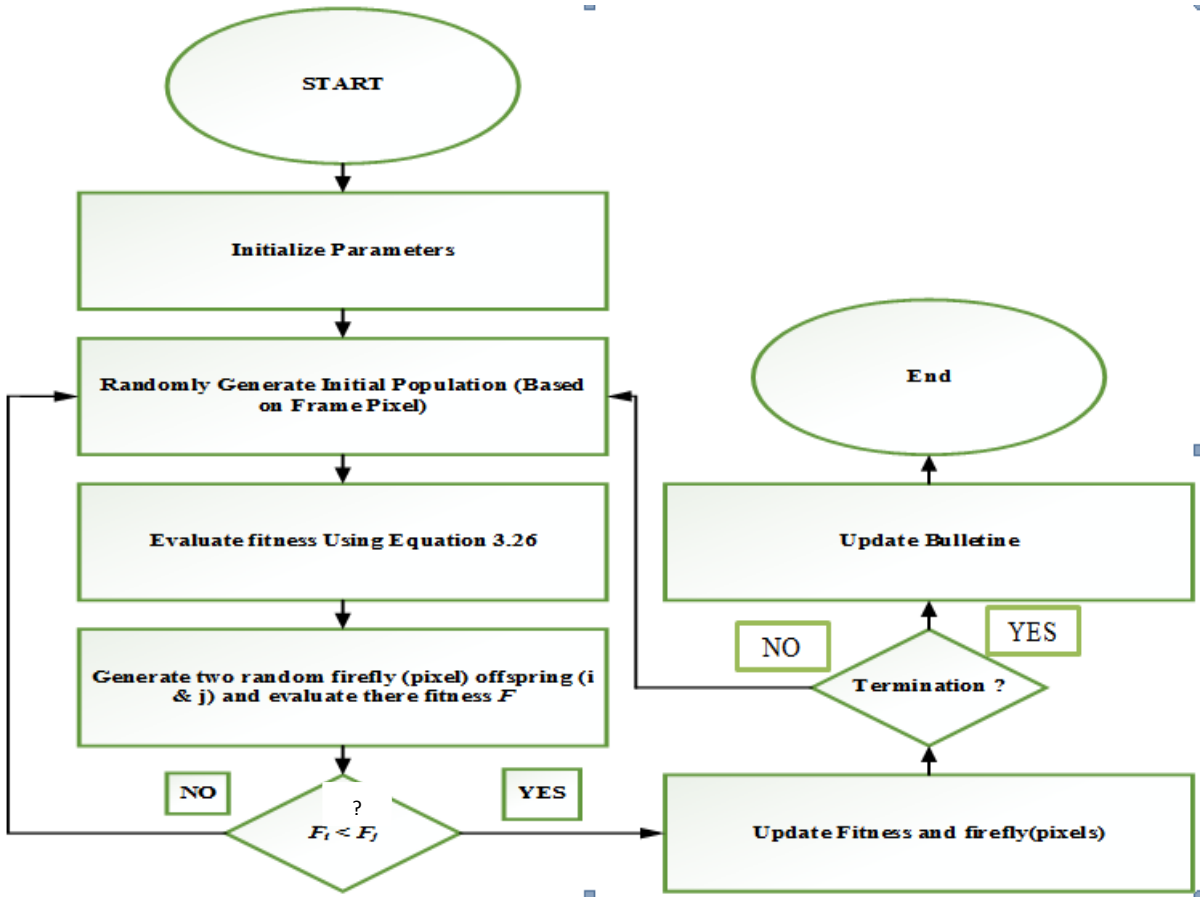


Figure 3.8: Flow Chart of Compression Using Firefly Algorithm

Figure 3.8 is the flowchart implementation of the Firefly Optimization Algorithm (FOA) compression technique. The implementation started by initialising the parameters used, followed by randomly generating the initial population based on frame pixels, and then followed by evaluating the fitness using equation (3.26). Two random firefly (pixel)offspring (i and j) are generated and their fitness F evaluated. The first decision box is used to initialise the fitness functions F1 and F2. For F1 less than F2, the process is "Yes" and the process goes ahead and updates the fitness and the firefly (pixels). If otherwise, the process is "No" then it proceedsback to randomly generate another population based on frame pixels. The second decision box decides on the satisfaction of the generated update fitness and the firefly

pixels. In the second decision box, if "Yes" the process moves to update the bulletin and ends the entire process. But if "No" then it goes back to generate another population based on frame pixels and starts all over again as in Appendix E. The difference between Figure 3.8 and 3.9 is the modification made in latter indicated with a hidden detail. At this point the pixel intensity and determination of their distance apart are evaluated. The decision box then decides if the minimum distance is really achieved, then the process goes ahead to determine the pixel attractiveness and its intensity as in equations (3.29) through (3.31). But if otherwise, the process retains the previous distance and moves back again to evaluate the pixel intensity and also determines their newdistance apart as in Appendix F.

All the relevant mathematical models describing the implementation of the FOA are covered in chapter two. The detailed MATLAB script used for its implementation can be found in Appendix F. As stated earlier, the major problem with FOA is the ease at which the algorithm falls into local minima, which is usually caused by the imbalance between exploration and exploitation. Since the FOA works on the principle of attractiveness and light intensity, when the distance between two fireflies are far apart, the attractiveness reduced thereby reducing the light intensity. In this research, the idea of attractiveness and intensity are interpreted in terms of pixel and luminous intensity of an image. In order to reduce the effect of pixel intensity with larger distance part, this research employed the following mathematical equation.

$$r = \sqrt{\frac{I_s}{I_0}} \tag{3.29}$$

where,

r is the pixel intensity apart obtained from an inverse square law relationship.

I_s is the intensity of the pixel source

I_0 is the original pixel intensity.

Thus, the attractiveness and light intensity of the FOA is modified as follows.

$$I = I_0 e^{-\gamma\left(\sqrt{\frac{I_s}{I_0}}\right)} \tag{3.30}$$

$$\beta = \beta_0 e^{-\gamma\left(\frac{I_s}{I_0}\right)} \tag{3.31}$$

Equations(3.30) and (3.31) are employed in order to reduce the effect of higher distance which occurs between some pixels of the image. Every parameter of these equations is as discussed in subchapter 2.2.22 of chapter two. The flow chart of the modified FOA is given in Figure 3.9. The impact of the modification is to bring our own contributions in improving the standard technique in terms of the PSNR and the modification by including the root mean square in the standard equationsin order to reduce the effect of pixel intensity with larger distance part. When the images samples were subjected to this modified firefly compression technique, a significant amount of improvement was achieved.

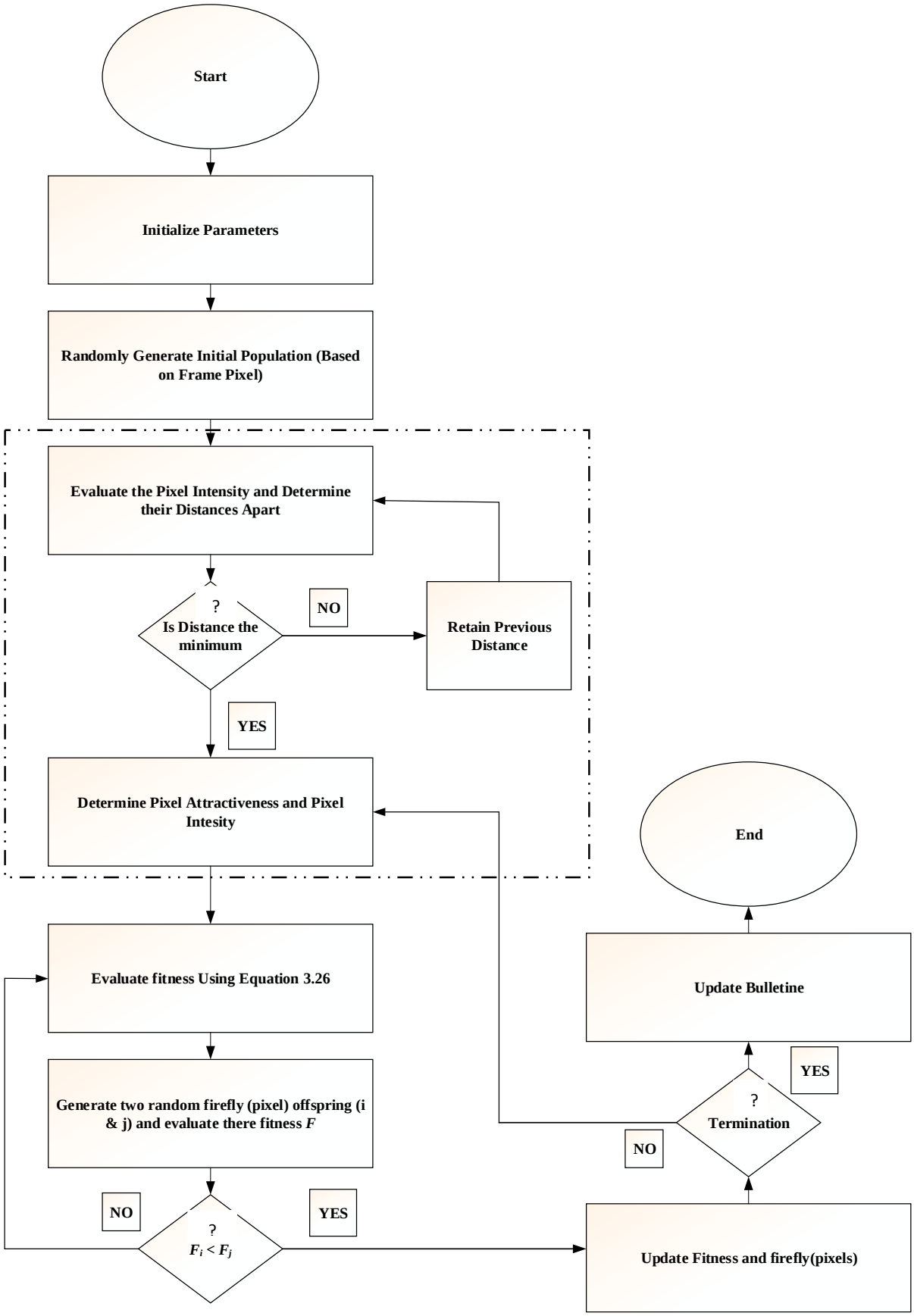


Figure 3.9 Flow Chart Implementation of Modified FOA

CHAPTER FOUR

RESULTS AND DISCUSSION

4.1 Introduction

In this section, the performance of the techniques employed is evaluated. Simulation results are presented in order to show the efficiency of the technique used and also to shed more light on the results obtained in this research work.

4.2 Results

The acquired video samples were read into the MATLAB toolbox R2014b, converted into image frames, and the resulting image frames were resized appropriately. Elimination of the hue and saturation (black and white grey level) was performed and the results are presented in Figure 4.1.



Figure 4.1: Luminance Intensity of Black and White Video Frames

In order to determine the efficiency of the luminance enhancement technique, the pixel values of the hue and saturation free image in Figure 4.1 were randomly varied by the introduction of noise, then filtered to smoothen the output as shown in appendix A. The filtered image was passed through the luminance enhancement technique and the result is shown Figure 4.2.

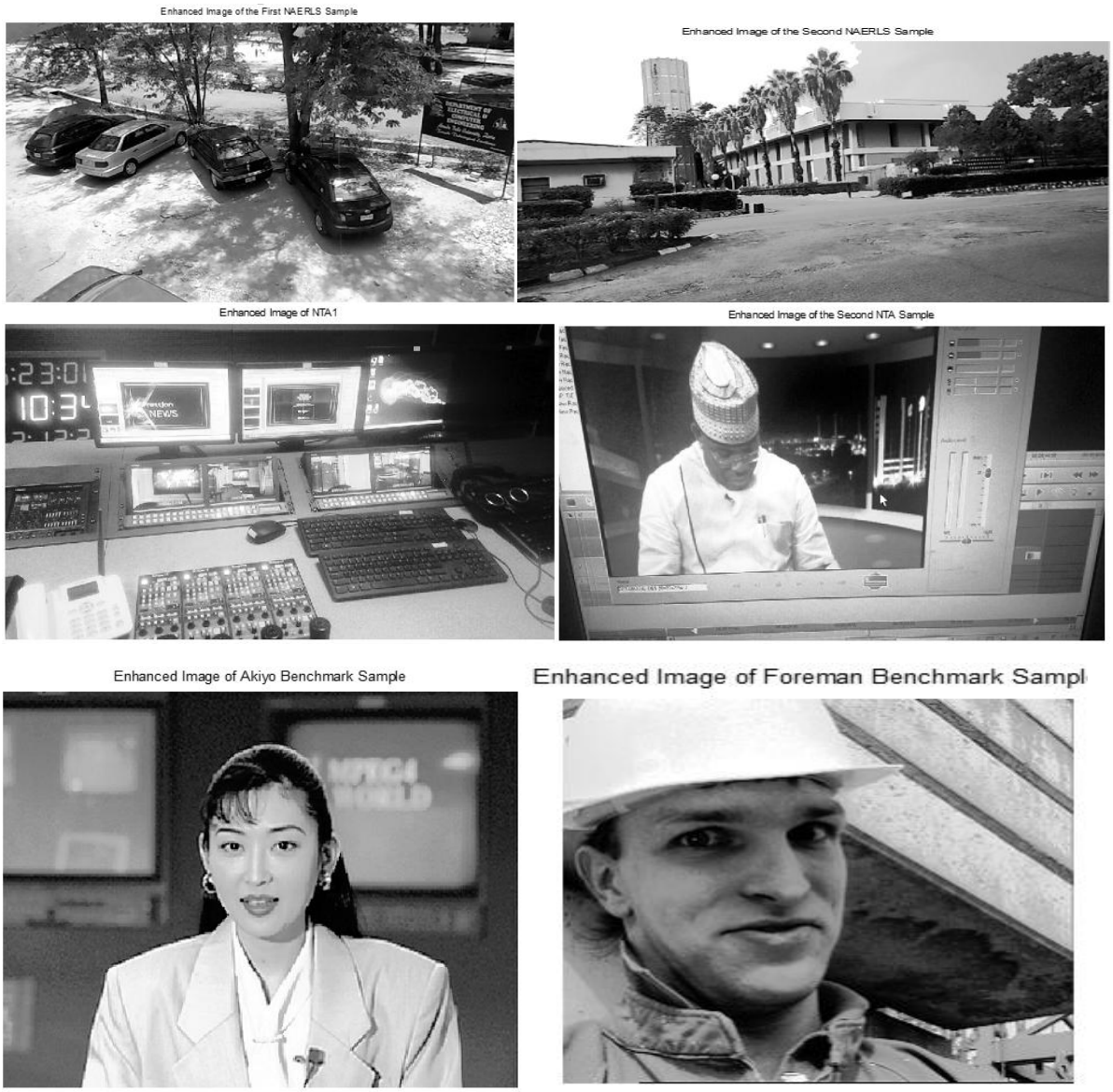


Figure 4.2: Luminance Intensity of Black and White Video Frames after Enhancement

The impact of the technique (luminance enhancement) used clearly shows that, each frame in Figure 4.2 appears to be brighter and sharper when compared with their equivalent frame in Figure 4.1. In order to provide a clear justification for this statement, the histogram of each of the sampled video frames and the benchmark video frames were generated from

theMATLAB stimulation environment using equation 2.4 as shown in Figure 4.3.This is the vectors representation of the image pixels at different intensities using histogram distribution as in first part of appendix A.

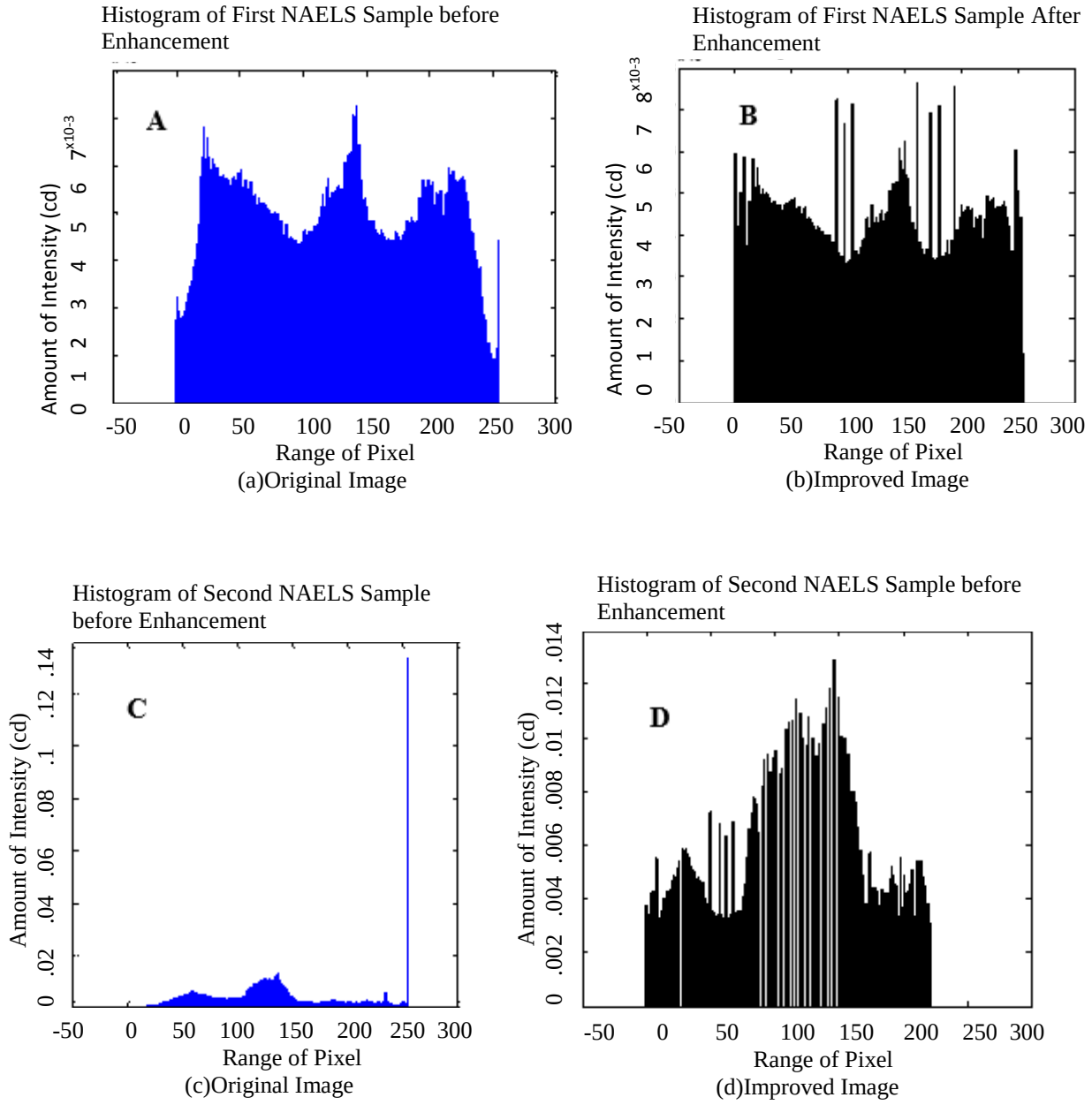


Figure 4.3: Histograms of Original and Improved NAERLS Sample Images

Figure 4.3(A&C) shows histogram representation of the NAERLS1 and NAERLS2 sampled image before the application of the luminance enhancement technique while Figure 4.3(B&D) shows the histogram of the frames after enhancement. It should be observed that, the

luminous intensity of the NAERLS1 frame (Figure 4.3-A) has been significantly improved and distributed across the pixel values in Figure 4.3(B). Similarly, the luminous intensity of NAERLS2 sample frame given in Figure 4.3(C) is improved significantly across the pixel values of the frame in Figure 4.3(D). Note this is the improvement of the source original image in terms of the PSNR as can be seen in the last part of appendix A

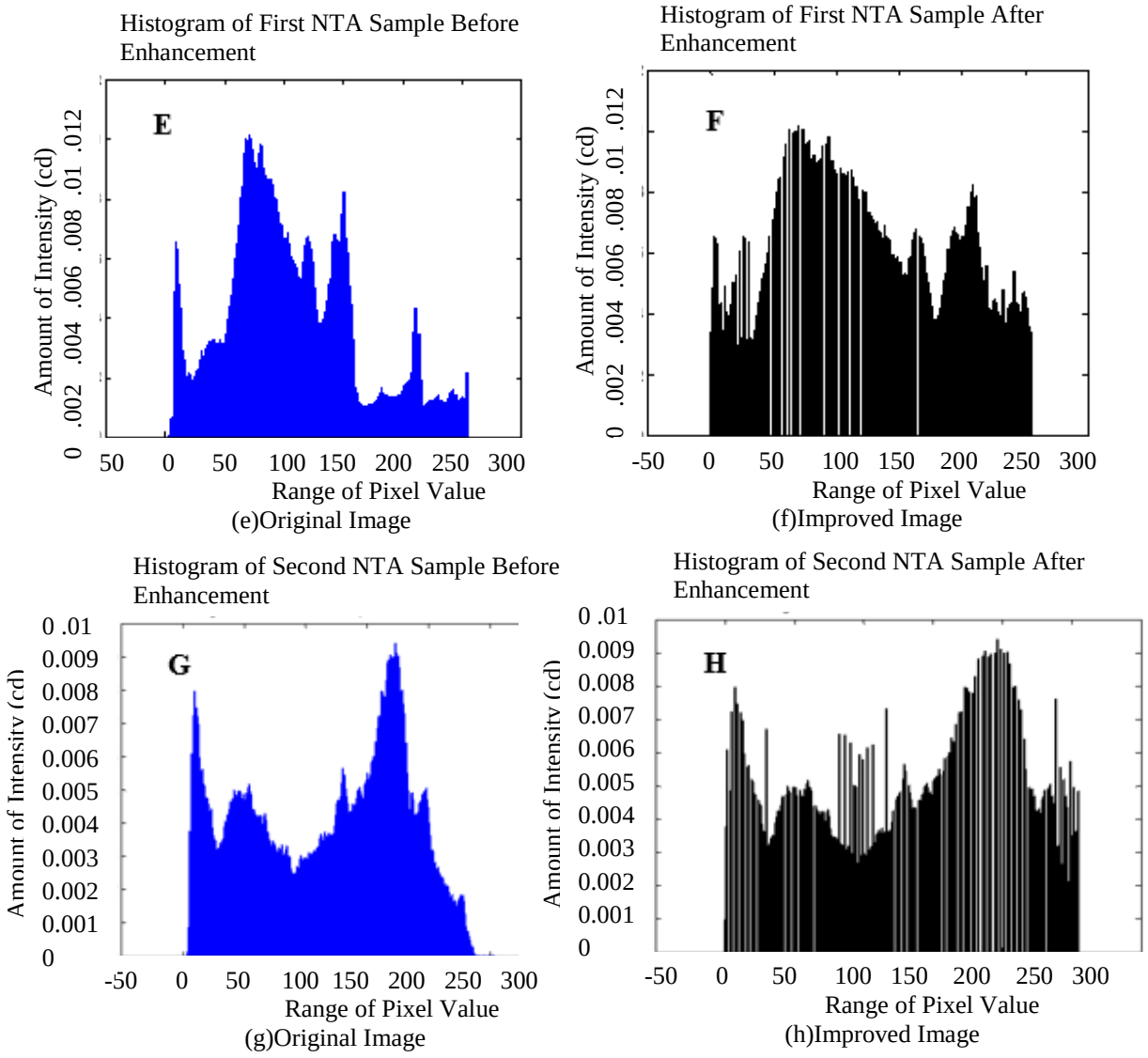


Figure4.4: Histograms of Original and Improved NTA Sample Images

Figure 4.4(E) and Figure 4.4(F) show the respective histograms of the NTA1 sample video frames before and after luminance enhancement. Also, Figure 4.4(G) and Figure 4.4(H) show the histogram of the NTA2 sample video frame before and after the luminance enhancement.

