IMPLEMENTATION OF NEW THREE STEP SEARCH (NTSS) ALGORITHM FOR MOTION ESTIMATION USING MATLAB

NOR HASHELA BINTI JAMADIN

UNIVERSITI TEKNIKAL MALAYSIA MELAKA

IMPLEMENTATION OF NEW THREE STEP SEARCH (NTSS) ALGORITHM FOR MOTION ESTIMATION USING MATLAB

NOR HASHELA BINTI JAMADIN

This report is submitted in partial fulfillment of the requirements for the award of Bachelor of Electronic Engineering (Telecommunication Electronics)

With Honours

Faculty of Electronic and Computer Engineering

Universiti Teknikal Malaysia Melaka

April 2009

UNIVERSTI TEKNIKAL MALAYSIA MELAKA
FAKULTI KEJURUTERAAN ELEKTRONIK DAN KEJURUTERAAN KOMPUTER

BORANG PENGESAHAN STATUS LAPORAN
PROJEK SARJANA MUDA II

Tajuk Projek : IMPLEMENTATION OF NEW THREE STEP SEARCH ALGORITHM FOR MOTION ESTIMATION USING MATLAB

Sesi Pengajian : 2005/2009

Saya NOR HASHELA BINTI JAMADIN
(HURUF BESAR)

mengaku membenarkan Laporan Projek Sarjana Muda ini disimpan di Perpustakaan dengan syarat-syarat kegunaan seperti berikut:

1. Laporan adalah hakmilik Universiti Teknikal Malaysia Melaka.
2. Perpustakaan dibenarkan membuat salinan untuk tujuan pengajian sahaja.
3. Perpustakaan dibenarkan membuat salinan laporan ini sebagai bahan pertukaran antara institusi pengajian tinggi.
4. Sila tandakan (√) :

☐	**SULIT***	(Mengandungi maklumat yang berdarjah keselamatan atau kepentingan Malaysia seperti yang termaktub di dalam AKTA RAHSIA RASMI 1972)
☐	**TERHAD***	(Mengandungi maklumat terhad yang telah ditentukan oleh organisasi/badan di mana penyelidikan dijalankan)
☑	**TIDAK TERHAD**	

Disahkan oleh:

____
(TANDATANGAN PENULIS)

____
(COP DAN TANDATANGAN PENYELIA)

Alamat Tetap: 267-A MUKIM 11, SG. BATU,
TELUK KUMBAR,11920
BAYAN LEPAS, PENANG.

Tarikh: 30 APRIL 2009

Tarikh: 30 APRIL 2009

"I hereby declare that this report is the result of my own work except for quotes as cited in the references."

Signature :………………………………….

Author :NOR HASHELA BINTI JAMADIN

Date :30 APRIL 2009

"I hereby declare that I have read this report and in my opinion this report is sufficient in terms of the scope and quality for the award of Bachelor of Electronic Engineering (Telecommunication Electronics) With Honours."

Signature	:……………………………………
Supervisor's Name	:REDZUAN BIN ABDUL MANAP
Date	:30 APRIL 2009

DEDICATION

To My Beloved Parents

ACKNOWLEDGEMENT

In preparing this thesis, I would like to thankful to Allah the Almighty Who guide and bless me towards the right in the journey of my life. In particular, I would like to express my sincere appreciation and gratitude to my honorable supervisor, En. Redzuan Bin Abdul Manap, for his guidance, suggestion, encouragement, critics and patience in guiding me towards the completion of this final year project.

I also would like to extend my sincere appreciation to my dearest friends especially Hajar, Irwan Zilah, Elya and Ada in sharing similar research interest and all BENT students who were always there to give support and encouragement to me.

Last but not least, I would like to thank my loving parents, En. Jamadin Bin Ali and Pn. Fadzilah Binti Othman and also my siblings. Without their love, support and patience, I would not be able to go through the hard times and they given me strength to finish what I have started.

May Allah bless all of you.

Thank You.

ABSTRACT

The three step search (TSS) algorithm has been widely used as the motion estimation technique in some low bit-rate video compression applications, owing to its simplicity and effectiveness. However, TSS uses a uniformly allocated checking point pattern in its first step, which becomes inefficient for the estimation of small motions. A new three-step search (NTSS) algorithm is implemented in this project. The features of NTSS are that employs a center-biased checking point pattern in the first step, which is derived by making the search adaptive to the motion vector distribution, and a halfway-stop technique to reduce the computation cost. Besides that, the performance of NTSS will compare with the others Block Matching Algorithm (BMA) when the simulation process is done.

ABSTRAK

Algoritma tiga langkah carian (TSS) biasanya digunakan sebagai teknik penganggaran gerakan dalam aplikasi pemampatan video berkadar bit rendah kerana kemudahannya dan keefektifannya. Walaubagaimanapun, TSS menggunakan paten titik pencarian yang seragam dalam langkah pertama, di mana tidak efektif dalam penganggaran pergerakan yang kecil. Algoritma tiga langkah carian baru (NTSS) telah dicadangkan di dalam projek ini. Ciri-ciri oleh algoritma NTSS ini ialah ia menggunakan kecenderungan corak titik tengah pada langkah pertama, di mana ia diterbitkan dengan membuat pengubahsuaian dalam pengagihan vector pergerakan, dan teknik perhentian setengah jalan untuk mengurangkan kerumitan pengkomputeran. Selain itu, pencapaian oleh algoritma NTSS ini akan dibandingkan dengan algoritma yang lain selepas simulasi dijalankan.