It is observed from both Figure 4.4(F) and Figure 4.4(H) that the histograms of the frames from both cases have an improved luminous intensity with a much better distribution across the pixels values of each frame. This is an indication of how much of enhancement is done on the frames.

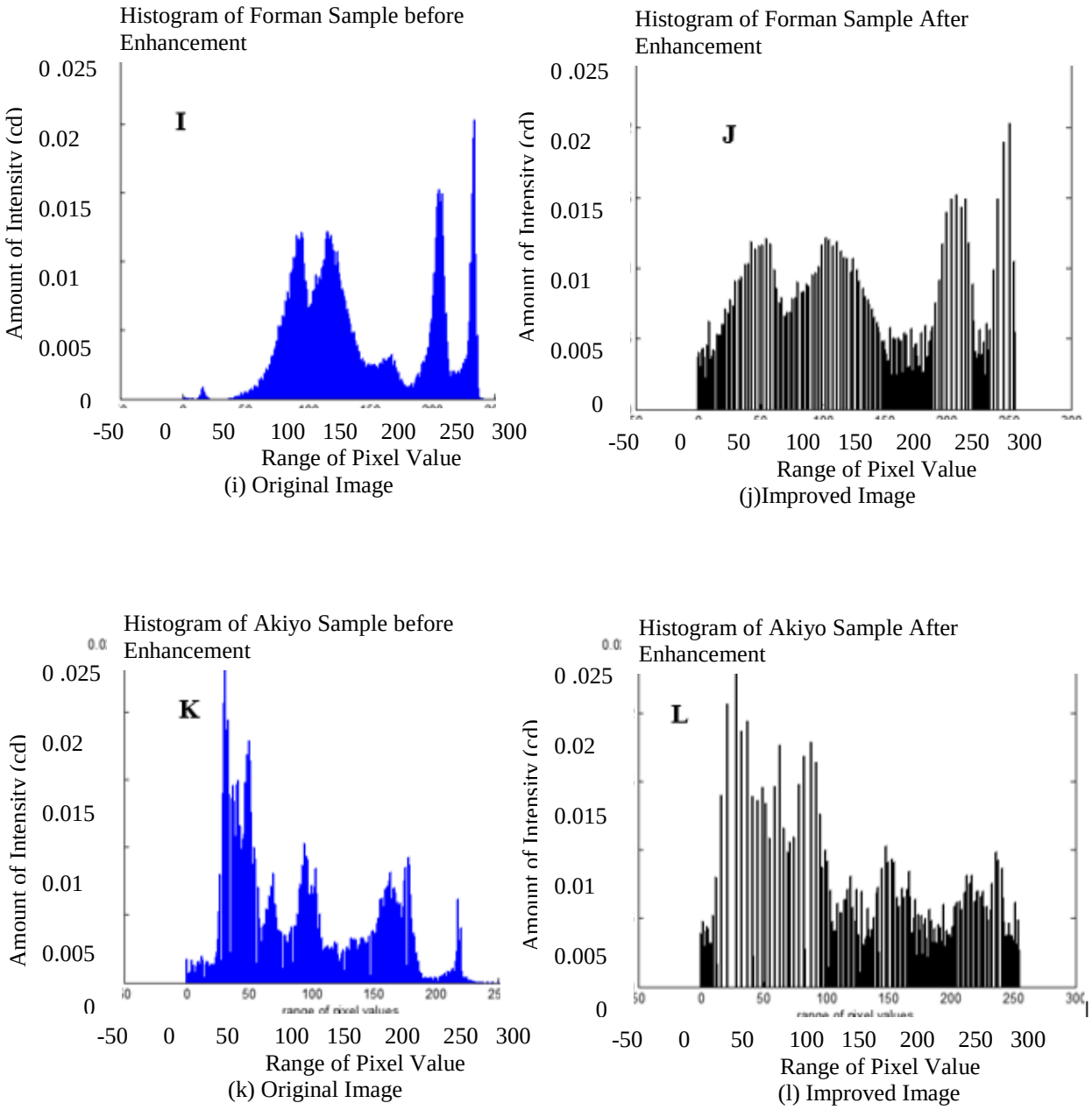
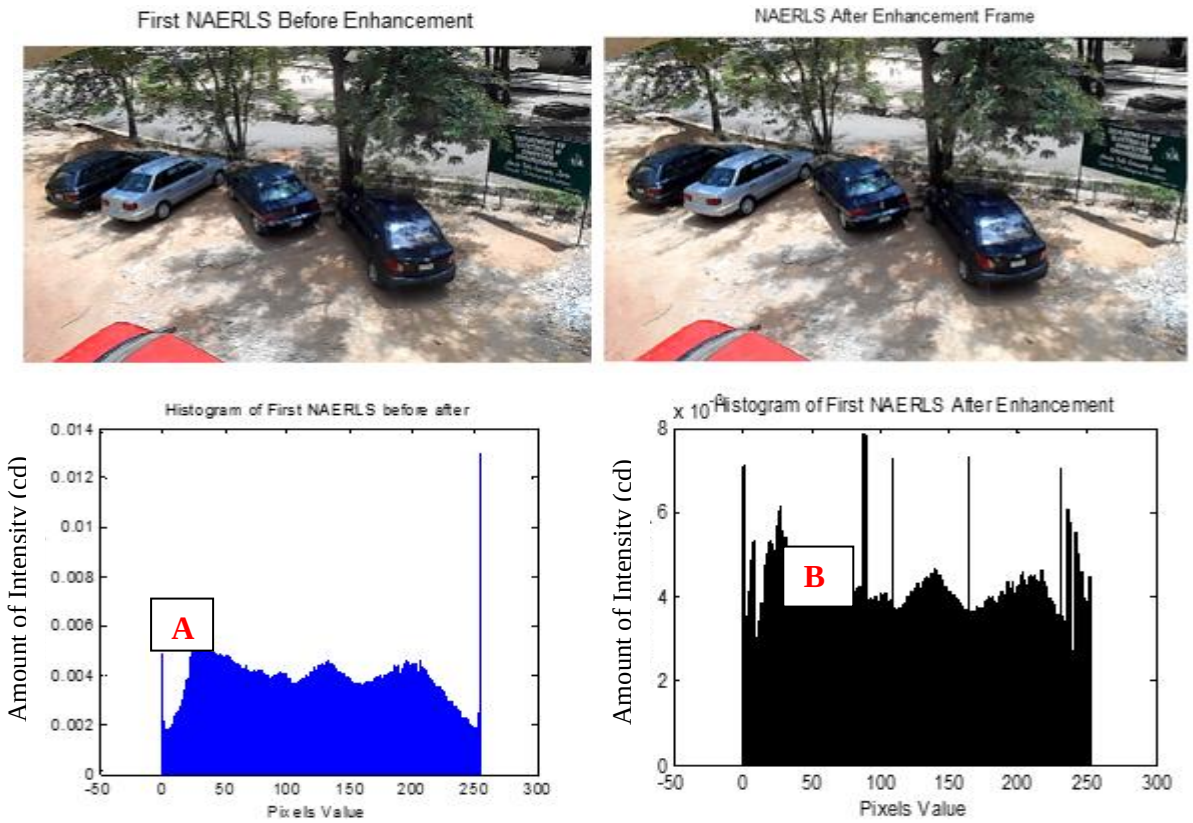


Figure 4.5: Histograms of Original and Improved Benchmark Sample Images

Figure 4.5(I) and Figure 4.5(J) show the respective histograms of the *Foreman.avi* benchmark video frames before and after luminance enhancement, while Figure 4.5(K) and

Figure 4.5(L) show the histograms of the *Akiyo.avi* benchmark video frame. In both figures, it is observed that the intensity of the frames has improved throughout the pixels values with a much better distribution. This also, is an indication of how much enhancement is done on the benchmark frames.

In order to evaluate the robustness of the luminance enhancement techniques presented in this report, the technique is applied to the coloured frame of all the six sampled videos. This is to determine how much of improvement can be done on the frames containing hue, saturation and luminance all together using the brightness (luminance) enhancement technique. Figure 4.6 shows the original *NAERLS1.avi* sample video frame and the enhanced coloured frames.

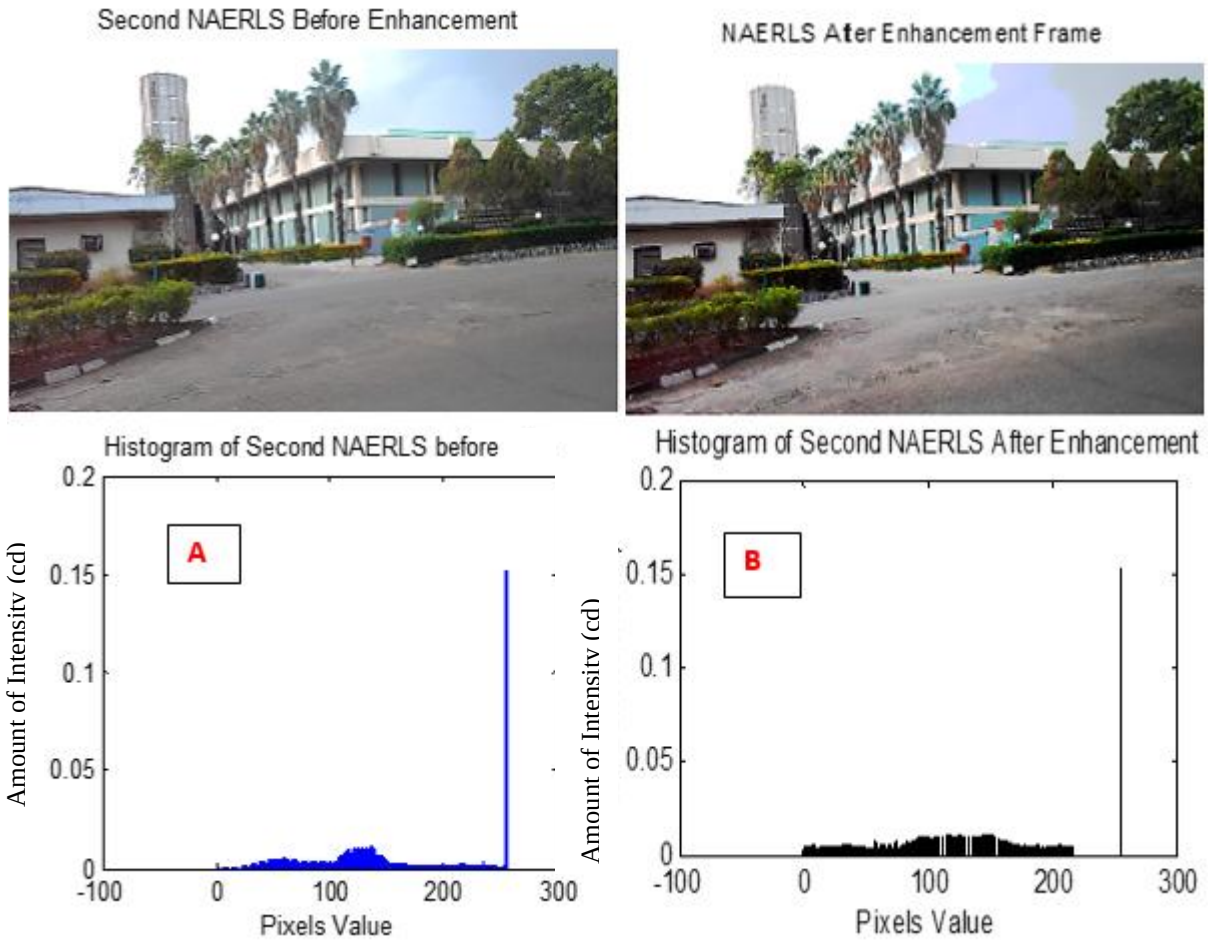


Original Image Improved ImageFigure 4.6: *Figure 4.6:*

Coloured Images and Histograms of Original and Improved NAERLS1 Sample

From Figure 4.6, it is observed that the histogram of the original sample frame (Figure 4.6 A) has been significantly improved (Figure 4.6 B) after the application of the enhancement technique.

Figure 4.7 shows the original *NAERLS2.avi* sample video frame and the enhanced coloured frames.



Original Image Improved Image

Figure 4.7: Coloured Images and Histograms of Original and Improved NAERLS2 Sample

Careful observation of Figure 4.7 shows that, the histogram of the original video frame (Figure 4.7A) has been improved (Figure 4.7B) as a form of enhancement.

Also, the enhancement technique was applied to the coloured sample frames obtained from the Nigerian Television Authority. The result obtained for the first sampled frame (*NTA1.avi*) is given in the Figure 4.8.

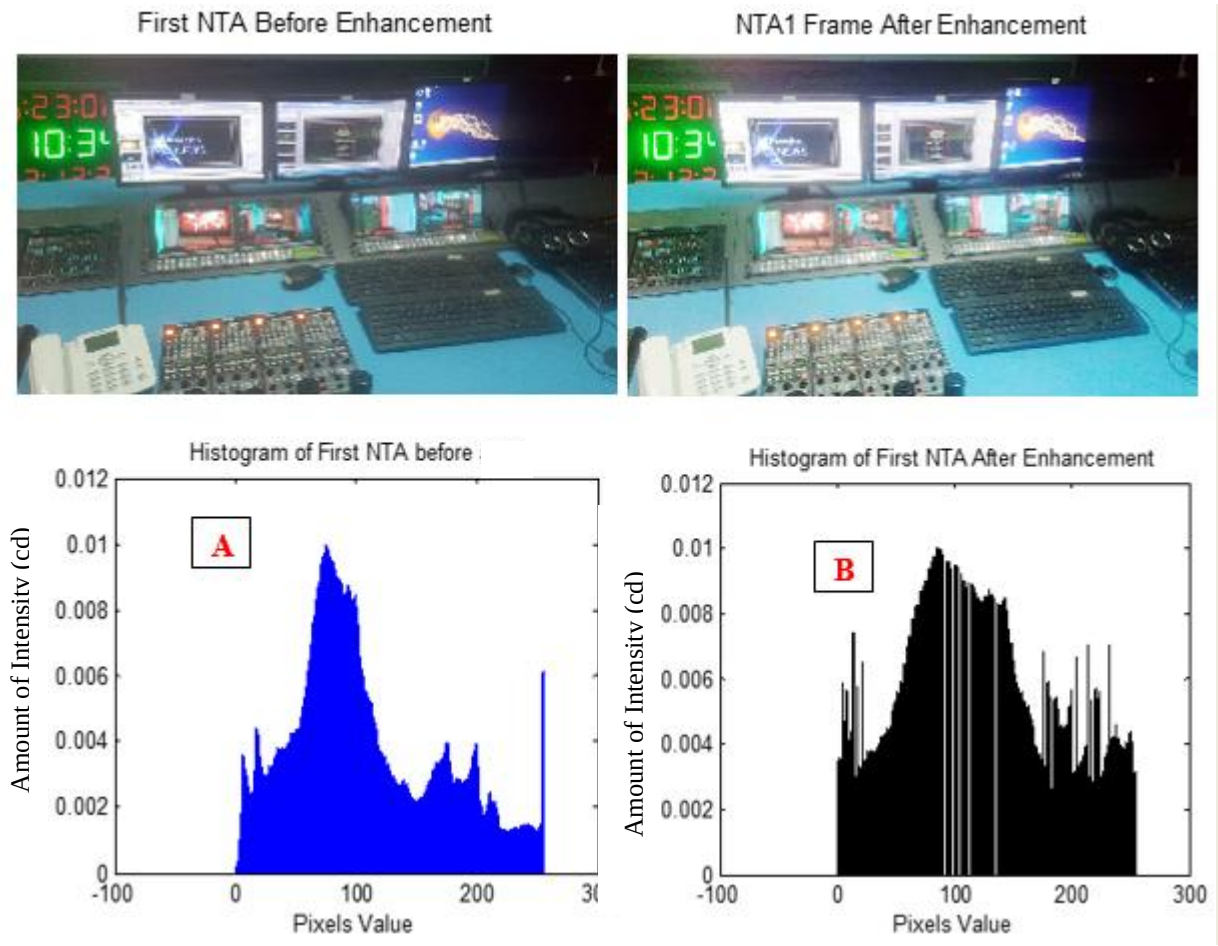


Figure 4.8: Coloured Images and Histograms of Original and Improved NTA1 Sample

Figure 4.8(B) shows the improved version of Figure 4.8(A) after enhancement. Similarly, The result obtained for the first sampled frame (*NTA2.avi*) is given in the Figure 4.9.

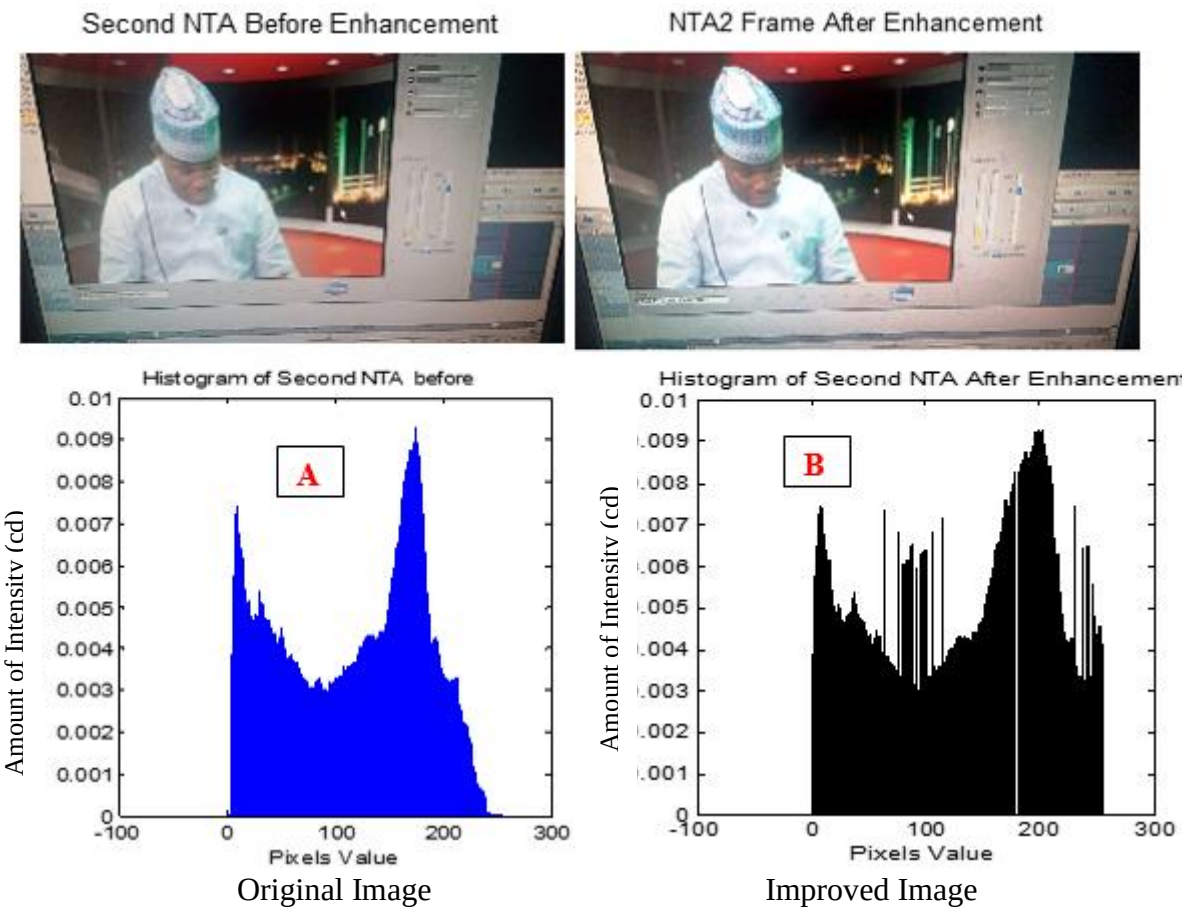
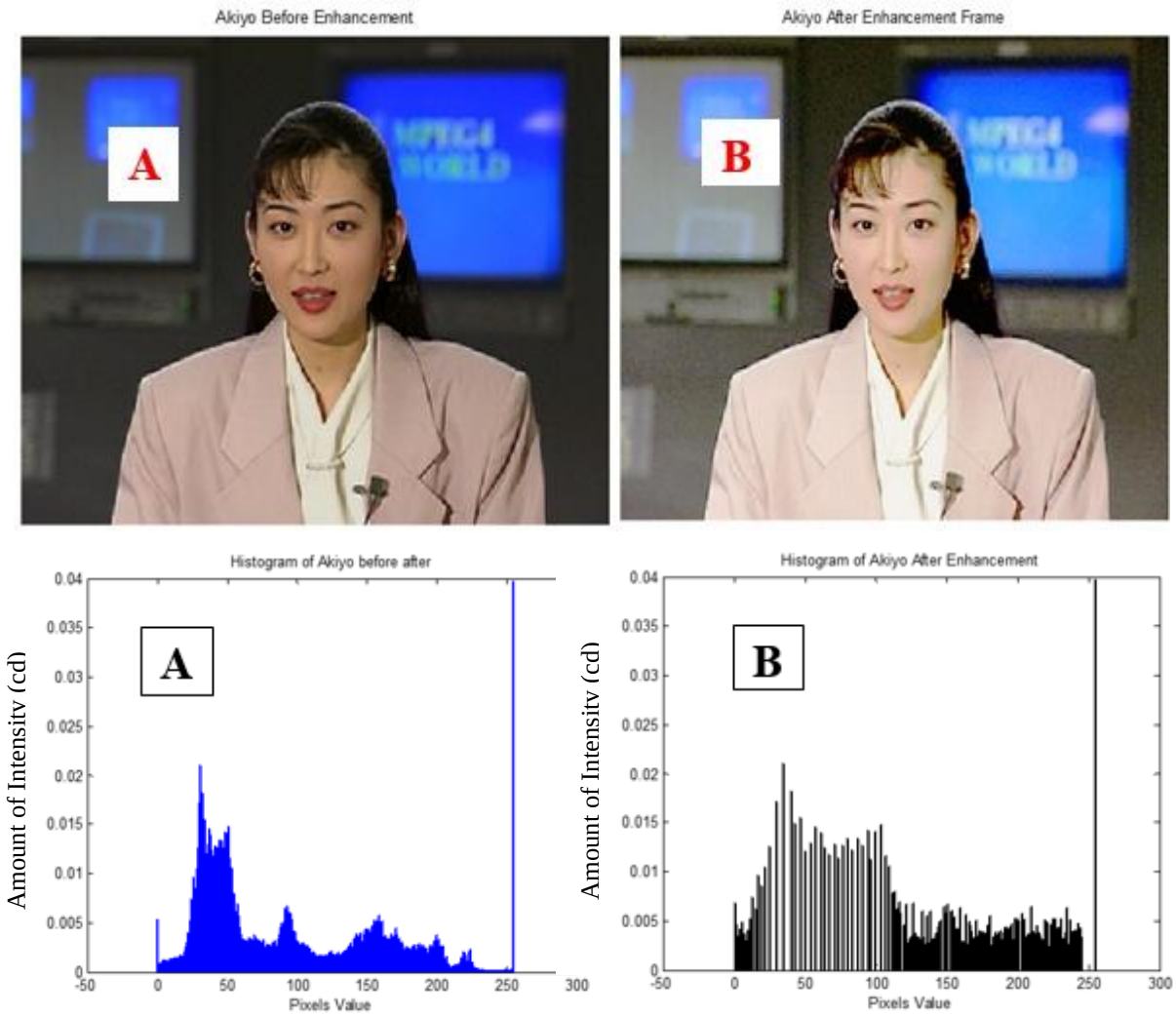


Figure 4.9: Coloured Images and Histograms of Original and Improved NTA2 Sample

From Figure 4.9, it is observed that the histogram of the original NTA2 sample video frame of Figure 4.9(A) shows a much better improvement in Figure 4.9(B) after the application of the enhancement technique.

The enhancement technique was also applied to the coloured sampled frame of the two benchmark videos. The original and enhanced frames of Akiyo benchmark video, with their corresponding histograms are given in Figure 4.10.



Original Image Improved Image

Figure 4.10: Coloured Images and Histograms of Original and Improved sample of Akiyo

From Figure 4.10, it is observed that the histogram of the original *Akiyo.av* benchmark sample video frame of Figure 4.10(A) shows a much better improvement in Figure 4.10(B) after the application of the enhancement technique as in second part of appendix A..

The enhancement technique was also applied to the coloured sampled frame of the Foreman benchmark video. The original and enhanced frames of Foreman benchmark video, with their corresponding histograms are given in Figure 4.11.

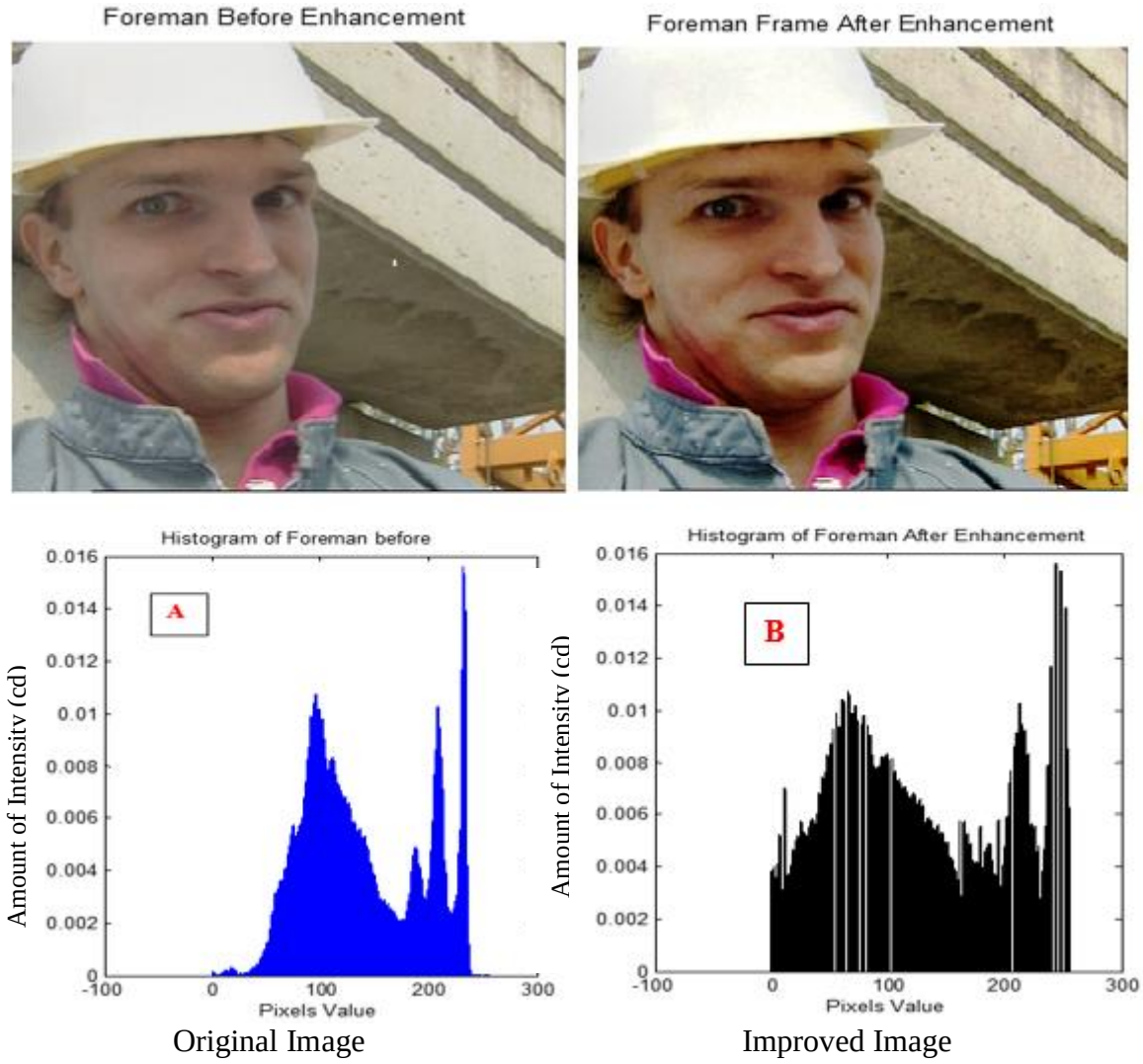


Figure 4.11: Coloured Frame of Foreman before and After Enhancement

Also, Figure 4.11 shows the *Foreman.avi* benchmark video frame before and after the application of the enhancement technique. The histogram of the Foreman frame after the application of the enhancement technique (Figure 4.11-B) shows a better distribution and improvement.

4.3 Performance Evaluation using Peak Signal-to-Noise Ratio

In order to further evaluate the performance of this method, the Peak Signal-to-Noise Ratio (PSNR) is used to measure the quality of reconstructed sample frames of video signals.

Table 4.1:Simulation Result of Performance Evaluation using PSNR on Black and White Frames

S/N	SAMPLE	SIZE	FRAME	PSNR
1	NAERLS1.avi	18.1Mb	157	31.95Db
2	NAERLS2.avi	10.3Mb	155	22.30dB
3	NTA1.avi	9.6Mb	152	17.71dB
4	NTA2.avi	11.2Mb	200	23.17dB
5	Akiyo.avi	11Mb	300	15.06dB
6	Foreman.avi	7.25Mb	100	19.17dB

The PSNR results of the black and white sampled video frame signals and coloured sampled video frame signals given in Table 4.1 were obtained from Figure 4.1 and Figure 4.2, respectively. In order to make a justifiable conclusion, the PSNR of the brightness enhancement technique on the coloured sample frames were determined as shown in Table 4.2. Note all tables in this section were computed from MATHLAB simulation environment R 2014a and effected as in the second part of appendix A.

Table.4.2 Simulation Result of Performance Evaluation using PSNR on Coloured Frames.

S/N	SAMPLE	SIZE	FRAME	PSNR
1	NAERLS1.avi	18.1Mb	157	36.45dB
2	NAERLS2.avi	10.3Mb	155	26.65dB
3	NTA1.avi	9.6Mb	152	24.45dB
4	NTA2.avi	11.2Mb	200	28.90dB
5	Akiyo.avi	11.0Mb	300	21.19dB
6	Foreman.avi	7.25Mb	100	28.06dB

The PSNR decibel values of Table 4.2 for the coloured frames show improvement when compared with their corresponding values in Table 4.1 for the hue and saturation free (black and white) frames.

This indicates that, the brightness enhancement technique is more effective and efficient for coloured video than hue and saturation free frames. The bar chart showing the amount of improvement on the coloured frames over the black and white frames is given in Figure 4.12.

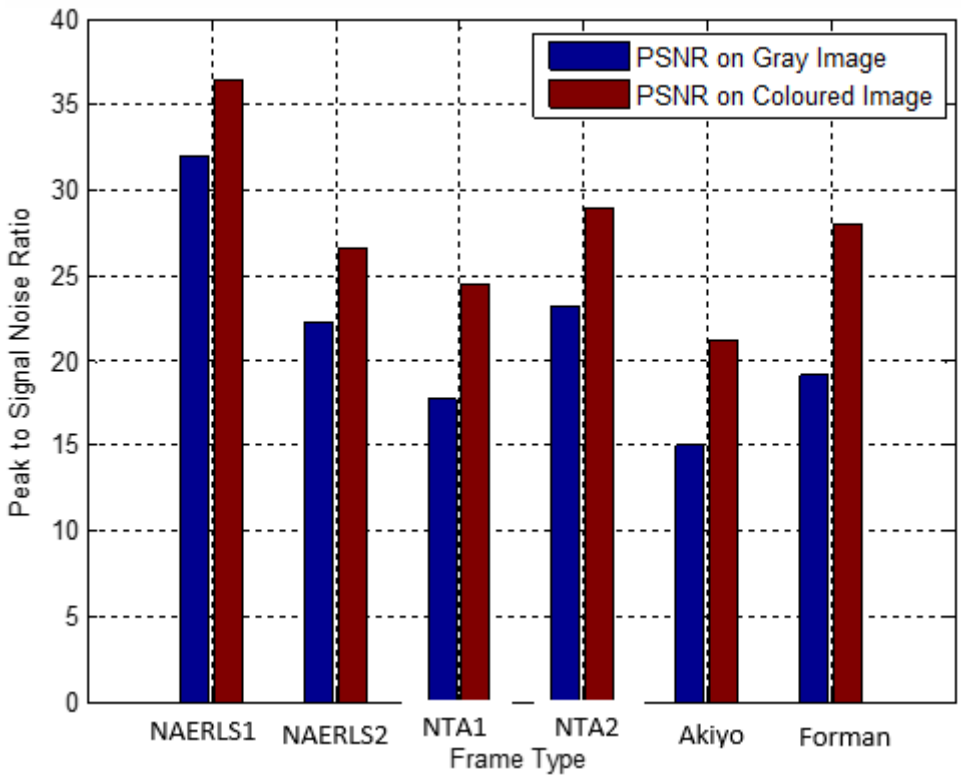


Figure 4.12: Brightness Enhancement Comparison on Greyscale and Coloured Frames

Figure 4.12 shows the amount of brightness enhancement on the grey scale frames and the coloured frames. Both *NAERLS1.avi* and *NAERLS2.avi* coloured sample frames show a PSNR percentage improvement of 12.45% and 16.32% over the grey scale sampled images. The *NTA1.avi* and *NTA2.avi* coloured frames shows a PSNR percentage improvement of 27.57% and 19.83%, respectively. Similarly, the respective PSNR percentage improvement of 28.93% and 31.68% were achieved for *Akiyo.avi* and *Forman.avi*benchmark video frames. This percentage improvement over the grey scale frames indicates the efficiency of this

brightness enhancement method on coloured images. This is justifying the aim ofthis research.

4.4 Compression Analysis

The performance of the enhanced compression technique is evaluated in this subsection. Since the essence of compression is to reduce the size of the video data for easy transmission, the performance of the enhanced compression technique is evaluated using sample size (bytes), compression ratio, and peak signal-to-noise ratio (dB). The standard DCT, DWT, and LWT were developed, implemented, and their resultant compressed outputs were enhanced using the developed brightness enhancement model. The final outputs from these three techniques were used as the benchmark for evaluationagainst that of the Firefly Optimization Algorithm (FOA), which was also enhanced using the proposed brightness enhancement model. The resulting size of the sample frames after compression is given in Table 4.3.

Table 4.3: Simulation Result of Performance Comparison of the Sample Size after Compression

Sample	**Original Size**	**DWT**	**DCT**	**LWT**	**FOA**
NAERLS1.avi	18.1Mb	12.38Mb	7.10Mb	6.41Mb	6.10Mb
NAERLS2.avi	10.3Mb	8.74Mb	6.93Mb	5.30Mb	4.33Mb
NTA1.avi	9.60Mb	7.36Mb	5.43Mb	5.06Mb	5.19Mb
NTA2.avi	11.2Mb	9.22Mb	7.60Mb	7.12Mb	6.85Mb
Akiyo.avi	11.0Mb	3.45Mb	2.81Mb	2.11Mb	2.10Mb
Forman.avi	7.25Mb	0.205Mb	0.165Mb	0.196Mb	0.171Mb

From Table 4.3, it is observed that a significant amount of size has been reduced using the compression technique. These compressed sizes were obtained before the application of the achieved brightness enhancement techniques. However, the pixel intensity is improved by histogram distribution to achieve better image representation. For easy interpretation and explanation, the bar chart showing the percentage reduction in image size is given in Figure

4.13. Note that the above computations were achieved as in equation (3.25) and last part of appendix D.

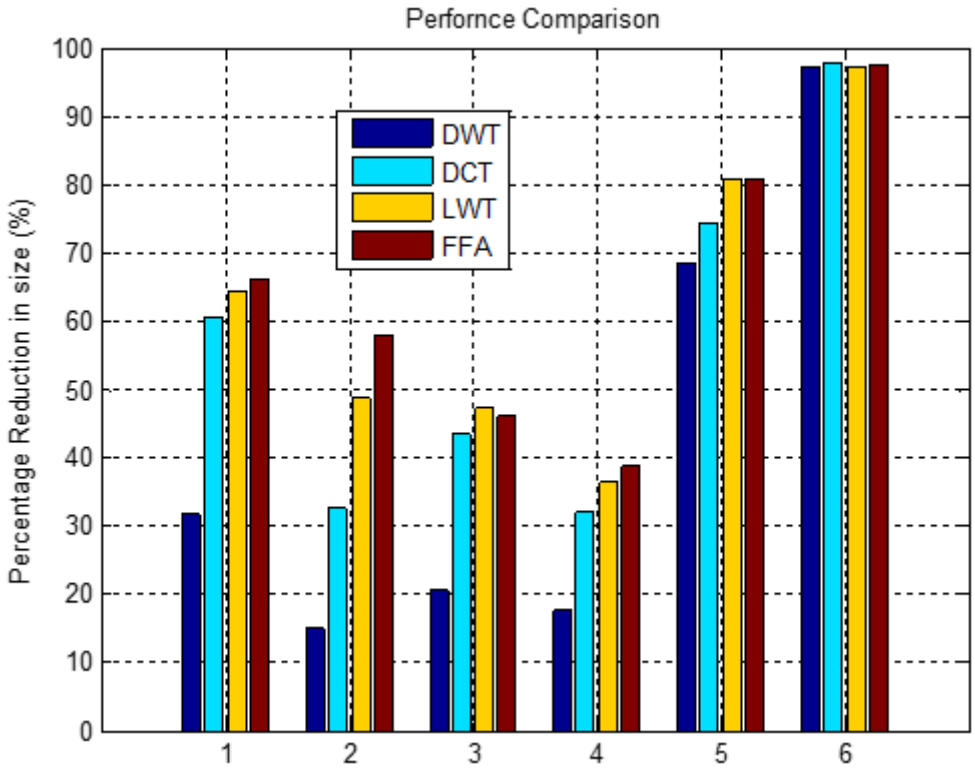


Figure 4.13: Percentage Reduction in Size

It isobserved from Figure 4.13 that the enhanced LWT compresses the video signal better when compared with the DCT and DWT. About 64%, 59%, 58%, 38%, 80% and 98% reduction in *NAERLS1.avi, NAERLS2.avi, NTA1.avi, NTA2.avi, Akiyo.avi* and *Forman.avi* video data, respectively was obtained using the LWT. Also, respective values of 60%, 32% 43% 31%, 75% and 97% were obtained using the DCT, while the values obtained using DWT were 31%, 15%, 22%, 18%, 69% and 97%, respectively. However, these results were compared with the results obtained using the developed FOA compression technique and it was found that FOA compression method attained a reduction of 66.4%, 58%, 46%, 39%, 81% and 97% for NAERLS1.avi, NAERLS2.avi, and NTA1.avi, NTA2.avi, Akiyo.avi and

Forman.avi video data respectively. This FOA improvement over other techniques is calculated using the following equation:

$$\frac{Technique\quad Size\quad -\quad FOA\quad Size}{Technique\quad Size} x\quad 100\,\% \qquad (4.1)$$

where, technique size can be original sample size, or any of the compressed sample size obtained from DWT, DCT and LWT compression techniques, respectively.