TABLE OF CONTENTS

CHAPTER	TITLE	PAGE
	PROJECT TITLE	i
	VERIFYING FORM	ii
	DECLARATION	iii
	SUPERVISOR APPROVAL	iv
	DEDICATION	v
	ACKNOWLEDGEMENT	vi
	ABSTRACT	vii
	ABSTRAK	viii
	TABLE OF CONTENTS	ix
	LIST OF TABLE	xii
	LIST OF FIGURE	xiii
	LIST OF ABBREVIATIONS	xv
	LIST OF APPENDICES	xvii

I	**INTRODUCTION**	**1**
	1.1 Problem Statement	2
	1.2 Objective	2
	1.3 Scope of Project	3
	1.4 Report Structure	3
II	**LITERATURE REVIEW**	**5**
	2.1 Video Compression	5
	2.2 Coding Techniques	8
	2.3 Motion Estimation & Motion Compensation	10
	2.4 Block Matching Algorithm	14
	2.4.1 Full Search Algorithm(FS)	17
	2.4.2 Cross Search Algorithm (CS)	18
	2.4.3 Four Step Search Algorithm (4SS)	22
	2.4.4 Diamond Search Algorithm (DS)	26
	2.4.5 Cross Diamond Search Algorithm (CDS)	30
III	**METHODOLOGY**	**33**
	3.1 Methodology Overview	33

IV **A NEW THREE STEP SEARCH ALGORITHM** **37**

4.1 Introduction 37

4.2 NTSS Algorithm 38

V **RESULTS AND DISCUSSION** **43**

5.1 First Stage of Simulation 44

5.2 Second Stage of Simulation 53

5.3 Summary 62

VI **CONCLUSION** **63**

6.1 Conclusion 63

6.2 Suggestion for Future Work 66

REFERENCES **67**

APPENDIX **69**

LIST OF TABLES

NO	TITLE	PAGE
Table 2.1	Uncompressed bit rates	6
Table 2.2	A Classification of Coding/Compression Techniques	9
Table 5.1	Number of Search Points for Block Matching Algorithm	45
Table 5.2	PSNR value for Block Matching Algorithm	46
Table 5.3	Speed up ratio compared to Full Search Algorithm	46
Table 5.4	Number of Search Points for Block Matching Algorithm	54
Table 5.5	PSNR value for Block Matching Algorithm	54
Table 5.6	Speed up ratio compared to Full Search Algorithm	55

LIST OF FIGURE

NO	TITLE	PAGE
Figure 2.1	A generic M-PEG1 and M-PEG2 encoder structure	8
Figure 2.2	Reference frame and Predicted frame	11
Figure 2.3	Motion Compensation	11
Figure 2.4	Generating motion compensation error surfaces using block matching (relative to the two reference frames)	12
Figure 2.5	The current and previous frames in a search window	15
Figure 2.6	An example of CSA search for w=8pels/frame	20
Figure 2.7	Flowchart of cross search algorithm	21
Figure 2.8	Step of 4SS algorithm	24
Figure 2.9	Flowchart of Four Step Search Algorithm	25
Figure 2.10	Diamond Search	28
Figure 2.11	Flowchart of Diamond Search Algorithm	29

Figure 2.12	Search patterns used in CDS algorithm	30
Figure 2.13	Example of CDS	32
Figure 2.14	Flowchart of Cross Diamond Search algorithm	32
Figure 3.1	Flowchart of Methodology	34
Figure 4.1	NTSS Search Point	41
Figure 4.2	The block diagram of the NTSS algorithm	42
Figure 5.1	Number of search points and PSNR value	47
Figure 5.2	Original frame and predicted frame of Tennis Video	48
Figure 5.3	Number of search points and PSNR value	49
Figure 5.4	Original frame and predicted frame of News Video	50
Figure 5.5	Number of search points and PSNR value	51
Figure 5.6	Original frame and predicted frame of Claire Video	52
Figure 5.7	Number of search points and PSNR value	56
Figure 5.8	Original frame and predicted frame of Tennis Video	57
Figure 5.9	Number of search points and PSNR value	58
Figure 5.10	Original frame and predicted frame of News Video	59
Figure 5.11	Number of search points and PSNR value	60
Figure 5.12	Original frame and predicted frame of Claire Video	61

LIST OF ABBREVIATIONS

ADPCM	-	Adaptive Differential Pulse Code Modulation
BDM	-	Block Distortion Measure
BMA	-	Block Matching Algorithm
CDS	-	Cross Diamond Search
CIF	-	Common Intermediate Format
CS	-	Cross Search
CSP	-	Cross-shaped search pattern
DCT	-	Discrete Cosine Transform
DPCM	-	Differential Pulse Code Modulation
DS	-	Diamond Search
FFT	-	Fast Fourier Transform
FS	-	Full Search
FSS	-	Four Step Search
JPEG	-	Joint Photographic Experts Group
LDSP	-	Large diamond search pattern
MATLAB	-	Matrix Laboratory
MPEG	-	Moving Pictures Experts Group