The results of the compression ratio analysis of the respective compression techniques when subjected to the brightness enhancement model are given in Table 4.4.

Table 4.4:Simulation Result of Compression Ratio Analysis

Sample	**DWT**	**DCT**	**LWT**	**FOA**
NAERLS1.avi	13.698	15.201	22.983	26.356
NAERLS2.avi	11.146	11.204	11.656	14.811
NTA1.avi	10.939	11.127	17.901	16.222
NTA2.avi	11.493	12.841	14.710	20.100
Akiyo.avi	10.620	11.541	18.424	21.001
Forman.avi	11.475	11.868	14.081	14.750

As the compression ratio increases the image quality degrades because of the artefacts (noise) resulting from the block based scheme. This implies that, high compression ratio is an indication of how much signal reduction was achieved due to compression. For the *NAERLS1.avi* sample video, it can be observed from Table 4.4 that, the LWT produced the best compression in comparison with DCT and DWT. The LWT produced a compression ratio improvement of 33.86% and 40.40% over the DCT and DWT techniques, respectively. On the other hand, the DCT produced a compression ratio improvement of 9.89% over DWT. For the *NAERLS2.avi* sampled video, the LWT technique produced an improvement of 3.88% and 4.38% over the DCT and DWT, respectively, while the DCT produced an

improvement of 0.52% over DWT. For the NTA sampled videos, the LWT achieved compression improvement of 37.84% and 38.89% over DCT and DWT, respectively for the NTA1 sample. Similarly, for the NTA2 sample, the LWT produced an improvement of 12.70% and 21.87% over DCT and DWT, respectively. In both samples, the DCT performed better than DWT with 1.69% and 10.50% on NTA1 and NTA2 samples, respectively. For the benchmark video frames, the LWT also produced better compression with 37.36% and 42.36% improvement over DCT and DWT, respectively for the *Akiyo.avi* benchmark video. Also, for the *Forman.avi* benchmark video, the LWT produced improvement of 15.72% and 18.51% over the DCT and DWT, respectively. The DCT performed better than DWT with 7.98% and 3.31% on *Akiyo.avi* and *Forman.avi* benchmark videos respectively. However, in all this technique, the developed FOA based compression produced the best result in comparison with the LWT, DCT and the DWT techniques. For the test video frame, the FOA performed better than LWT with a percentage improvement of 12.5%, 21.3%, -10.35% and 12.27% for the NAERLS1.avi, NAERLS2.avi, NTA1.avi and NTA2.avi respectively, while a percentage improvement of 12.27% and 4.54% was also obtained for the Akiyo.avi and forman.avi video frames. Also, the FOA based compression method performed better, than the DCT on the test video frame with a percentage improvement of 41.34%, 24.35%, 31.41% and 36.11% for the *NAERLS1.avi, NAERLS2.avi, NTA1.avi* and *NTA2.avi*, respectively. While an improvement of 45.05% and 19.54% was also obtained for the *Akiyo.avi* and Forman.avi video frame. Similarly, in comparison with the DWT technique, a percentage improvement of 46.91%, 24.75%, 32.57% and 42.82% was obtained for the *NAERLS1.avi, NAERLS2.avi, NTA1.avi* and *NTA2.avi* sample frame, respectively, while an improvement of 49.43% and 22.20% was also obtained for the *Akiyo.avi* and *Forman.avi* benchmark frames.

After the compressed output was reconstructed,the respective results of Peak Signal to Noise Ratio (PSNR) for various techniques used areevaluatedas shown inTable 4.5

Table 4.5: Simulation Results of Peak Signal-to-Noise Ratio (PSNR) of Various Techniques before and after their Enhancement

Sample	DWT	E_DWT	%	DCT	E_DCT	%	LWT	E_LWT	%	FOA	E_FOA	%	mFOA	E_mFOA	%
NAERLS1.avi	19.23dB	20.42dB	5.83	18.98dB	20.43dB	7.10	21.89dB	23.36dB	6.30	77.27dB	79.85dB	3.23	83.39dB	83.69dB	0.36
NAERLS2.avi	15.75dB	16.93dB	6.97	16.76dB	17.35dB	3.40	16.92dB	17.84dB	5.16	81.95dB	84.14dB	2.60	84.75dB	85.11dB	0.42
NTA1.avi	15.09dB	15.78dB	4.37	15.41dB	16.21dB	4.94	16.01dB	16.57dB	3.38	76.26dB	79.14dB	3.64	80.19dB	80.82dB	0.80
NTA2.avi	16.17dB	17.29dB	6.48	16.61dB	17.25dB	3.71	16.94dB	17.63dB	3.91	77.23dB	82.24dB	6.10	81.94dB	83.21dB	1.53
Akiyo.avi	17.40dB	18.04dB	3.55	17.54dB	18.28dB	4.05	20.17dB	21.09dB	4.00	80.17dB	82.93dB	3.33	83.13dB	83.27dB	0.17
Forman.avi	17.55dB	18.92dB	3.44	17.97dB	19.10dB	5.92	18.64dB	19.98dB	6.71	79.86dB	83.59dB	4.46	83.07dB	83.54dB	0.56

Table 4.6: Percentage Improvement of mFOA over LWT and FOA for PSNR

Sample	**LWT (dB)**	**FOA (dB)**	**mFOA (dB)**	**Percentage (%) Improvement of mFOA over LWT**	**Percentage (%) Improvement of mFOA over FOA**
NAERLS1.avi	21.98	77.27	83.39	73.64	7.34
NAERLS2.avi	16.92	81.95	84.75	80.04	3.30
NTA1.avi	16.01	76.26	80.19	80.03	4.90
NTA2.avi	16.94	77.23	81.94	80.40	5.75
Akiyo.avi	20.17	80.17	83.13	75.74	3.56
Forman.avi	18.64	79.86	83.07	77.56	3.86

Table 4.7: Percentage Improvement of Enhanced mFOA over Enhanced LWT and FOA for PSNR

Sample	**E_LWT (dB)**	**E_FOA (dB)**	**E_mFOA (dB)**	**Percentage (%) Improvement of E_mFOA over E_LWT**	**Percentage (%) Improvement of E_mFOA over E_FOA**
NAERLS1.avi	23.36	79.85	83.69	72.09	4.59
NAERLS2.avi	17.84	84.14	85.11	79.04	1.14
NTA1.avi	16.56	79.14	80.82	79.51	2.08
NTA2.avi	17.63	82.24	83.21	78.81	1.17
Akiyo.avi	21.09	82.93	83.27	74.67	0.41
Forman.avi	19.98	83.59	83.54	76.08	-0.06

Table 4.5 showsresult of peak signal-to-noise ratio (PSNR) of various techniques and after their enhancement. From the table, all the five techniques (DCT, DWT, LWT FOA and mFOA) where significantly improved when enhanced. Example, with the NAELS 1avi sample video DWT was enhanced by 5.83%, DCT enhanced by 7.10%, LWT enhanced by 6.30%, FOA enhanced by 3.23% and the modified FOA (mFOA) enhanced by 0.36%.

4.4.1 Comparison of Techniques before Enhancement

Table 4.6 shows percentage improvement of mFOA over LWT and FOA for PSNR. From the table, it has been observed that the mFOA compression technique produced the highest PSNR values in all the six video samples compared to the FOA and LWT. For the respective individual sample video framesof NAERLS1.avi, NAERLS2.avi, NTA1.avi, and NTA2.avi, the mFOA produced PSNR percentage improvement of 73.64%, 80.04%, 80.03% and 80.40% over LWT.However, for the benchmark video frameof Akiyo.avi and Forman.avi, the mFOA also produced a percentage improvement of 75.74% and 77.56% over LWT.

Furthermore, for the respective individual sample video framesof NAERLS1.avi, NAERLS2.avi, NTA1.avi, and NTA2.avi, the mFOA produced PSNR percentage improvement of 7.34%, 3.30%, 4.90% and 5.75% over FOA.For the benchmark video frameof Akiyo.avi and Forman.avi, however, the mFOA also produced a percentage improvement of 3.46% and 3.86% over FOA.

4.4.2 Comparison of Techniques after Enhancement

Table 4.7 shows percentage improvement of enhanced mFOA over enhanced LWT and FOA for PSNR. From the table, it has been observed that the E-mFOA compression technique produced the highest PSNR values in all the six video samples compared to the E-FOA and E-LWT. For the respective individual sample video framesof NAERLS1.avi, NAERLS2.avi, NTA1.avi, and NTA2.avi, the E-mFOA produced PSNR percentage improvement of 72.09%, 71.04%, 79.51% and 78.81% over E-LWT.However, for the benchmark video frameof

Akiyo.avi and Forman.avi, the E-mFOA also produced a percentage improvement of 74.67% and 76.08% over E-LWT.

Also, for the respective individual sample video framesof NAERLS1.avi, NAERLS2.avi, NTA1.avi, and NTA2.avi, the E-mFOA produced PSNR percentage improvement of 4.59%, 1.14%, 2.08% and 1.17% over E-FOA and for the benchmark video frameof Akiyo.avi and Forman.avi, the E-mFOA also produced a percentage improvement of 0.41% and -0.06% over E-FOA. The last two results obtained for Akiyo and the Forman were not good enough, these might have been attributed to the intensity of light when the sample videos were taken.

4.5 Minimization Plots

Since the Firefly Optimization Algorithm (FOA) was implemented as a minimization tool of intra pixel distance as discussed in subsection 3.7, and equation (3.27).The following plots were generated to show the minimization procedure of the image pixel intensity with respect to standard FOA and the modified FOA (mFOA). The best achievable pixel intensity (encoded objective function) for each of the images was obtained at its global minimalwithout affecting quality of the image resolutions. Note that the three plots were computed from the MATHLAB simulation environment R2014a and implemented as in last part of appendix F2.

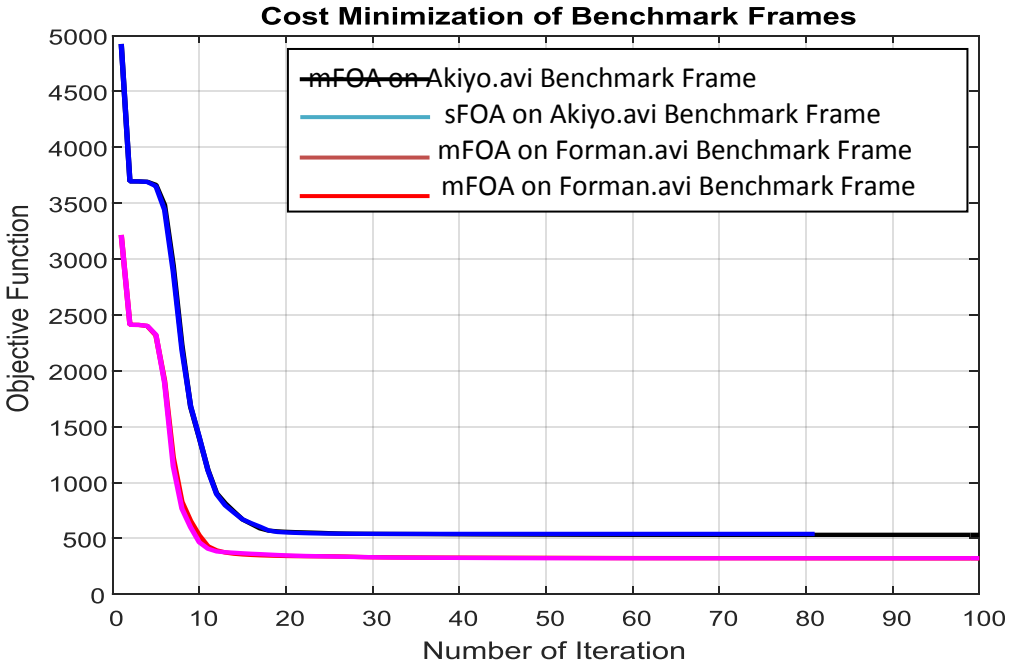


Figure 4.14: Minimization Plot of Benchmark Video Frame

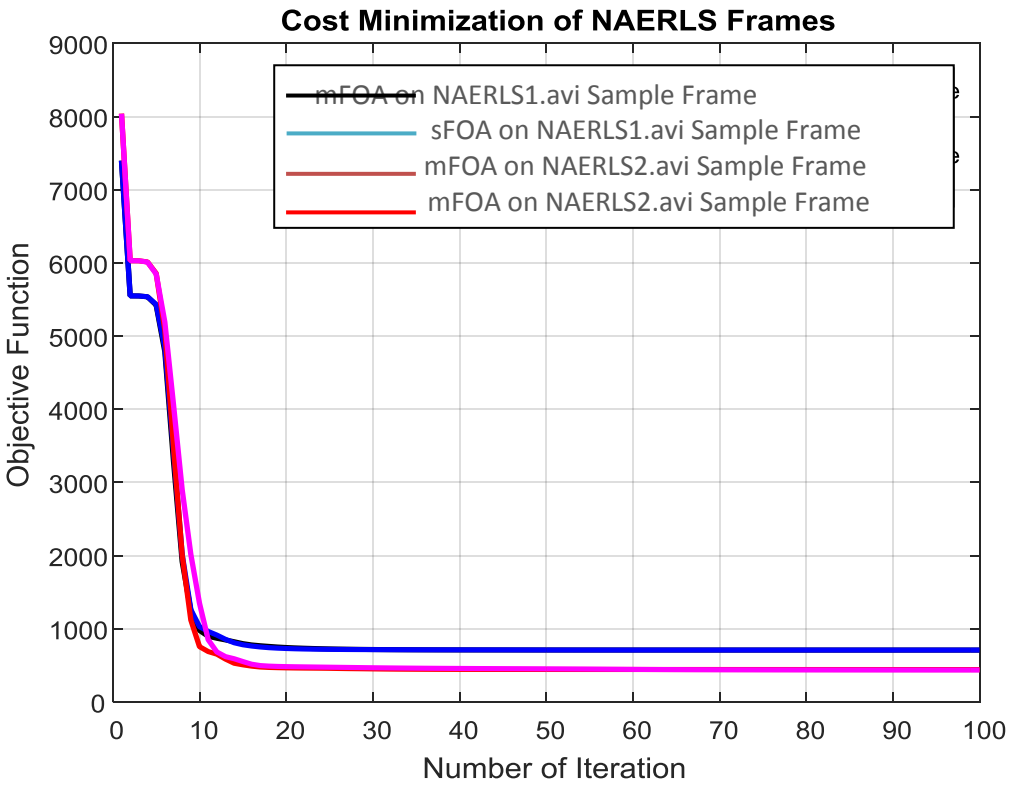


Figure 4.15: Minimization Plot of NAERLS Video Frames

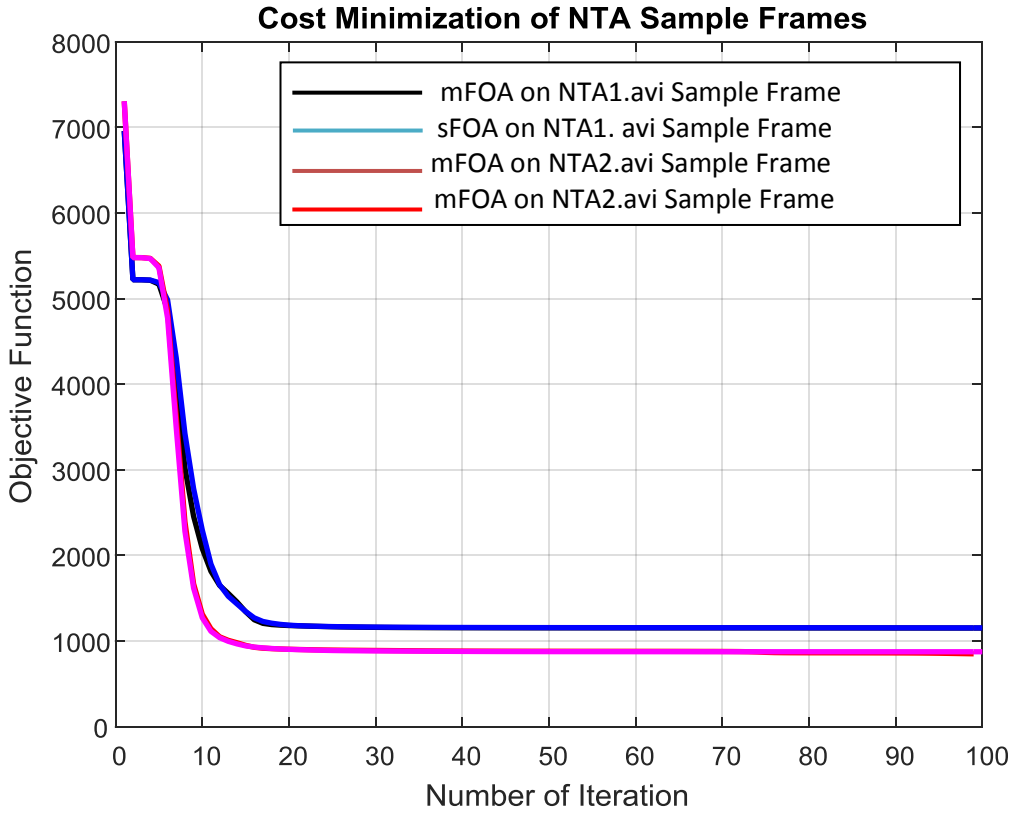


Figure 4.16: Minimization plot of Benchmark Video Frames

Figure 4.14,Figure 4.15, and Figure 4.16 show the standard and modified FOA compression plots for the benchmark video frames (Akiyo.avi and Forman.avi), NAERLS sample video frame (NAERLS1.avi and NAERLS2.avi), and the NTA video frames (NTA1.avi and NTA2.avi), respectively. Note that these plots are separated in order to ensure their easy interpretation.The objective function, that is, function of the luminous intensity of each image is plotted against the number of possible iterations for all the six images as shown in Figure 4.14 through 4.16, respectively. When the algorithm starts in each of the plots, that is, when the iteration is 1, the luminance intensity (objective function) of the image is minimized as shown in the respective plots.

NTA 1, for example, is minimized to about 5200 dB and that of NTA 2 to about 5400 dB as shown in Figure 4.16. The small delay observed as a trap on each curve is an indication that the minimization is taken a longer period at that point, which is a local minimalphenomenon.

As the iteration increases to 7 for NTA 1, the objective function is able to jump from the local minimal of 5200 dB to 1200 dB, where it is trapped again to another minimal which cannot escape but remains almost constant at this value. When the local minimal cannot be improved upon it is then called global minimal. At this point, the best possible solution for NTA 1 is achieved. Similarly, the same process is carried out on NTA 2 to attain its possible solution (global minimal) at 800 dB after 19 iterations and maintains almost a constant value as shown in Figure 4.16. Therefore, from the three plots the level of minimization is determined for the six video frames and the level of contras after any level of iteration.

CHAPTER FIVE

SUMMARY, CONCLUSION, AND RECOMMENDATION

5.1 Introduction

This chapter presents the final phase of this thesis. Brief overviews of the main research findings are summarised. The conclusions of the research are also presented, followed by a summary of the significant contributions of this research work. Suggestions for further research conclude this chapter.

5.2 Summary of Findings

This researchdeveloped an improved luminance enhancement based on pixels' intensity analysis and histogram distribution. The research also presented compression techniques based on developed Modified Firefly Optimization Algorithm (mFOA) for efficient video compression. Four standard compression techniques which were Discrete Cosine Transform (DCT),Discrete Wavelet Transform (DWT), Lifting Wavelet Transform (LWT) andFOA were improved using the brightness enhancement model. The developed enhancement model was used to improve the performance of the compressed video in order to minimize the loss of image quality during compression. A number of MATLAB 2015a simulations were performed and the following findings were recorded:

i. The size of video data was dependent on the quality of the video sample andthe total number of frames contained in the sample.
ii. The brightness of video data was dependent on the luminance content of the video sample and not on the hue and saturation content.
iii. An efficient luminance improvement achieved if emphasis was on the pixel intensity improvement and equal histogram distribution in the sampled video.

iv. The lifting wavelet transform produced the best compression than the discrete cosine transforms and discrete wavelet transforms under the same transform condition by producing a higher compression ratio and higher peak signal to noise ratio.

v. The standard FOA technique was much faster with a reduced compression size of the final video frames when compared to the DWT, DCT, and LWT compression techniques.

vi. The modified FOA(mFOA) compression technique performed efficiently with the highest improved peak signal to noise ratio when compared with all othertechniques. This improved behaviour is attributed to attractiveness and luminance intensity of fireflies toward individual members and the modifications made to overcome the effectof higher distance between some pixels of the image,thereby reducing luminance intensity. This was achieved using the inverse square law relations equations (3.29) through (3.21).

vii. The developed luminance enhancement model performed efficiently on coloured video frame than on black and white video frame with a better peak signal to noise ratio and improved histogram.

viii. The developed luminance enhancement techniques do not affect the computational complexity of the existing video compression algorithms.

5.3 Conclusion

This thesis has presented the development of a brightness enhancement technique for video frame pixel improvement based on pixel intensity analysis. The research also achieved an FOA based compression technique using the light intensity and attractionbehaviour of fireflies in nature. This optimization technique was further modified by nullifying the effect of long distance between any two pixels in an image using the inverse square law relation.

The enhanced standard transform techniques, including the modified FOA were used on a total of six (four acquired and two benchmark) sample video data to achieve better results in terms of image quality. The performance of the developed enhancement model was initially evaluated on the sampled video frames. Thereafter, it was applied to enhance the output of the developed compression techniques. Simulation results showed that, the developed enhancement method was efficient with an improved pixel intensity and histogram distribution in all the video frames as shown in Table 4.5. The peak signal to noise ratio evaluation depicted in Table 4.5 showed an efficient signal quality of the enhanced techniques compared to the standards. From thegreyscale frames (black and white) of Table 4.1, PSNR values of 31.95dB, 22.30dB, 17.71dB, 23.17dB, 15.06 dB and 19.17dB were obtained for the NAERLS1.avi, NAERLS2.avi, NTA1.avi, NTA2.avi, Akiyo.avi and Forman.avi video frames, respectively. Also, for the coloured video frames of Table 4.2, PSNR values of 36.45dB, 26.65dB, 24.45dB, 38.90dB, 21.19dB and 28.06dB were obtained for the NAERLS1.avi, NAERLS2.avi, NTA1.avi, NTA2.avi, Akiyo.avi and Forman.avi video frames, respectively. From equation (4.1), the PSNR percentage improvement of 12.45%, 16.32%, 27.57% and 19.83% were achieved over the grey level colour (black and white) for the NAELS1.avi, NAELS2.avi, NTA1.avi and NTA2.avi respectively. Also, a percentage improvement of 28.93% and 31.68% were obtained for the coloured image over the grey level image for Akiyo.avi and Forman.avi benchmark video frame, respectively.

The performance of the modified FOA compression technique was compared with both the LWT and FOA before and after enhancement. Simulation results show that the mFOA method is highly efficient with a significant improvement in peak signal to noise ratio. It was observed that the mFOA based technique produced a PSNR percentage improvement of 73.64%, 80.04%, 80.03% and 80.40% over LWT on the NAERLS1.avi, NAERLS2.avi, NTA1.avi and NTA2.avi sample videos before enhancement. Also, the mFOA based

technique produced a percentage improvement of 7.34%, 3.30%, 4.90% and 5.75% over the FOA on the same sample framebefore enhancement. Similarly, for Akiyo.avi and Forman.avi benchmark video frames, the mFOA produced a percentage improvement of 74.78% and 77.56% over LWT before enhancement respectively. Also, the mFOA obtained a PSNR percentage improvement of3.46% and 3.86% over FOA on the Akiyo.avi and Forman.avi benchmark video framesbefore enhancement.

Similarly, it was also observed that, the enhanced modified FOA (E-mFOA) based technique produced a PSNR percentage improvement of 72.09%, 79.04%, 79.51% and 78.81% over LWT on the NAERLS1.avi, NAERLS2.avi, NTA1.avi and NTA2.avi sample videos after enhancement. Also, the E-mFOA based technique produced a percentage improvement of 4.59%, 1.14%, 2.08% and 1.17% over the E-FOA on the same sample frame after enhancement. Similarly, for Akiyo.avi and Forman.avi benchmark video frames, the E-mFOA produced a percentage improvement of 74.67% and 76.08% overE- LWT after enhancement. Also, the E-mFOA obtained a PSNR percentage improvement of 0.41% and -0.06% on the Akiyo.avi and Forman.avi benchmark video frames after enhancement.

5.4 Significant Contributions

A significant number of research works have been conducted in the area of video processing. Many research efforts have also been made on video compression for easy video transmission. Due to the advancement of digital technology, research attendance has been drawn back to the reduction of data size for efficient storage and transmission. Based on these concepts, the significant contributions of this research are highlighted as follows:

i. The researchachieved a luminance enhancement when extended to coloured video data by considering the video main object pixel intensity and background object isolation. Using equation (4.1) on the values of the grey level PSNR in Tables 4.1 and

those of the coloured in Table 4.2, it was discovered that the PSNR percentage improvement of 12.45%, 16.32%, 27.57% and 19.83% for coloured image were achieved over the grey level (black and white) for the NAELS1.avi, NAELS2.avi, NTA1.avi and NTA2.avi, respectively. Also, a percentage improvement of 28.93% and 31.68% wereobtained for the coloured images over the grey level images for Akiyo.avi and Forman.avi benchmark video frame, respectively.

ii. The research developed a modified Firefly Optimization Algorithm(mFOA) which was used for efficient compression. The output of this mFOA based compression was enhanced using the developed luminance enhancement model in item 2. It was found that the mFOA was better than the LWT by a PSNR improvement of 73.64%, 80.04%, 80.03% and 80.40%for NAERLS1.avi, NAERLS2.avi, NTA1.avi and NTA2.avi captured video frames, respectively before enhancement and an improvement of 75.78% and 77.56% for Akiyo.avi and Forman.avi benchmark video frames, respectively.

iii. Furthermore,the mFOA was also discovered to have outperformed the FOA with a PSNR improvement of 7.34%, 3.30%, 4.90% and 5.75%for NAERLS1.avi, NAERLS2.avi, jNTA1.avi and NTA2.avi captured video frames, respectively before enhancement and an improvement of 3.56% and 3.86% on Akiyo.avi and Forman.avi benchmark video frames, respectively.

iv. Similarly, the enhancedmFOA (E-mFOA) compression technique also produced a PSNR improvement of 72.09%, 79.04%, 79.51% and 78.81% over enhanced LWT (E-LWT)for NAERLS1.avi, NAERLS2.avi, NTA1.avi and NTA2.avi capture video frames, respectively after enhancement and an improvement of 74.67% and 76.08% for Akiyo.avi and Forman.avi benchmark video frames, respectively.

v. Furthermore, theenhanced mFOA (E-mFOA) compression technique also produced a better PSNR improvement of 4.59%, 1.14%, 2.08% and 1.17% over enhanced FOA (E-FOA) for NAERLS1.avi, NAERLS2.avi, NTA1.avi and NTA2.avi captured video frames, respectively after enhancement. Except for the Akiyo.avi and Forman.avi benchmark video frames, where an insignificant improvement of 0.41% and -0.06% were registered due to the amount of light present when the video clip was taken.

5.5 Recommendations for Further Research

Video processing is one of the most important research areas in digital signal processing research. Video compression has been extensively investigated and due to the high amount of size usually associated with digital video, these sizes which are usually a challenge for digital video storage and transportation still posed a big problem in video processing. It is important to understand the fundamental governing principles behind video processing. The more researchers focus on finding an alternative approach to reduce video size, the more researches are faced with other problems: The following areas for further research are recommended:

i. The performance of the luminance enhancement technique can be investigated for video sample with large size and multiple objects around the scene.
ii. The enhancement techniques can be applied to image processing for efficient edge detection, lane detection, etc.
iii. The enhancement can be applied to other form of signal transformation techniques such as Multistage Vector Quantization, Embedded Zero Tree Wavelet, etc.
iv. The developed enhancement algorithm can also be applied in colour quantization enhancement.
v. The performance of the achieved mFOA based compression can be verified on a video sample of large amount of size (say 1GB).

REFERENCES

Abdullah1, M.S. and Rao, N. S. (2013) “Image Compression using Classical and Lifting based Wavelets” Vol. 2, Issue 8, ISSN (Online): 2278-1021. International Journal of Advanced Research in Computer and Communication Engineering’ Retrieved on 25th November, 2015. www.ijarcce.com.

Ahmed, N., Natarajan, T. , Rao, K.R. (2006) "Discrete Cosine Transform" p. 90-93, ISSN : 0018-9340. Departments of Electrical Engineering and Computer Science, Kansas State University Retrieved on 25th November, 2015. http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=1672377&abstractAccess=no&user Type=inst.

Albert Wong and Shyh-Liang Lou (2000) “Iasa Hand Book of Medical Imaging Processing and Analysis” Applied Physics Laboratory Johns Hopkins University Laurel Maryland. Retrieved on 1st January, 2014. http://mariorad.com/books/General%20radiology/053%20Handbook%20of%20Medical%20Imaging%20Processing%20and%20Analysis%20-%20Isaac%20Bankman.pdf.

Altunbasak, Y. and Kamaci, N. (2004) “An analysis of the DCT coefficient distribution with the H.264 video coder” ISSN: 1520-6149, Volume: 3, pp. 177-80. Retrieved on August, 2014. IEEE International Conference. http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=9248

Amara Graps (1995) "An Introduction to Wavelets" Institute of Electrical and Electronics Engineers, Inc. Personal use of this material is permitted. Retrieved on 20th January, 2015.

Amir Said (2004) “Introduction to Arithmetic Coding-Theory and Practice” Imaging Systems Laboratory HP Laboratories Palo Alto HPL. Retrieved on 1st April, 2012. http://www.hpl.hp.com/techreports/2004/HPL-2004-76.pdf.

Andrew B. Watson, (1994) "Image Compression Using the Discrete Cosine Transform" NASA Ames Research Center, Mathematica Journal, 4(1), p. 81-88. Retrieved on 2nd September, 2014. http://vision.arc.nasa.gov/publications/mathjournal94.pdf.

Anila S. and Devarajan N. (2012) “Preprocessing Technique for Face Recognition Applications under Varying Illumination Conditions” Volume 12 Issue 11 Version 1.0. Type: Double Blind Peer Reviewed International Research Journal. Global Journal of Computer Science and Technology Graphics & Vision. Retrieved on 14nd May, 2014.

Augustin J. and Senthil K. (2012) “OPTIMAL CUBE FOR 3D-DCT BASED VIDEO COMPRESSION FOR DIFFERENT MOTION LEVELS” pp. 526-529. Retrieved on 15TH May, 2014. http://ictactjournals.in/paper/IJIVP_VOL3_ISS2_Paper6_526_529.pdf.

Babu, D. V. and Alamelu, N. R. (2015) “PERFORMANCE ANALYSIS OF MEDICAL IMAGES APPLYING NOVEL MORPHO CODEC” VOL. 10, NO. 9, ISSN 1819-6608 ARPN Journal of Engineering and Applied Sciences. Retrieved on 25th November, 2015. www.arpnjournals.com.

Ballabeni A., Apollonio F. I., Gaiani M., Remondino F. (2015) ADVANCES IN IMAGE PRE-PROCESSING TO IMPROVE AUTOMATED 3D RECONSTRUCTION.3D Virtual Reconstruction and Visualization of Complex Architectures pp 25-27, Volume XL-5/W4. The International Archives of the Photogrammetry, Remote Sensing and Spatial Information Sciences. Retrieved on 20th April, 2014.

Bhaskaranand, Malavika, and Jerry D. Gibson. (2009) "Distributions of 3D DCT coefficients for video." Acoustics, Speech and Signal Processing, pp. 793- 796. ICASSP. IEEE International Conference on. IEEE. Retrieved 20th January, 2015. https://scholar.google.com/scholar?cluster=13756603452538859329&hl=en&as_sdt=0,5&sciodt=0,5.

Bisjerdi, M. H., Behrad, A. (2012) "Video Compression USING a New Active Mesh Based Motion Compensation Algorithm in Wavelet Sub-Bands" Vol. 3, pp.368-376. Journal of Signal and Information Processing. Retrieved on 25th November, 2015. http://www.SciRP.org/journal/jsi

Blinn, J.F. (1992) "The World of Digital Video" Proceedings of IEEE Compute Applications, P. 106-112. Retrieved on 25th November, 2015. https://cognitech.com/pdfs/samplingTheory.pdf.

Boon, C. S., Guleryuz, O. G., (2006) "Sparse super-resolution reconstructions of video from mobile Devices in Digital TV Broadcast applications" vol. 5, pp 79-881. Retrieved on 25th November, 2015. http://eeweb.poly.edu/~onur/publish/san_diego_sr.pdf

Branko (2009) "Neuron Network Applied to Video Encoder"ISBN 978-953-307-027-8, pp. 572. : Micro Electronic and Mechanical Systems, downloaded from SCIYO.COM.Retrieved on August, 2014. http://cdn.intechopen.com/pdfs-wm/6645.pdf.

Brochier M. G., Vacavant A., Cerutti G., Kurtz C., Weber J., Tougne L. (2015) "Tree leaves extraction in natural images: Comparative study of pre-processing tools and segmentation methods" TRANSACTIONS ON IMAGE PROCESSING. Retrieved on 20th April, 2014. http://dx.doi.org/10.1109/TIP.2015.2400214

Catania, (2008) "Principles of Image Compression" Arcangelo BrunaAdvanced System Technology.Retrieved on 13th May, 2013.

Chan Y. H. "Fundamentals of Digital Image Processing" Applications of image processing. 5th Edition. Retrieved on 27th July, 2016.

Chandan S. R. and Sukadev M. (2013) "A Hybrid Image Compression Scheme using DCT and Fractal Image Compression "The International Arab Journal of Information Technology, Vol. 10, No. 6, pp. 553 562.
Retrieved on 24th April, 2014. http://ccis2k.org/iajit/PDF/vol.10,no.6/4378.pdf.

Chang-Hoon Yeo (2007) "Cross-Colour Noise Reduction Algorithms for NTSC Signals" Conference Location: Las Vegas, NV, pp. 1 -2. Publisher: IEEE. Retrieved on 20th August, 2015. http://140.98.202.196/xpl/mostRecentIssue.jsp?punumber=4145986.

Chen, K., Zhou, Y., Zhang, Z., Dai, M., Chao, Y., & Shi, J. (2016). Multilevel image segmentation based on an improved firefly algorithm. Mathematical Problems in Engineering.Retrieved on 24th April, 2014.

Chengging Li., Kongfeng Z., Vijayan K. A., Dietmar S., (2014) "No-Reference Video Quality Assessment Based on Artifact Measurement and Statistical Analysis" (Volume: 25, Issue: 4), ISSN: 1051-8215, pp. 533 – 546, Circuits and Systems for Video Technology, IEEE Transactions.Retrieved on 24th April, 2014.

Chesta Jain, Vijay Chaudhary,KapilJain,SaurabhKarsoliya, (2011) "Performance Analysis of Integer Wavelet Transform for Image Compression". vol. 5, pp 79-88. Retrieved on 6th November, 2015.

Chris Solomon & Toby Breckon, (2010) "Fundamentals of Digital Image Processing A Practical Approach with Examples in Matlab" School of Physical Sciences, Universit y of Kent, Canterbury, UK.vol. 5, pp 90-101. Retrieved on 25th December, 2014.

Cuizhu, S., (2006) "Automatic Image Quality Improvement for Video conferencing" pg 701-704. Retrieved on 25th May, 2015. https://www.google.com.ar/patents/US8175382.

Deepa S., Yashwant K., Vijayshri . (2014) "Image Compression using Discrete Cosine Transform and Adaptive Huffman Coding" International Journal of Emerging Trends & Technology in Computer Science (IJETTCS), Volume 3, Issue 1, Web Site: www.ijettcs.org. pp. 90-94. Retrieved on 25th May, 2014. http://www.ijettcs.org/Volume3Issue1/IJETTCS-2014-02-01-049.pdf.

Deepak, G., and Lee, I., (2011) "Nonlinear transfer function-based local approach for colour image enhancement." Consumer Electronics, IEEE Transactions on 57. No. 2(2011): 858-865. Retrieved on 25th December, 2015. http://ieeexplore.ieee.org/Xplore/d efdeny.jsp?url=http%3A%2F%2Fieeexplore.ieee.org.

Dhanasekar, B. And Ramamoorthy, B. (2010) "Restoration of blurred images for surface roughness evaluation using machine vision" Volume 43, Issues 1–2 Pages 268–276. Retrieved on 30th October, 2015. http://edlib.net/2015/icidret/icidret2015029.pdf.

Djordje Mitrovic (2006) "Video Compression" – University of Edinburgh. Retrieved on 2nd July, 2014. http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/AV0506/s0 561282.pdf

Er. Ramandeep Kaur Grewal and Navneet Randhawa (2012) "IMAGE COMPRESSION USING DISCRETE COSINE TRANSFORM & DISCRETE WAVELET TRANSFO RM" International Journal of Computing & Business Research ISSN (Online): 2229-6166 Proceedings of'I Society 2012' at GKU, Talwandi Sabo Bathinda (Punjab). Retri eved on 17th May, 2014.http://www.researchmanuscripts.com/isociety2012/32.pdf.

Esakkirajan, S.and Vennila, I, (2011) "Salt and pepper noise removal in video using adaptive decision based median filter". pp 87-90. IEEE International Conference on Multimedia, Signal Processing and Communication Technologies, pp 87-90. Retrieved on 25th December, 2014. http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnu mber=6150444.

Faghih, M.M. and Moghaddam, M.E (2011) "Neural grey edge: Improving grey edge algorith m using neural network" ISSN: 1522-4880, pp. 1705 – 1708, Publisher: IEEE. Retrieved on 2nd September, 2016. http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?pu number=6094293.

Fan, Y. and Jin Wu., (2010) "An improved image contrast enhancement in multiple-peak images based on histogram equalization.t" In Computer Design and Applications (ICCDA), International Conference on. vol. 1, pp. 346. IEEE.Retrieved on 5th November, 2016.

Fernando Pereira, (2014/2015) “BASICS ON DIGITAL AUDIO AND VIDEO REPRESENT ATIO”Audio and Video Communication. Retrieved on 25th April, 2014. http://www.img.lx.it.pt/~fp/cav/ano2014_2015/Slides%202014-2015/CAV_2_Basics_Digital_AV_Representation_2014-2015%20Web.pdf.

Fillali F., (2012) “Image Restoration using a Network of Reduced and Regularized Neural Networks” International Journal of Computer Applications (0975 – 8887), Volume 54, No.8. Retrieved on 3rd July, 2014. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.258.8018&rep=rep1&type=pdf.

Fisher, W. (2008). “Digital Video and Audio Broadcasting Technology. A Practical Engineering Guide”. 2nd edition. Springer, ISBN978-3-540-76357 4, Berlin. Retrieved on 10th December, 2015 http://www.springer.com/us/book/9783540763581.

Fister, I., Fister Jr, I., Yang, X.-S., & Brest, J. (2013). A comprehensive review of firefly algorithms. Swarm and Evolutionary Computation, 13, 34-46. Retrieved on 29th December, 2016.

Frank Y. Shih (2010) “Image Processing and Pattern Recognition Fundamental and Techniques” Wiley IEEE Press. 445 Hoes Lane Piscataway, NJ 08854.Retrieved on 20th June, 2014.

Frank Y. Shih (2010) “Image Processing and Pattern Recognition Fundamental and Techniques” Wiley IEEE Press. 445 Hoes Lane Piscataway, NJ 08854. Retrieved on 3rd December, 2014.

Garg R. and Kumar A. (2012) “COMPARISION OF VARIOUS NOISE REMOVALS USING BAYESIAN FRAMEWORK” Vol.2, pp-265-270 ISSN: 2249-6645. International Journal of Modern Engineering Research (IJMER).Retrieved on 6th February, 2014.www.ijmer.com

Gupta M., Garg K., Kaushik A. (2011) “Review: Image Compression Algorithm” ISSN: 2231-0711, Vol 1, Issue 10, P(649-654), IJCSET. www.ijcset.net.
Retrieved on April, 2015.

Ghodke, V. N. and Ganorkar, S. R. (2013) “Image Enhancement Using Spatial Domain Techniques and Fuzzy Intensification Factor” Volume 3, pp 566-579. International Journal of Emerging Technology and Advanced Engineering. Retrieved on 4th January, 2016. http://www.ijetae.com/files/Volume3Issue10/IJETAE_1013_68.pdf.

Gwanggil Jeon, (2014) “Colour Image Enhancement by Histogram Equalization in Heterogeneous Colour Space” p15 27. Retrieved from http://www.sersc.org/journals/IJMUE/vol9_no7_2014/26.pdf November, 2014.

Hamsavahini, R., Naveena, S., Meti Reddy Kshetra, Chaitra, R., Chaithra, K. (2014) “A Survey on FPGA implementation of 3D DWT using Lifting based Algorithm” ISSN (Print): 2279-0047 ISSN (Online): 2279-0055, P.156. International Journal of

Emerging Technologies in Computational and Applied Sciences (IJETCAS). Retrieved on 25th November, 2015.www.iasir.net.

Hassan, F. and Pouriya, E. (2013) "Quality Enhancement of Synthesized Video by Improvement of VLC Using Artificial Neural Network" Research Journal of Applied Sciences, Engineering and Technology 6(17): 3098-3109, 2013 ISSN: 2040-7459; e-ISSN: Pg. 3098-3109. Retrieved on 15th April, 2015 http://maxwellsci.com/print/rjaset/v6 3098-3109.pdf.

Hazarathaiah, A., Rao, B. P. (2014) "Medical Image Compression using Lifting based New Wavelet Transforms" Vol. 4, No. 5, pp. 741~750. ISSN: 2088-8708. International Journal of Electrical and Computer Engineering (IJECE). Retrieved on 25th November, 2015.

Horng, M. H. (2012). Vector quantization using the firefly algorithm for image compression. Expert Systems with Applications, 39(1), 1078-1091.Retrieved on 19th May, 2016.