MSE	-	Mean Square Error
NTSS	-	New Three Step Search
PCM	-	Pulse Code Modulation
PSNR	-	Peak Signal-to-Noise ratio
QCIF	-	Quarter Common Intermediate Format
SAE	-	Sum of absolute error
SDSP	-	Small diamond search pattern
SSE	-	Sum of squared error
TSS	-	Three Step Search
VLC	-	Variable Length Decoder

LIST OF APPENDICES

NO	TITLE	PAGE
A	Frames sequence of Tennis Video	69

CHAPTER 1

INTRODUCTION

To achieve high compression ratio in video coding, a technique known as Block Matching Motion Estimation has been widely adopted in various coding standards. This technique is implemented conventionally by exhaustively testing all the candidate blocks within the search window. This type of implementation, called Full Search (FS) Algorithm gives the optimum solution. However, substantial amount of computational workload is required in this algorithm. To overcome this drawback, many fast Block Matching Algorithms (BMA's) have been proposed and developed. Different search patterns and strategies are exploited in these algorithms in order to find the optimum motion vector with minimal number of required search points.

One of these fast BMA's, which is proposed to be implemented in this project, is called New Three Step Search (NTSS) Algorithm. The student is required to implement the algorithm in MATLAB and then compare its performance to FS algorithm as well as to other fast BMA's in terms of the peak signal-to-noise ratio (PSNR) produced, number of search points required and computational complexity.

1.1 PROBLEM STATEMENT

Substantial amount of computational workload is required during the execution of FS algorithm; however this drawback can be overcome by many types of fast BMA which have been proposed and developed. Different search patterns and strategies are exploited in these fast BMAs in order to find the optimum motion vector with minimal number of required search points. However, there is a need to determine which of these fast BMAs perform the best as well as to identify the most suitable algorithm for different types of video sequences.

1.2 OBJECTIVE

The objective of this project is to implement the NTSS algorithm in MATLAB and to compare its performance to FS algorithm as well as to other fast Block Matching Algorithms such as Diamond Search (DS), Four Step Search (4SS), Cross Search (CS) and Cross Diamond Search (CDS).

1.3 SCOPE OF PROJECT

This project will be focusing on three main areas of study, which are literature review on video coding, BMAs and NTSS, the development and implementation of NTSS algorithm using MATLAB platform and also performance analysis of NTSS to FS algorithm and NTSS to other BMAs.

1.4 THESIS STRUCTURE

The following thesis contents are divided into five chapters.

The first chapter of this thesis is a brief introduction on the project being carried out such as objective of the study, problem statements, the scope and structure of the project.

In chapter II, the BMA will be discussed in details such as on video compression, coding technique, motion estimation and the general ideas of other fast BMAs that have been implemented.

Subsequently, the methodology of the project will be discussed in chapter III of this thesis including the literature reviews and the techniques that was carried out.

In chapter IV, the NTSS Algorithm will be discussed in details such as on concept of the algorithm and the motion estimation for NTSS.

Chapter V is the discussion on simulation results obtained by using MATLAB and results comparison to the other algorithms.

Finally, Chapter VI will conclude this thesis with a critical review and recommendation for future works.

CHAPTER 2

LITERATURE REVIEW

2.1 Video Compression

Representing video material in a digital form requires a large number of bits. The volume of data generated by digitizing a video signal is too large for most storage and transmission systems. This means that compression is essential for most video applications.

Table 2.1 shows the uncompressed bit rates of several popular video formats. From this table it can be seen that even QCIF at 15 frames per second (i.e. relatively low-quality video, suitable for video telephony) requires 4.6 Mbps for transmission or storage.

There is a clear gap between the high bit rates demanded by uncompressed video and the available capacity of current networks and storage media[1].

Table 2.1 Uncompressed bit rates

Format	Luminance resolution	Chrominance resolution	Frames per second	Bits per seconds (uncompressed)
ITU-R 601	858 x 525	429 x 525	30	216 Mbps
CIF	352 x 288	176 x 144	30	36.5 Mbps
QCIF	176 x 144	88 x 72	15	4.6 Mbps

Uncompress video generally exceeds network bandwidth capacity and requires too much space in disk storage. Consequently, it is not practical to transmit video without using compression. Video compression is design to minimize the average number of bits used to represent the video sequence in digital form by maintaining the quality of sufficient video even with little apparently loss in quality.

Compressing video is essentially the process of throwing away data for thing that we can’t perceive. Compress video is called to make the files become smaller and easier to store. Video can be compressed without impacting the quality because it will affect the part that the human’s can’t perceive. The main idea for the video compression is to get the right balance of quality and size file.

Video compression can be performed in two different methods which is lossy and lossless. For lossless method, each pixel is kept unchanged, resulting in identical and bit for bit image after decompressed. While for lossy method, the resulting video compression is different from original but it is still good enough to use. Lossy method is commonly used in most cases.