Hyung-Seung., (2009) "video Adaptive enhancement using neural network" Volume: 55, Issue: 3. pp. 1637 -1644, ISSN: 0098 306. Retrieved on 17th May, 2014. http://www.ewh.ieee.org/soc/ces/index.html.

Ian T. Young, Jan J. Gerbrands, Lucas J. vanVliet (1995) "Fundamentals of Image Processing" Delft University of Technology. Retrieved on 10th January, 2013.http://www.google.com.ng/url?sa=t&rct=j&q=&esrc=s&source=web&cd=1&ved=0CBwQFjAA&url=http%3A%2F%2Frepository.tudelft.nl%2Fassets%2Fuuid%3A1d58e4e5 4a0365a0506808fcf2de82%2FImageProcessingFundamentals.pdf&ei=TKyTVcO6NLDQ7AbP_ruYAw&usg=AFQjCNGnGJjf29x3QmLsv_TJ3NwW3HLktw&sig2=sziLf3FTVdmeAdLf6ZIHog.

Image Processing Toolbox™ User's Guide R2014b, P. 245 267. Retrieved on 20th December, 2016. http://www.mathworks.com/help/pdf_doc/images/imagestb.pdf . Image sources: https://media.xiph.org/video/derf/

Isaac N. Bankman (2000) "HANDBOOK OF MEDICAL IMAGING PROCESSING AND ANALYSIS" Applied Physics Laboratory Johns Hopkins University Laurel, Maryland. ISBN: 0-12-077790-8. Retrieved on 10th August, 2015.

Jayaraj, V and Ebenezer, D, (2010) "A New and Efficient Algorithm for the Removal of High Density Salt and Pepper Noise in Images and Videos", PP 409-413, IEEE Second International Conference on Computer Modeling and Simulation. Retrieved on 17th November, 2014.http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=5421557.

Jerome G., Mathieu W., Francoise P., Mickael R., (2010) "LLVM – Based and Scalable MPEG-RVC Decoder", Journal of Real Time Image Processing, (fulltext32). Retrieved on 17th February, 2014. http://airccse.org/journal/ijsc/papers/2411ijsc07.pdf.

Jerry Gibson, (2000) "HANDBOOK OF IMAGE AND VIDEO PROCESSING" An Academic Press Series in Communications, Networking, and Multimedia 32 Jamestown Road, London, NW1 7BY, UK. Library of Congress Catalog. Number: 99-691 ISBN 0-121197905. Retrieved on 17th May, 2014. http:llwww.hbuk.co.uk/apl.

John G. Apostolopoulos, Wai- tian Tan, Susie J. Wee (2002) "Video Streaming: Concepts, Algorithms, and Systems" Mobile and Media Systems Laboratory, HP Laboratories Palo Alto. Retrieved on 1st July, 2014. http://www.hpl.hp.com/techreports/2002/HPL-2002-260.pdf.

Kamboj P. and Rani V. (2013) "A BRIEF STUDY OF VARIOUS NOISE MODEL AND FI LTERING TECHNIQUES" Volume 4, No. 4, Journal of Global Research in Compute r Science. Retrieved on 17th May, 2014. REVIEW ARTICLE Available Online at www.jgrcs.info.

Katiyar, S., (2012) "TV and Satellite Communication" 1st Edition: 2007-2008, p5-53. katriab ooks@yahoo.com. Retrieved on 17th May, 2014. https://www.nikubook.com/tv-and-satellite-communication-by-sapna-katiyar.html

Kaur S., Kumar P. (2017) **"**Study on Various Techniques of Image Enhancement"International Journal of Computer Applications p (0975 – 8887). Volume 158. Retrieved May, 2017.

Kejgir, S. G., Kokare, M. (2012) "Lifting Wavelet Transform with Singular Value Decomposition for Robust Digital Image Watermarking" Vol. 39, No.18.pp34-39 International Journal of Computer Applications (0975 – 8887).Retrieved on 17th July, 2014

Keller, S. H., F. Lauze,, (2013) "Video Super-Resolution using Simultaneous Motion and Intensity Calculations" Vol. 6, pp. 35-50. Retrieved on 30th October, 2015http://www.jpi er.org/PIER/pier136/30.12110809.pdf

Keith, Jack "Video demystified fourth edition. A handbook for the digital engineer" ISBN: 978-0-7506-7822-3 copyright © 2005 Elsevier Inc.Retrieved on 17th May, 2014.

Kim, K., N. Neretti,, (2008) "MAP Fusion Method for Super resolution of Images with Locally Varying Pixel Quality" VOL. 5, pp. 69-78. Retrieved on 17th June, 2014. https://www.cs.tau.ac.il/~nin/papers/KimMAPFusionApr08.pdf in July, 2014.

Komal R. H. (2013) "Application of Genetic Algorithm for Image Enhancement and Segmentation" International Journal of Advanced Research in Computer Engineering & Technology (IJARCET) Volume 2, Issue 4, pg 2278 - 1323. Retrieved on 15th April, 2015. http://worldwidescience.org/topicpages/i/identification+genetic+algorithm.html

Kwok, N. M., Ha, Q. P. (2010) "Colour image contrast enhancement using a local equalizatio n and weighted sum approach." In Automation Science and Engineering (CASE). Con ference on, pp. 568- 573. IEEE. Retrieved on 16th May, 2015. http://ieeexplore.ieee.org/ xpl/articleDetails.jsp?arnumber=55847

Le Dinh, K. and Chon-Tam Le Dinh (2007) "Cascaded Neural Network-based S-VHS Restoration" Volume: 53 Issue: 2, ISSN: 0098-3063, pp. 513 – 518. Publisher: IEEE.Retrieved on 10th May, 2014. http://140.98.202.196/xpl/RecentIssue.jsp?punumber=30.

Li and Drew (2003) “Fundamentals of Multimedia, Chapter 9 Image Compression Standards” pp33-39. Retrieved on 15th September, 2014 http://www.cs.rutgers.edu/~elgammal/classes/cs334/slide9_short.pdf.

Li C., Huang R., Ding Z., Gatenby J.C., Metaxas D. N., Gore J. C. (2011) “A Level Set Method for Image Segmentation in the Presence of Intensity In homogeneities With Application to MRI” VOL. 20, NO. 7. pp23-30. IEEE TRANSACTIONS ON IMAGE PROCESSING.Retrieved on 2nd December, 2014.

Li C., Xu C., Gui C., Martin D. (2010) “Distance Regularized Level Set Evolution and Its Application to Image Segmentation” VOL. 19, NO. 12. pp33-45. IEEE TRANSACTIONS ON IMAGE PROCESSING.Retrieved on 13th November, 2015.

Lin Z. and H. Y. Shum (2004) “Fundamental Limits of Reconstruction-Based Super resolution Algorithms under Local Translation” Vol. 9 No. 22. pp88-91 IEEE TRANSACTIONS ON PATTERN. ANALYSIS AND MACHINE INTELLIGEN VOL. 26, NO. 1. PP. 526. Retrieved on 11th August, 2014. http://asp.eurasipjournals.com/content/2006/1/073767

Lina (2009) "A short introduction to wavelets and their applications" Vol. 09 Issue 2, pp57-68. Institute of Electrical and Electronics Engineers. Retrieved on 12th May, 2016. http://dl.acm.org/citation.cfm?id=1669919

Liu K. C. (2012) “Prediction error pre-processing for perceptual colour image compression” Liu EURASIP Journal on Image and Video Processing. VOL. 10 No. 20. pp 55-68. Retrieved on 17th May, 2015. http://jivp.eurasipjournals.com/content/2012/1/3

Liu, H. and Heynderickx, I. (2008) “A no-reference perceptual blockiness metric” ISSN: 1520-6149, pp. 865 – 868. Conference Location: Las Vegas, NVA coustics, Speech and Signal Processing, ICASSP. IEEE International Conference. Retrieved on 14th May, 2015.

Lu S.P., Zhang S. H., (2011) “Saliency-Based Fidelity Adaptation Preprocessing for Video Coding” JOURNAL OF COMPUTER SCIENCE AND TECHNOLOGY 26(1): DOI 10.1007/s11390-011-1122-y.Retrieved on 17th May, 2014.

Maneesha Gupta, Dr.Amit Kumar Garg (2012) “Analysis Of Image Compression Algorithm Using DCT” International Journal of Engineering Research and Applications (IJERA) ISSN: 2248-9622, Vol. 2, Issue 1, pp.515-521. Retrieved on 10th June, 2014.www.ijera.com.

Mark Nelson and Jean-loup Gailly, (2002) “Data Compression Techniques” Book, 2nd Ed College of Applied Studies University of Bahrain. Retrieved on 13th March, 2016 .http://staff.uob.edu.bh/files/600435156_files/The_Data_Compression_215.pdf

Mathew, K. and Shibu, S. (2011) “Wavelet based Technique for Super Resolution Image Reconstruction” International Journal of Computer Applications (0975 – 8887), Vol. 33, No.7, pp.33-37. Retrieved on 6th March, 2015. http://iasir.net/IJEBEApapers/IJEBEA13-146.pdf.

MATLAB User's Guide R2014b. Retrieved on March, 2014. http://cn.mathworks.com/help/pdf_doc/coder/coder_ug.pdf

Mauro, M., Livio L., Tea A., Alfonso D., Leonardi R.,(2011) "Optimal rate adaptation with Integer Linear Programming in the scalable extension of H.264/AVC", ISSN: 1522-4880, pp. 1625 – 1628, Image Processing (ICIP), 18th IEEE International Conference on.Retrieved on 22nd June, 2016.

McAndrew, A. (2004) "An Introduction to Digital Image Processing with Matlab" School of Computer Science and Mathematics, Victoria University of Technology:Retrieved on 10th July, 2015. http://share.its.ac.id/pluginfile.php/371/mod_resource/content/1/An_Introduction_To_Digital_Image_Processing_With_Matlab.pdf.

Mohamed A. A., Fawzy E., Mahmoud S., Gouda I. S. (2015) "A comparative study of noise removal from High Resolution Remote Sensing Images" Vol.03 Issue-06, ISSN: 2321-1776. Page 78 International Journal in IT and Engineering, Impact Factor- 4.747 http://www.ijmr.net.in email id- irjmss@gmail.com

Muzhir Shaban Al-Ani1 and Talal Ali Hammouri2 (2011) "Video Compression Algorithm Based on Frame Difference Approaches" Vol.2, No.4, pp. 72 International Journal on Soft Computing (IJSC). Retrieved on 10th June, 2014. http://airccse.org/journal/ijsc/papers/2411ijsc07.pdf.

Nabeel H. K., (2006) "Video Clip Image Compression Using DCT Technique" Department of Computer Science, University of Babylon, Iraq. INTERNATIONAL CONFERENCE OF COMPUTER ENGINEERING.Retrieved on 10th June, 2014.

Naccari, M., (2009) "No-Reference Video Quality Monitoring for H.264/AVC Coded Video" ISSN: 1520-9210, Volume: 11 Issue: 5, pp. 932 – 946. Multimedia, IEEE Transactions. Retrieved on 18th June, 2015. http://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=6046.

Naccari, M., (2008) "No-reference modelling of the channel induced distortion at the decoder for H.264/AVC video coding" ISSN: 1522-4880, pp. 2324 – 2327. IEEE International Conference. Retrieved on 12th June, 2015. http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=4667700

Naemura, T. and Tanaka, M. (2011) "Rate-distortion analysis of super-resolution image/video decoding" ISSN: 1522-4880, PP. 1629 – 1632. Image Processing (ICIP), 18th IEEE International Conference. Retrieved on 10th January, 2015.

Nageswara R. T., and Srinivasa K. D. (2008)"Image Compression Using Discrete Cosine Transform" Georgian Electronic Scientific Journal: Computer Science and Telecommunications No.3 (17), pp345-356. Retrieved on 29th March, 2014. http://ww

w.researchgate.net/publication/229009384_Image_Compression_Using_Discrete_Cosine_Transform.

Narwaria, M and Weisi Lin (2010) "Objective Image Quality Assessment Based on Support Vector Regression" ISSN: 1045-9227, Volume: 21, Issue: 3, pp. 515 – 519. Neural Networks, IEEE. Transactions. Retrieved on 20th April, 2015. http://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=72.

Naveen Kumar N. & Ramakrishna S. (2012) "An Impressive Method to Get Better Peak Signal Noise Ratio(PSNR), Mean Square Error (MSE) Values Using Stationary Wavelet Transform (SWT)" Vol. 12, Issue 12, Version 1.0. pp 342-357. Global Journal of Computer Science and Technology. Retrieved on 14th December, 2016.

Ndajah P., Kikuchi H., & Yukawa M. (2011). "An investigation on the quality of Denoised Image" International Journal of Circuits, Systems and Signal Processing. 5(4), pp. 423-434.Retrieved on 11th July, 2016.

Patidar P., Srivastava S.,Gupta M., Nagawat A. (2010) "Image De-noising by Various Filters for Different Noise" International Journal of Computer Applications. Vol. 9-45.
p(0975 –8887). Retrieved on 14thApril, 2014.

Padmavathi S. Priyalakshmi I. B. Soman K. P. (2012) "HIERARCHICAL DIGITAL IMAGE INPAINTING USING WAVELETS" Vol.3, No.4. pp14-23. Signal & Image Processing: An International Journal (SIPIJ). Retrieved on 3rd July, 2016.

Pandey A. K., Agarwal K., Haroon M. (2015) "A Hybrid Approach for Enriching Image using Mamdani Neuro -Fuzzy Technique and its Comparative Analysis" Vol. 121, No.19, International Journal of Computer Applications (0975-8887). Retrieved on 23rd June, 2016.

Patel, J., Pathak, K., (2014) "Implementation of the 5/3 Lifting 2D Discrete Wavelet Transform" Vol 2, Issue 3 ISSN: 2321-9939. International Journal of Engineering Development and Research (www.ijedr.org). Retrieved on 25th November, 2015.

Patil1, Apoorva G., Ankita V., Shikhar S., (2013) "Audio and Speech Compression Using DCT and DWT Techniques" Vol. 2, Issue 5, International Journal of Innovative Research in Science, Engineering and Technology. www.ijirset.com. Retrieved on 10th June, 2014.

Paulinas and Ušinskas (2007) "A Survey of Genetic Algorithms Applications for Image Enhancement and Segmentation" Vol.36, No.3, P.178-201. ISSN 1392 – 124X INFORMATION TECHNOLOGY AND CONTROL. Retrieved on 13th June, 2014. http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.120.4391&rep=rep1&type=pdf i.

Paulinas and Ušinskas (2007) "A Survey of Genetic Algorithms Applications for Image Enhancement and Segmentation" Vol.36, No.3, P.178-201. ISSN 1392 – 124X INFORMATION TECHNOLOGY AND CONTROL. Retrieved on 31st September, http://citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.120.4391&rep=rep1&type=pdf.

Pennebaker W. B. and Mitchell, J. L. (1993) “JPEG Still Image Data Compression Standard,” Vol.2, No.4, pp. 67 Newyork: International Thomsan Publishing. Retrieved on 24th September, 2014

PEREIRA, F. (2006) “The MPEG4 Standard: Evolution or Revolution” Retrieved on 16th January, 2016. http://homepages.inf.ed.ac.uk/rbf/CVonline/LOCAL_COPIES/AV0506/s0561282.pdf.

Pereira, F., (2015) “BASICS ON DIGITAL AUDIO AND VIDEO REPRESENTATION Audioand Video Communication” Retrieved on 25th August. http://www.iitrpr.ac.in/sites/default/files/research_publication-rajibkjha.pdf.

Preet Kaur, Geetu lalit (2012) "Comparative Analysis of DCT, DWT &LWT for Image Compression" International Journal of Innovative Technology and Exploring Engineering (IJITEE) ISSN: 2278-3075, Volume-1, Issue-3.Retrieved on 21st September, 2016 http://www.ijser.org/paper/A-Robust-Image-Watermarking-based-on-LWT-and-Spread-Spectrum.html.

Priyanka Singh, Priti Singh, Rakesh Kumar Sharma,(2011) “JPEG Image Compression based on Biorthogonal, Coiflets and Daubechies Wavelet Families”, Applications (0975 – 8887), Volume 13– No.1. International Journal of Computer.

Rabiul Islam, S. Md., Huang, X., Ou, K. L. (2015) “IMAGE COMPRESSION BASED ON COMPRESSIVE SENSING USING WAVELET LIFTING SCHEME” Vol.7, No.1. The International Journal of Multimedia & Its Applications (IJMA). Retrieved on 25th November, 2015.

Raghavendra, G., Anita, R. (2013) “Implementation of Lifting-Based Two Dimensional Discrete Wavelet Transform on FPGA Using Pipeline Architecture” Vol. 5, No. 5. P. 230. ISSN: 2231-2803. International Journal of Computer Trends and Technology (IJCTT). Retrieved from http://www.ijcttjournal.org. On 25TH August, 2015.

Raid, A.M., (2014) “Jpeg Image Compression Using Discrete Transform - A Survey” International Journal of Computer Science & Engineering Survey (IJCSES) Vol.5, No.2, pp. 39-47. Retrieved from http://airccse.org/journal/ijcses/current2014.html. On 10th June, 2015.

Rajib, J., Kumar, Rajlaxrni Chouhan, Prabir Kumar Biswas, and Kiyoharu Aizawa. (2012) “Internal noise-induced contrast enhancement of dark images.” In Image Processing (ICIP) 19th IEEE International Conference. pp. (973-976). IEEE.Retrieved on 25th August, 2016. http://www.iitrpr.ac.in/sites/default/files/research_publication-rajibkjha.pdf

Rajkumar, T. M. P. and V Latte Mrityunjaya (2011) “ROI Based Encoding of Medical Images: An Effective Scheme Using Lifting Wavelets and SPIHT for Telemedicine” Vol. 3, No. 3. pp234-242. International Journal of Computer Theory and Engineering. Retrieved on 20th November, 2015. http://vosvrdaweb.utia.cas.cz/cykly/IEEEwavelet.pdf

Roopashree.S, Sachin S. Rohan R. S. (2012) "Enhancement and Pre-Processing of Images Using Filtering" Volume-1, Issue-5, and ISSN: 2249 – 8958. International Journal of Engineering and Advanced Technology (IJEAT). Retrieved on 5th April, 2016.

Rudin, L., J.M. Morel, et al (2014) "Sampling Theory for Digital Video Acquisition: The Guide for the Perplexed User" Vol.7, No.5, p246. http://worldwidescience.org/topicpages/d/digital+video+disc.html.Retrieved on 16th May, 2016.

Sangkeun Lee (2008) "Blind cross colour noise detection and reduction "Volume: 54 Issue: 4, ISSN: 0098-3063, pp. 1825 -1829, IEEE Transactions. http://140.98.202.196/xpl/RecentIssue.jsp?punumber=30. Retrieved on 11th December, 2015.

Sengee, Nyarnlkhagva. And Heung Choi. (2008) "Brightness preserving weight clustering his togram equalization." Consumer Electronics, IEEE Transactions on 54, no. 3:1329-1337.Retrieved on 14th January, 2016.

Sheebha, S.P. and Sriraman, L,(2012) "A Modified Algorithm for Removal of Salt and Pepper Noise inColour Images", pp 356-361. IEEE Third International Conference on Intelligent Systems, Modelling and Simulation (ISMS). Retrieved on 19th May, 2016. http://ieeexplore.ieee.org/Xplore/defdeny.jsp?url=http%3A%2F%2Fieeexplore.ieee.org%2Fstamp%2Fstamp.jsp%3Ftp%3D%26arnumber%3D6169729&denyReason= 131&arnumber=6169729&productsMatched=null&userType=inst.

Siddavatam R., Sood A., Jayasree S. P., Ghrera S. P. (2011) "An Intelligent Recursive Algorithm for 95% Impulse Noise Removal in Grey scale and Binary Images using Lifting Scheme" Vol 1, pp19-21 Proceedings of the World Congress on Engineering and Computer Science. Retrieved on 15th February, 2016.

Singh, G. M., et al (2013) "A Review of Image Enhancement Techniques in Image Processing" Vol. 7, No. 2, p278-288.Retrieved on 6th July, 2016. VintaSoftImaging.Net.

Staelens, N. et al. (2013) "Constructing a No-Reference H.264/AVC Bitstream-Based Video Quality Metric Using Genetic Programming-Based Symbolic Regression" Volume: 23 Issue: 8, ISSN: 1051-8215, pp. 1322 - 1333, Publisher: IEEE. Retrieved on 13th May, 2016. http://ieeexplore.ieee.org/xpl/RecentIssue.jsp?punumber=76.

Staelens, N., G. V. Wallendael., (2011) "No-Reference Bit stream-based Visual Quality Impairment Detection for High Definition H.264/AVC Encoded Video Sequences" IEEE TRANSACTIONS ON BROADCASTING. Vol. 3, No. 5, p569. Retrieved on May, 2014.

Strang G., (1999) "The Discrete Cosine Transform," SIAM Review, Volume 41, Number 1, pp.135-147. Retrieved on 16th March, 2014. http://epubs.siam.org/doi/abs/10.1137/S0036144598336745#citedBySection

Syed Ali Khayam (2003) "The Discrete Cosine Transform DCT): Theory and Application" Department of Electrical & Computer Engineering Michigan State UniversityVol.7, No.5, p246.Retrieved on March, 2014.

Takahashi, K., (2011) "Rate-distortion analysis of super-resolution image/video decoding" ISSN: 1522-4880, pp. 1629 - 1632. 18th IEEE International Conference.Ret

rieved on November, 2015. http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=6094293.

Tomas F., (2010) "A Complete Video Coding Chain Based On Multi-Dimensional Discrete Cosine Transform" Department of Radio Electronics, Brno University of Technology, Purkynova 118, 61200 Brno, Czech Republic. RADIO ENGINEERING, VOL. 19, NO. 3 .pp421-428. Retrieved on February, 2016. http://www.radioeng.cz/fulltexts/2010/10_03_421_428.pdf.

Vasicek, Z. and Sekanina, L. (2010) "Reducing the Area on a Chip Using a Bank of Evolved Filters" Vol. 2, No. 5, p.346. Retrieved on March, 2016. vasicek@fit.vutbr.cz sekanina@fit.vutbr.cz

Verma R. and Ali J. (2013) "A Comparative Study of Various Types of Image Noise and Efficient Noise Removal Techniques" Volume 3, ISSN: 2277 128X. International Journal of Advanced Research in Computer Science and Software Engineering.Retrieved on August, 2016. Available online at: www.ijarcsse.com

Vishnu T. P., (2013) "Scalability Techniques of MPEG-2 Standard for Video Compression" Volume 3, Issue 1.International Journal of Advanced Research in Computer Science and Software Engineering. Available online at: www.ijarcsse.com Retrieved on 27/01/2015

Vishwakarma, A. K. and Mishra, A. (2012) "Colour Image Enhancement Techniques: A Critical Review" Vol. 3 No. 1, p554-678. Indian Journal of Computer Science and Engineering (IJCSE).Retrieved on July, 2016.

Wallace G. K, (1991) "The JPEG Still Picture Compression Standard," Communications of the ACM. Retrieved on May, 2014. http://web.stanford.edu/class/ee398a/handouts/papers/Wallace%20-%20JPEG%20-%201992.pdf.

Walter Fischer (2010) "Digital Video and Audio Broadcasting Technology A Practical Engineering Guide" Second Edition. Retrieved from http://www.amazon.com/Digital-Video-Audio-Broadcasting-Technology/dp/3642116116.Retrieved on October, 2015.

Wang Jian, Qin Wu and Dong Xiao-gang, (2011) "A new adaptive weight algorithm for salt and pepper noise removal", pp 26-29. IEEE International Conference on Consumer Electronics, Communications and Networks. Retrieved on November, 2016. http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=5768748.

Wang Jian, Qin Wu and Dong Xiao-gang,(2011) "A new adaptive weight algorithm for salt and pepper noise removal", pp 26-29. IEEE International Conference on Consumer Electronics, Communications and Networks. Retrieved on May, 2016. http://ieeexplore.ieee.org/xpl/articleDetails.jsp?arnumber=5768748.

Wang, Y., Ostermann, J and Zhang, Y. Q., (2001) "Video Processing and Communications" Vol. 3, pp.22-30. Retrieved from http://www.associatechair.ece.ufl.edu/syllabi/2014-8-Fall/6512-6562DWuF14.pdf on1st JANUARY, 2015.

Wei-Yi Wei, (2010) "Digital Video Compression Fundamentals and Standards" Pg 1. E-mail: r97942024@ntu.edu.twRetrieved from http://disp.ee.ntu.edu.tw/meeting/%E7%B6%

AD%E6%AF%85/Digital%20Video%20Compression%20Fundamentals%20and%20Standards/Digital%20Video%20Compression%20Fundamentals%20and%20Standards.pdf. On 3rd May, 2015

Xiuqi Li and Borko Furht (2000) "An Approach to Image Compression Using Three-Dimensional DCT" Department of Computer Science and Engineering, Florida Atlantic University, Boca Raton, FL 33431 Email:xli@cse.fau.edu, borko@cse.fau.edu. htt/citeseerx.ist.psu.edu/viewdoc/download?doi=10.1.1.135.4268&rep=rep1&type=pdf. Retrieved on January, 2014.

Yang, X.-S (2008) "Nature-Inspired Metaheuristic Algorithms". Luniver Press. Vol.7, No.5, p246.Retrieved on November, 2016.

Yang, X.-S. (2010). Firefly algorithm, Levy flights and global optimization Research and Development in Intelligent Systems XXVI (pp. 209-218): Springer.Retrieved on November, 2016.

Yao Wang, (2016) "Image and Video Processing Colour Image Perception and Representation"Volume 41, Number 1, pp.135-147.Retrieved on November, 2016.

Yi Wan,(2010) "A novel quadratic type variational method for efficient salt-and-pepper noise removal", ISSN:1945-7871, pp 1055-1060. IEEE International Conference on Multimedia and Expo (ICME). Retrieved on November, 2016. http://ieeexplore.ieee.org/xpl/freeabs_all.jsp?arnumber=5583306&abstractAccess=no&userType=inst.

Young, I. T., J. J. Gerbrands., (2007) "Fundamentals of Image Processing"Vol.6, pp.24-33. Retrieve from https://www.academia.edu/5715717/INCREMENTAL_SEGMENTATION_OF_LIDAR_POINT_CLOUDS_WITH_AN_OCTREE STRUCTURED_VOXEL_SPACE. Retrieved on January, 2015.

Ze-Nian L, Mark S. D., Jiangchuan L., (2004) "Fundamentals of Multimedia" Second Edition. Texts in Computer Science. Retrieved on April, 2014. http://www.just.edu.jo/~qabuein/courses/cis302/book.pdf.

Zhang D. and Liang J. (2015) "View Synthesis Distortion Estimation with a Graphical Model and Recursive Calculation of Probability Distribution" VOL. 25, NO 5, IEEE TRANSACTIONS ON CIRCUITS AND SYSTEMS FOR VIDEO TECHNOLOGY.Retrieved on November, 2016.

Zhang D., Liang J. I., Singh I. (2013) "Fast transmission distortion estimation and adaptive error protection for H.264/AVC-based embedded video conferencing systems" Vol.25, Issue 5, pp417-429. Signal Processing: Image Communication.Retrieved on December, 2016.

Zhengying, C. et al (2014) "Quality Assessment for Comparing Image Enhancement Algorithms" Open Access version, provided by the Computer Vision Foundation.yhtiang@pku.edu.cn.Retrievedfrom http://dl.acm.org/citation.cfm?id=2680123. Retrieved on 2nd April, 2015.

Zhiyuan S., (2012) "Research on quality assessment metric based on H.264/AVC bitstream" ISSN: 2163-5048, pp. 1 - 5. Publisher: IEEE. Retrieved on November, 2016. http://ieeexplore.ieee.org/xpl/mostRecentIssue.jsp?punumber=6313472

Appendix A
M-file implementation

```
clc
close all
clear all
FILM=VideoReader('video_Name.avi');
MOVIE=read(FILM);
size(MOVIE);
vidHeight = FILM.Height;
MOVIE_cropped=MOVIE(1:vidHeight/2,:,:,:);
aviFILM = VideoWriter('cropped_video.avi','nameOfVideo AVI');

open(aviFILM);
writeVideo(aviFILM,MOVIE_cropped);
close(aviFILM);
vidFILM = VideoWriter('cropped_video.avi');
open(vidFILM);
for k=1:FILM.NumberOfFrames
  imshow(MOVIE(k));
  writeVideo(vidFILM,currFrame);
end

clear all
close all
clc
format longg;
format compact;
PS=imread('Akiyo.bmp');
imshow(PS);
imhist(rgb2grey(PS))
figure
imshow(rgb2grey(PS))
title('NTA First Sample')
xlabel('range of pixel values')
ylabel('Amount of Intesity')
% imwrite(rgb2grey(PS),'Original.bmp');
PS=rgb2grey(PS);
figure
imhist(PS)
```

```
title('Histogram of Akiyo Sample before Enhacement');
figure
imshow(edge(PS,'canny'))
% PS=imnoise(PS,'salt & pepper');
[m,n]=size(PS);
GP=zeros(1,256);
for k=0:255
    GP(k+1)=length(find(PS==k))/(m*n);
end
 figure,bar(0:255,GP,'b')
 title('Histogram of Akiyo Sample before Enhacement')
 xlabel('range of pixel values')
ylabel('Amount of Intesity')
%  xlabel('')
%  ylabel('')
S1=zeros(1,256);
for i=1:256
for j=1:i
        S1(i)=GP(j)+S1(i);
end
end
S2=round((S1*256)+0.5);
for i=1:256
    GPeq(i)=sum(GP(find(S2==i)));
end
figure,bar(0:255,GPeq,'k')
title('Histogram of Akiyo After Enhancement')
xlabel('range of pixel values')
ylabel('Amount of Intesity')
PA=PS;
for i=0:255
    PA(find(PS==i))=S2(i+1);
end
[row,col]=size(PA);
% figure,bar(0:255,PA,'g')
% title('Histogram of Akiyo After Enhancement')
figure,imshow(PA)
title('Enhanced Image of Akiyo Benchmark Sample')
Image=imread('unspecified13.jpg');
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%
greymage=PS;
A=imresize(PS,[row,col]);
[rows,columns] = size(A);
% subplot(2, 2, 1);
% imshow(A, []);
% title('Original Grey Scale Image', 'FontSize', fontSize);
set(gcf, 'Position', get(0,'Screensize'));
Image2=PA;
% Display the second image.
```

```
% subplot(2, 2, 2);
% imshow(noisyImage, []);
% title('Noisy Image', 'FontSize', fontSize);
SQRERROR = (double(A) - double(Image2)).^ 2;
% subplot(2, 2, 3);
% imshow(SQRERROR, []);
% title('Squared Error Image', 'FontSize', fontSize);
mse = sum(sum(SQRERROR))/(rows * columns);
PSNR = 10*log10( 256^2 / mse);
output = sprintf('The mean square error is %.2f.\nThe PSNR = %.2f', mse, PSNR);
msgbox(output);
%     FRAME=imread('unspecified13.jpg');
%     FRAME=im2double(FRAME);
%     h1=1/9*ones(3,3);
%     FRAMEf1=imfilter(FRAME,h1,'replicate');
%     h2=1/25*ones(5,5);
%     FRAMEf2=imfilter(FRAME,h2,'replicate');
%     MSE1=mean(mean((FRAME-FRAMEf1).^2));
%     MSE2=mean(mean((FRAME-FRAMEf2).^2));
%     MaxI=1;% the maximum possible pixel value of the images.
%     PSNR1=10*log10((MaxI^2)/MSE1);
%     PSNR2=10*log10((MaxI^2)/MSE2);
%
```

Appendix B

M-file Implementation for coloured frames

```
close all;clc;
PS=imread('Specify Frames.*jpg,.*png,.*bmp,...');
figure
imshow(PS)
title('Akiyo Before Enhancement')
[m,n]=size(PS);
GP=zeros(1,256);
for k=0:255
    GP(k+1)=length(find(PS==k))/(m*n);
end
 figure,bar(0:255,GP,'b')
 title('Histogram of Akiyo before after')
xlabel('Pixels Value')
ylabel('Amount of Intesity')
 S1=zeros(1,256);
for i=1:256
for j=1:i
        S1(i)=GP(j)+S1(i);
end
end
S2=round((S1*256)+0.5);
for i=1:256
    GPeq(i)=sum(GP(find(S2==i)));
end
figure,bar(0:255,GPeq,'k')
title('Histogram of Akiyo After Enhancement')
 xlabel('Pixels Value')
 ylabel('Amount of Intesity')
PA=PS;
for i=0:255
    PA(find(PS==i))=S2(i+1);
end
figure
imshow(PA)
title('Akiyo After Enhancement Frame')
%%
[rows,columns]=size(PA);
% A=imresize(gdiff1,[row,col]);
% [rows,columns] = size(A);
set(gcf, 'Position', get(0,'Screensize'));
Image2=PA;
squaredErrorImage = (double(PA) - double(PS)).^ 2;
mse = sum(sum(squaredErrorImage))/(rows * columns);
PSNR = 10*log10( 256^2 / mse);
message = sprintf('The mean square error is %.2f.\nThe PSNR = %.2f', mse, PSNR);
msgbox(message);
```

Appendix C
M-file Implementation of Discrete Wavelet Transform

```
clc;close all;
clear all;
map = grey(256);

n=1:256;
x=sin((pi/64)*n)+0.01*(n-100);
function g = dwt(f,h)
N = length(h);  %L = length(f);
c = f;
h0  = fliplr(h);                    % Scaling filter
h1 = h;  h1(1:2:N) = -h1(1:2:N);        % Wavelet filter
L = length(c);
% The DWT
% get db filter length 6
[Lo_D,Hi_D,Lo_R,Hi_R] = wfilters('bior5.5');
figure(1);
subplot(2,2,1); plot(Lo_D);
title('Decomposition LOW-pass filter')
subplot(2,2,2); plot(Hi_D);
title('Decomposition HIGH-pass filter')
subplot(2,2,3); plot(Lo_R);
title('Reconstruction LOW-pass filter')
subplot(2,2,4); plot(Hi_R);
title('Reconstruction HIGH-pass filter')
% 1D dwt and idwt
g = dwt(x,Lo_R);
y = idwt(g, Lo_R);
figure(2);
% subplot(3,1,1); plot(x);
subplot(2,1,1); plot(g,'LineWidt',2);
title('Discrete Wavelet Transform Behaviour')
subplot(2,1,2); plot(y,'LineWidt',2);
title('Inverse Discrete Wavelet Transform Behaviour')
% 2D dwt and idwt
imData=imread('Akiyo.bmp');
figure(3);
imshow(imData);

% first dwt
[N, M]=size(imData);
% dwt in row
dwt_row_image=zeros(N, M);
tmpData=zeros(1, M);
for i=1:N
   tmpData(1, 1:M)=imData(i, 1:M);
   tmpData(1, 1:M)=dwt(tmpData, Lo_R);
```

```
   dwt_row_image(i, 1:M)=tmpData(1, 1:M);
end

%showData=zeros(N, M);
%showData(1:N, 1:M/2)=wcodemat(dwt_row_image(1:N, 1:M/2),256,'mat'); %wcodemat is
command to normalize the image range
%showData(1:N, M/2+1:M)=wcodemat(dwt_row_image(1:N, M/2+1:M),256,'mat');
figure(4);
imshow(dwt_row_image, map);


% dwt in column
tmpData=zeros(1, N);
dwt1_imData=zeros(N, M);
for i=1:M
   tmpData(1, 1:N)=dwt_row_image(1:N, i)';
   tmpData(1, 1:N)=dwt(tmpData, Lo_R);
   dwt1_imData(1:N, i)=tmpData(1, 1:N)';
end
showData=zeros(N, M);
showData(1:N/2, 1:M/2)=wcodemat(dwt1_imData(1:N/2, 1:M/2),256,'mat');
showData(1:N/2, M/2+1:M)=wcodemat(dwt1_imData(1:N/2, M/2+1:M),256,'mat');
showData(N/2+1:N, 1:M/2)=wcodemat(dwt1_imData(N/2+1:N, 1:M/2),256,'mat');
showData(N/2+1:N, M/2+1:M)=wcodemat(dwt1_imData(N/2+1:N, M/2+1:M),256,'mat');

figure(5);
imshow(dwt1_imData, map);
c = [c(mod((-(N-1):-1),L)+1) c];          % Make periodic
subplot(3,1,1);
plot(f);
subplot(3,1,2);
plot(c);
d = conv(c,h1);   d = d(N:2:(N+L-2));     % Convolve & d-sample
c = conv(c,h0);   c = c(N:2:(N+L-2));     % Convolve & d-sample
g = [c,d];
tmpData1=zeros(1, N);
idwt1_imData=zeros(N, M);
for i=1:M

   tmpData1(1, 1:N)=dwt1_imData(1:N, i)';
   tmpData1(1, 1:N)=idwt(tmpData1, Lo_R);
   idwt1_imData(1:N, i)=tmpData1(1, 1:N)';
end
figure(6);
imshow(idwt1_imData,map);
idwt_row_image=zeros(N, M);
tmpData1=zeros(1, M);
for i=1:N
   tmpData1(1, 1:M)=idwt1_imData(i, 1:M);
   tmpData1(1, 1:M)=idwt(tmpData1, Lo_R);
```

```
   idwt_row_image(i, 1:M)=tmpData1(1, 1:M);
end
figure(7);
imshow(idwt_row_image,map);
function f = idwt(g,h)

L = length(g);   N = length(h);
h0 = h;
h1 = fliplr(h);  h1(2:2:N) = -h1(2:2:N);
LJ = L/2;
c  = g(1:LJ);
w  = mod(0:N/2-1,LJ)+1;
d  = g(LJ+1:L);
cu(1:2:L+N) = [c c(1,w)];
du(1:2:L+N) = [d d(1,w)];
c  = conv(cu,h0) + conv(du,h1);
c  = c(N:N+L-1);
f = c;
```

Appendix D

M-file Implementation of Discrete Cosine Transform

```
close all
clear all
clc
Frames = VideoReader('foreman291.yuv'); % this reads the input video file form the
directory
 get(Frames); % Extract the numbers of frames in the image
 nFrames = Frames.NumberOfFrames;
for k = 1 : nFrames % select form the first frame to the last that is been extracted
   Read_extracted = read(Frames, k); % read the extracted file
%create the file name
   j=k-1;
   Storage_location='file_compressed\';suffix='.bmp';%'.png';
   file=[Storage_location,num2str(j),suffix];

% Create Folder if already not present
   Folder=fullfile('file_compressed'); % Name of Folder
if (exist(Folder) == 0) % Checkk already Present
      mkdir (Folder);       % Create if already not present
end
%Save the image file for each frame
   imwrite(Read_extracted,file,'BMP');
end
pathname = 'file_compressed\';
dirlist = dir( [pathname '*.bmp','*.png','gif'] ); % this kist the files in the directory (total
number of frames extracted from the image)
Image_format='bmp';
for xx = 1:length(dirlist)
    I = imread([pathname, dirlist(xx).name]);
size(I);  %[height, width] = size(A) where Height = 1 and width = 2
width=size(I,2);
%N_Samples_1 = floor(width / 1);
N_Samples_2 = floor(width / 2);
%N_Samples_4 = floor(width / 4);
%N_Samples_8 = floor(width / 8);
for k=1:3  % all colour layers: RGB
for i=1:size(I, 1)  %  returns the number of all rows
      rowDCT=dct(double(I(i,:,k)));
      Image_compression(i,:,k) = idct(rowDCT(1:N_Samples_2), width);
end
end
Image_compression_format=uint8(Image_compression);
Arrange_colour=strcat('file_compressed\',int2str(xx),'.',Image_format);
imwrite(Image_compression_format,Arrange_colour)
W=whos('Image_compression');
S=numel(I) ;W1= W.bytes; %Number of elements in an array or subscripted array expression
CR=W1*8/S;
```

```
end
path ='file_compressed\';
files = dir(fullfile(path,'*.bmp'));
writerObj = VideoWriter('compressed','MPEG-4');
open(writerObj)
for k = 1:nFrames
   image = ((imread(fullfile(path,files(k).name))));
    writeVideo(writerObj,image);
end
close(writerObj);
Compressed_video = VideoReader('compressed1.mp4');
nFrames = Compressed_video.NumberOfFrames;
vidHeight = Compressed_video.Height;
vidWidth = Compressed_video.Width;
% Preallocate movie structure.
mov(1:nFrames) = struct('cdata', zeros(vidHeight,vidWidth,3,'uint8'),'colourmap',[]);
% Read one frame at a time.
for k = 1 : nFrames
   Read_extracted = read(Compressed_video, k);
end
% Size a figure based on the video's width and height.
hf = figure;
set(hf, 'position', [150 150 vidWidth vidHeight])

% Play back the movie once at the video's frame rate.
movie(hf, mov, 1, Compressed_video.FrameRate);
video1 = ('foreman.avi);
video2 = ('compressed.mp4');
C_R = Compratio(video1, video2)
[MSE,Psnr]=CalMSE(video1,video2)
```

S

Appendix E

M-file Implementation of Lifting Wavelet Transform

```
function y = LWT(x, nlevel, wname)
error(nargchk(2, 3, nargin));
if nargin < 3
   wname = 'cdf97';
end
% check nlevel
if ~isreal(nlevel) || ~isnumeric(nlevel) || round(nlevel)~=nlevel
   error('WAVELIFT:InArgErr', ['The 2nd argument shall be ' ...
      'a real and numeric integer.']);
end
% check x
if ~isreal(x) || ~isnumeric(x) || (ndims(x) > 2)
   error('WAVELIFT:InArgErr', ['The first argument must' ...
      ' be a real, numeric 2-D or 1-D matrix.']);
end
if isinteger(x)
   x = double(x);
end
% check wname
if ~ischar(wname) || ~ismember(wname, {'cdf97', 'spl53'})
   error('WAVELIFT:InArgErr', ['The last argument must be a wavelet ' ...
      'name. \nCurrently only "cdf97" and "spl53" are supported.']);
end
switch wname
   case 'cdf97'
     lamdaz=struct('coeff',{[-1.5861343420693648,-1.5861343420693648],...
                    [-0.0529801185718856,-0.0529801185718856],...
                    [ 0.8829110755411875, 0.8829110755411875],...
                    [ 0.4435068520511142,0.4435068520511142]},...
              'zorder', {[0 1], [0 -1], [0 1], [0 -1]});
     L=struct('lamdaz',lamdaz,'K',[1/1.230174104914, 1.230174104914/2]);

     mode='lossy';
   case 'spl53'
     lamdaz = struct('coeff',  {[-.5, -.5], [.25 .25]}, ...
                'zorder', {[ 0,   1 ], [ 0, -1 ]});
     L = struct('lamdaz', lamdaz, 'K', [1, 1/2]);
     mode='lossless';
end
clear lamdaz;
y = x;
if nlevel > 0
   for i = 1:nlevel
      sx = size(x);
      [temp0, temp1] = colwavelift(x, L, 'd', mode);
```

```
        [temp0, temp1] = colwavelift([temp0; temp1]', L, 'd', mode);
        temp = [temp0', temp1'];
        y(1:sx(1), 1:sx(2)) = temp;
        x = temp(1:ceil(sx(1)/2), 1:ceil(sx(2)/2));
        if size(x,1)<=1 && size(x,2)<=1 && i~=nlevel
            warning('WAVELIFT:InArgDegrade', ['Only decompose to ' ...
                num2str(i) '-level instead of ' num2str(nlevel) ...
                ', \nas the approximation coefficients at ' num2str(i) ...
                '-level has row or/and column of length 1.']);
            break
        end
    end
else
    sx = size(x);
    nl = -nlevel;
    while sx(1)/2^nl<=1/2 && sx(2)/2^nl<=1/2,  nl = nl-1;  end
    if nl ~= -nlevel
        warning('WAVELIFT:InArgDegrade', ['Only reconstruct to ' ...
            num2str(nl) '-level instead of ' num2str(-nlevel) ...
            ', \n as the approximation coefficients at ' num2str(nl) ...
            '-level has row or/and column of length 1.']);
    end
     for i = 1 : nl
         sTarget = ceil(sx/2^(nl-i));
        target = y(1:sTarget(1), 1:sTarget(2));
         sLL = ceil(sTarget/2);
        temp0 = target(:, 1: sLL(2));
        temp1 = target(:, sLL(2)+1:end);
        temp = colwavelift(temp0', temp1', L, 'r', mode);
        temp = temp';
        temp0 = temp(1: sLL(1), :);
        temp1 = temp(sLL(1)+1 :end, :);
        temp = colwavelift(temp0, temp1, L, 'r', mode);
        y(1:sTarget(1), 1:sTarget(2)) = temp;
    end
end
function [y, opty] = colwavelift(x, optx, L, direction, mode)
error(nargchk(4, 5, nargin));
if nargin == 4
    mode = direction;
    direction = L;
    L = optx;
end
% check optx
if nargin == 5
    if size(optx, 2) ~= size(x, 2)
        error('COLWAVELIFT:InArgErr', ['The first two arguments must' ...
        ' have the same column numbers.']);
    end
    if ~isreal(optx) || ~isnumeric(optx) || (ndims(optx) > 2)
```

```
        error('COLWAVELIFT:InArgErr', ['The second arguments must' ...
        ' be a real, numeric 2-D or 1-D matrix.']);
    end
end
% check x
if ~isreal(x) || ~isnumeric(x) || (ndims(x) > 2)
    error('COLWAVELIFT:InArgErr', ['The first argument must' ...
    ' be a real, numeric 2-D or 1-D matrix.']);
end
% check direction
if ischar(direction) && ismember(direction, {'d', 'dec', 'f', 'forward'})
    direction = 'd';
elseif ischar(direction) && ismember(direction, {'r', 'rec', ...
        'b', 'backward', 'i', 'inverse'})
    direction = 'r';
else
    error('COLWAVELIFT:InArgErr', ['For the last argument, use "d" ' ...
        'to denote decomposition/forward lifting \n or "r" for ' ...
        'reconstruction/inverse lifting.']);
end
% check mode
if ~ischar(mode) || ~ismember(mode, {'lossy', 'lossless'})
    error('COLWAVELIFT:InArgErr', ['The last argument must be either' ...
        ' "lossy" or "lossless" to decide the reversibility.']);
end
if ~isstruct(L)
    error('COLWAVELIFT:InArgErr', ['Use a Matlab data type' ...
        ' "structure" to denote the lifting structure. \n Type' ...
        ' "help colwavelift" to see the way the structure shall be' ...
        ' organized.']);
end
if nargin == 4 % without optx
    y0 = x(1:2:end, :);
    y1 = x(2:2:end, :);
    y = x;
else % nargin == 5, ie, with optx
    y0 = x;
    y1 = optx;
    y = zeros(size([y0; y1]));
    y(1:2:end, :) = y0;
    y(2:2:end, :) = y1;
end
clear x optx;
sy = size(y);
if size(y, 1) > 1
    ry0 = size(y0, 1);
    ry1 = size(y1, 1);
   len = length(L.lamdaz);
    eval('plusminus = 1;');
    temp = [L.lamdaz.zorder];
```

```
rshift = - sum( temp(find(temp<0)) ) * 2;
lshift = sum(temp) * 2 + rshift;
if ry1 ~= ry0
   rshift = rshift + 1;
end
% extension
for i = 1: max(lshift, rshift)
   y = [y(2*i, :); y; y(sy(1)-1, :)];
end
if rshift > lshift
   y = y(rshift-lshift+1:end, :);
elseif rshift < lshift
   y = y(1:end+rshift-lshift, :);
end
y0 = y(1:2:end, :);
y1 = y(2:2:end, :);
clear y;
if strcmp(direction, 'r')
   % move lifting gains to front
   y0 = y0 / L.K(1);  L.K(1) = 1;
   y1 = y1 / L.K(2);  L.K(2) = 1;
   if rem(len,2) == 0
      temp = y0;
      y0 = y1;
      y1 = temp;
   end
   for i = 1: floor(len/2)
      temp = L.lamdaz(i).coeff;
      L.lamdaz(i).coeff = L.lamdaz(len-i+1).coeff;
      L.lamdaz(len-i+1).coeff = temp;
      temp = L.lamdaz(i).zorder;
      L.lamdaz(i).zorder =  L.lamdaz(len-i+1).zorder;
      L.lamdaz(len-i+1).zorder = temp;
   end
   eval('plusminus = -1;');
end
for i = 1: len
   eval('yconv = zeros(size(y0));');
   for j = 1: length(L.lamdaz(i).zorder)
      eval(['yconv = yconv +  circshift(y' num2str(rem(i-1,2)) ...
            ', -L.lamdaz(i).zorder(j)) * L.lamdaz(i).coeff(j);']);
   end
   if strcmp(mode, 'lossy')
      eval(['y' num2str(rem(i,2)) ' = y' num2str(rem(i,2)) ...
         ' + yconv * plusminus;']);
   else % 'lossless'
      eval(['y' num2str(rem(i,2)) '= y' num2str(rem(i,2)) ...
         ' + floor(yconv + .5) * plusminus;']);
   end
end
```

```
    if strcmp(direction, 'r') && rem(len,2) == 0
        temp = y0;
        y0 = y1;
        y1 = temp;
    end
    y0 = y0(1+lshift/2 : end-floor(rshift/2), :) * L.K(1);
    y1 = y1(1+lshift/2 : end-floor(rshift/2), :) * L.K(2);
    if ry1 < ry0
        y1 = y1(1:end-1, :);
    elseif ry1 > ry0
        y0 = y0(1:end-1, :);
    end
end
if nargout == 1
    y = zeros(sy);
    y(1:2:end, :) = y0;
    if sy(1) > 1
        y(2:2:end, :) = y1;
    end
elseif nargout == 2
    y = y0;
    opty = y1;
else
    error('COLWAVELIFT:OutArgErr', ['Invalid output argument numbers. '...
        'Use "help wavecdf97lift" for help.']);
end% for
```

Appendix F

m-Implementation of Fire Fly Algorithm

```
function FFA
% parameters [n N_iteration alpha betamin gamma]
para=[40 500 0.5 0.2 1];
format long;

help FOA.m
% Simple bounds/limits
disp('Solve the simple spring design problem ...');
Lb=[0.05 0.25 2.0];
Ub=[2.0 1.3 15.0];

% Initial random guess
u0=Lb+(Ub-Lb).*rand(size(Lb));

[u,fval,NumEval]=ffa_mincon(@cost,@FOAcompress,u0,Lb,Ub,para);

% Display results
bestsolution=u
bestojb=fval
total_number_of_function_evaluations=NumEval

function z=cost(x)
z=(2+x(3))*x(1)^2*x(2);

%%% Put your own constraints here ----------%%%
function [g,geq]=FOAcompress(x)

geq=[];

% Start FA
function [nbest,fbest,NumEval]...
        =foa_mincon(fhandle,nonhandle,u0, Lb, Ub, para)
% Check input parameters (otherwise set as default values)
if nargin<6, para=[20 50 0.25 0.20 1]; end
if nargin<5, Ub=[]; end
if nargin<4, Lb=[]; end
if nargin<3,
disp('Usuage: FA_mincon(@cost, @FOAcompress,u0,Lb,Ub,para)');
end

n=para(1);  MaxGeneration=para(2);
alpha=para(3); betamin=para(4); gamma=para(5);

% Total number of function evaluations
NumEval=n*MaxGeneration;
```

```
% Check if the upper bound & lower bound are the same size
if length(Lb) ~=length(Ub),
   disp('Simple bounds/limits are improper!');
   return
end

% Calcualte dimension
d=length(u0);

% Initial values of an array
zn=ones(n,1)*10^100;
% ------------------------------------------------
% generating the initial locations of n fireflies
[ns,Lightn]=init_ffa(n,d,Lb,Ub,u0);

% Iterations or pseudo time marching
for k=1:MaxGeneration,     %%%%% start iterations

% This line of reducing alpha is optional
 alpha=alpha_new(alpha,MaxGeneration);

% Evaluate new solutions (for all n fireflies)
for i=1:n,
   zn(i)=Fun(fhandle,nonhandle,ns(i,:));
   Lightn(i)=zn(i);
end

% Ranking fireflies by their light intensity/objectives
[Lightn,Index]=sort(zn);
ns_tmp=ns;
for i=1:n,
 ns(i,:)=ns_tmp(Index(i),:);
end

%% Find the current best
nso=ns; Lighto=Lightn;
nbest=ns(1,:); Lightbest=Lightn(1);

% For output only
fbest=Lightbest;

% Move all fireflies to the better locations
[ns]=ffa_move(n,d,ns,Lightn,nso,Lighto,nbest,...
      Lightbest,alpha,betamin,gamma,Lb,Ub);

end   %%%%% end of iterations

function [ns,Lightn]=init_ffa(n,d,Lb,Ub,u0)
  % if there are bounds/limits,
```

```
if length(Lb)>0,
   for i=1:n,
   ns(i,:)=Lb+(Ub-Lb).*rand(1,d);
   end
else
   % generate solutions around the random guess
   for i=1:n,
   ns(i,:)=u0+randn(1,d);
   end
end

% initial value before function evaluations
Lightn=ones(n,1)*10^100;

% Move all fireflies toward brighter ones
function [ns]=ffa_move(n,d,ns,Lightn,nso,Lighto,...
          nbest,Lightbest,alpha,betamin,gamma,Lb,Ub)
% Scaling of the system
scale=abs(Ub-Lb);

% Updating fireflies
for i=1:n,
% The attractiveness parameter beta=exp(-gamma*r)
   for j=1:n,
      r=sqrt(sum((ns(i,:)-ns(j,:)).^2));
      % Update moves
if Lightn(i)>Lighto(j), % Brighter and more attractive
   beta0=1; beta=(beta0-betamin)*exp(-gamma*r.^2)+betamin;
   tmpf=alpha.*(rand(1,d)-0.5).*scale;
   ns(i,:)=ns(i,:).*(1-beta)+nso(j,:).*beta+tmpf;
      end
   end % end for j

end % end for i

% Check if the updated solutions/locations are within limits
[ns]=findlimits(n,ns,Lb,Ub);

function alpha=alpha_new(alpha,NGen)
% alpha_n=alpha_0(1-delta)^NGen=10^(-4);
% alpha_0=0.9
delta=1-(10^(-4)/0.9)^(1/NGen);
alpha=(1-delta)*alpha;

% Make sure the fireflies are within the bounds/limits
function [ns]=findlimits(n,ns,Lb,Ub)
for i=1:n,
    % Apply the lower bound
  ns_tmp=ns(i,:);
  I=ns_tmp<Lb;
```

```
  ns_tmp(I)=Lb(I);

  % Apply the upper bounds
  J=ns_tmp>Ub;
  ns_tmp(J)=Ub(J);
  % Update this new move
  ns(i,:)=ns_tmp;
end

% ----------------------------------------
% d-dimensional objective function
function z=Fun(fhandle,nonhandle,u)
% Objective
z=fhandle(u);

z=z+getnonlinear(nonhandle,u);

function Z=getnonlinear(nonhandle,u)
Z=0;
% Penalty constant >> 1
lam=10^15; lameq=10^15;
% Get nonlinear constraints
[g,geq]=nonhandle(u);

% Apply inequality constraints as a penalty function
for k=1:length(g),
    Z=Z+ lam*g(k)^2*getH(g(k));
end
% Apply equality constraints (when geq=[], length->0)
for k=1:length(geq),
   Z=Z+lameq*geq(k)^2*geteqH(geq(k));
end

function H=getH(g)
if g<=0,
    H=0;
else
    H=1;
end

% Test if equalities hold
function H=geteqH(g)
if g==0,
    H=0;
else
    H=1;
end
```

Appendix E2

m-File of Objective Function

```
%
clc;
clear;
close all;
%% Select Image
Filter={'*.jpg;*.jpeg;*.png;.*bmp'};
[FileName, FilePath]=uigetfile(Filter);
pause(0.01);
if FileName==0
   return;
end
FullFileName=[FilePath FileName];
%% Load Image Data
Choices = {'RGB', 'HSV'};
ANSWER = questdlg('Select the colour coding:', ...
                  'Colour Coding', ...
                  Choices{1}, Choices{2}, ...
                  Choices{1});
pause(0.01);
img=imread(FullFileName);
img=im2double(img);
R=img(:,:,1);
G=img(:,:,2);
B=img(:,:,3);

X=[R(:) G(:) B(:)];
Z=X;

UseHSV = strcmpi(ANSWER, 'HSV');

if UseHSV
   Y=rgb2hsv(X);
   W=[3 1 2];
   for l=1:numel(W)
      Y(:,l)=Y(:,l)*W(l);
   end
   Z=Y;
end

%% Number of Desired Colours

ANSWER = inputdlg('Number of desired colours:','Colour Reduction',1,{'25'});
pause(0.01);

nColour = str2double(ANSWER{1});
```

```
%% Select Algorithm

Choices = {'ABC', 'Fire Fly Algorithm', 'SOM'};

ANSWER = questdlg('Select the clustering algorithm:', ...
            'Colour Coding', ...
            Choices{1}, Choices{2}, Choices{3}, ...
            Choices{1});
pause(0.01);

UseABC = strcmpi(ANSWER, Choices{1});
UseFFA = strcmpi(ANSWER, Choices{2});
UseSOM = strcmpi(ANSWER, Choices{3});

%% Perform Clustering

if UseABC
   Method = 'ABC';
   Options.MaxIter=1000;
   [IDX, C]=ABC(Z,nColour,'options',Options);
end
if UseFFA
   Method = 'Fire Fly Algorithm';
   [C, U]=FFA(Z,nColour);
   [MaxU, IDX]=max(U);
end
if UseSOM
   Method = 'SOM';
   NetSize=[floor(sqrt(nColour)) ceil(sqrt(nColour))];
   nColour = prod(NetSize);
   [IDX, C]=SOM(Z,NetSize);
end
%% Create Reduced Image
Z2=C(IDX,:);
if UseHSV
   Y2 = Z2;
   for l=1:numel(W)
      Y2(:,l)=Y2(:,l)/W(l);
   end
   X2=hsv2rgb(Y2);
else
   X2=Z2;
end
R2=reshape(X2(:,1),size(R));
G2=reshape(X2(:,2),size(G));
B2=reshape(X2(:,3),size(B));
img2=zeros(size(img));
img2(:,:,1)=R2;
img2(:,:,2)=G2;
```

```
img2(:,:,3)=B2;
%% Show Results
figure;
% subplot(1,2,1);
imshow(img);
title('Original Frame');
% subplot(1,2,2);
figure
imshow(img2);
title('Colour Reduced Frame');
[rows,columns]=size(img2);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
%%
% A=imresize(gdiff1,[row,col]);
% [rows,columns] = size(A);
set(gcf, 'Position', get(0,'Screensize'));
% Image2=img2;
squaredErrorImage = (double(img2) - double(img)).^ 2;
mse = sum(sum(squaredErrorImage))/(rows * columns);
PSNR = 10*log10( 256^2 / mse);
message = sprintf('The mean square error is %.2f.\nThe PSNR = %.2f', mse, PSNR);
msgbox(message);
```

Appendix F2

m-Implementation of Modified Fire Fly Algorithm (mFOA)

```
function [Obj,ID]=mFOA(n,X)
clc;
clear;
close all;
%% Problem Definition

model=nColor()%CreateModel();   % Create Model

model.Umax=400;

CostFunction=@(xhat) MyCost(xhat,model);          % Cost Function

VarSize=[model.K model.H];   % Size of Decision Variables Matrix

nVar=prod(VarSize);    % Number of Decision Variables

VarMin=0;         % Lower Bound of Variables
VarMax=1;         % Upper Bound of Variables
%% Firefly Algorithm Parameters
MaxIt=1000;         % Maximum Number of Iterations
nPop=25;           % Number of Fireflies (Swarm Size)
gamma=1;            % Light Absorption Coefficient
beta0=2;           % Attraction Coefficient Base Value
alpha=0.3;         % Mutation Coefficient
alpha_damp=0.98;    % Mutation Coefficient Damping Ratio
delta=0.07*(VarMax-VarMin);     % Uniform Mutation Range
m=2;
if isscalar(VarMin) && isscalar(VarMax)
   dmax = (VarMax-VarMin)*sqrt(nVar);
else
   dmax = norm(VarMax-VarMin);
end
%% Initialization
% Empty Firefly Structure
firefly.Position=[];
firefly.Cost=[];
firefly.Sol=[];
% Initialize Population Array
pop=repmat(firefly,nPop,1);
% Initialize Best Solution Ever Found
BestSol.Cost=inf;
% Create Initial Fireflies
for i=1:nPop
  pop(i).Position=unifrnd(VarMin,VarMax,VarSize);
```

```
  [pop(i).Cost, pop(i).Sol]=CostFunction(pop(i).Position);
  if pop(i).Cost<=BestSol.Cost
     BestSol=pop(i);
  end
end
% Array to Hold Best Cost Values
BestCost=zeros(MaxIt,1);
%% Firefly Algorithm Main Loop
for it=1:MaxIt
   newpop=repmat(firefly,nPop,1);
   for i=1:nPop
      newpop(i).Cost = inf;
      for j=1:nPop
         if pop(j).Cost < pop(i).Cost || i==j
            rij=norm(pop(i).Position-pop(j).Position)/dmax;
            beta=beta0*exp(-gamma*rij^m);
            e=delta*unifrnd(-1,+1,VarSize);
            %e=delta*randn(VarSize);

            newsol.Position = pop(i).Position ...
                        + beta*rand(VarSize).*(pop(j).Position-pop(i).Position) ...
                        + alpha*e;

            newsol.Position=max(newsol.Position,VarMin);
            newsol.Position=min(newsol.Position,VarMax);

            [newsol.Cost, newsol.Sol]=CostFunction(newsol.Position);

            if newsol.Cost <= newpop(i).Cost
               newpop(i) = newsol;
               if newpop(i).Cost<=BestSol.Cost
                  BestSol=newpop(i);
               end
            end

         end
      end

   end

   % Merge
   pop=[pop
       newpop];  %#ok

   % Sort
   [~, SortOrder]=sort([pop.Cost]);
   pop=pop(SortOrder);

   % Truncate
   pop=pop(1:nPop);
```

```
    % Store Best Cost Ever Found
    BestCost(it)=BestSol.Cost;

    % Show Iteration Information
    if BestSol.Sol.IsFeasible
        Flag=' (Feasible)';
    else
        Flag='';
    end
    disp(['Iteration ' num2str(it) ': Best Cost = ' num2str(BestCost(it)) Flag]);
   % Damp Mutation Coefficient
    alpha = alpha*alpha_damp;
    % Plot Solution
    figure(1);
    PlotSolution(BestSol.Sol,model);
    pause(0.01);
end
%% Results
figure;
plot(BestCost,'LineWidth',2);
xlabel('Iteration');
ylabel('Best Cost');
grid on;
```