"I confess that I have read this report and for my opinion I think this report is sufficed in partial fulfillment of requirements for the Bachelor of Electronic Engineering with Honors (Computer Engineering)."

Signature :
Supervised by : MR. NURULFAJAR BIN ABD MANAP
Date : 5th May, 2006

© Universiti Teknikal Malaysia Melaka

DEVELOPMENT OF IMAGE COMPRESSION ALGORITHM

MUHAMAD KHALEEL ALYASINI BIN ABDULLAH

This Report Is Submitted In Partial Fulfillment of Requirements for the Bachelor of Electronic Engineering with Honors (Computer Engineering)

Fakulti Kejuruteraan Elektronik Dan Kejuruteraan Komputer
Kolej Universiti Teknikal Kebangsaan Malaysia

April 2006

© Universiti Teknikal Malaysia Melaka

"I confess this report is all my own work except the certain passage that I have clarified each of their sources."

Signature : Khaleel

Prepared by : MUHAMAD KHALEEL ALYASINI BIN ABDULLAH

Date : 5th May 2006

© Universiti Teknikal Malaysia Melaka

ACKNOWLEDGEMENT

Praise to the Eternal One, Allah S.W.T. for blessing and guiding me through this entire project and for showing me the path so that I can complete this project. Million thanks to my parents which have been giving non-stop of support in whatever form they can provide. Not to forget Mr. NurulFajar Bin Abd Manap for being tolerant with my problems and attitude while guiding me the way to complete this project and thank you again for providing me materials, ideas and suggestion. To my friends who have always been there whenever I am troubled and helped me in my darkest day. Last but not least, to anyone who contributed their help and time in this project whether they are lecturers, persons and everyone direct or indirectly involved in this project.

© Universiti Teknikal Malaysia Melaka

ABSTRACT

This project is about developing an image compression algorithm and implements it in C++ language. Image compression is a method where to save bandwidth or storage which processes the pixels' values by certain rules and methods. C++ was chosen as the main medium for the implementation. This is due to its flexibility in terms of debugging and cost saving comparing to other mediums. The chosen method is the Discrete Cosine Transform method. This technique is chosen because of its practicability and can be said one of the highest rated technique. Upon of its completion, the images will be discussed in terms of quality which depending on the quantization tables used. As a result, it can be said that the image is successfully compressed and can be used in many applications such as video-on-demand, video conferencing and others.

© Universiti Teknikal Malaysia Melaka

ABSTRAK

Projek ini adalah mengenai pembangunan algoritma pemadatan imej yang dihasilkan dengan menggunakan bahasa pengaturcaraan C++. Teknik pemadatan imej ialah suatu teknik yang sering digunakan bagi menjimatkan ruang simpanan dan 'bandwidth'. Proses yang melibatkan nilai-nilai piksel imej yang dilakukan dan menghasilkan keputusan. Bahasa pengaturcaraan C++ dipilih kerana kelebihannya dari segi kos dan pembetulan kepada pengaturcaraan atau pun teknik jika disbanding dengan medium-medium lain seperti perkakas khas. Teknik Discrete Cosine Transform dipilih kerana teknik ini dapat menghasilkan keputusan yang agak bagus dan mudah untuk dilaksanakan berbanding dengan teknik-teknik lain. Akhirnya, imej-imej ini dapat dipadatkan dan akan dibandingkan dari segi kualiti serta teknik ini boleh digunakan dalam aplikasi-aplikasi dunia moden ini seperti 'video-on-demand', 'video conferencing' dan sebagainya.

© Universiti Teknikal Malaysia Melaka

CONTENTS

© Universiti Teknikal Malaysia Melaka

© Universiti Teknikal Malaysia Melaka

LIST OF FIGURES

© Universiti Teknikal Malaysia Melaka

© Universiti Teknikal Malaysia Melaka

LIST OF ABBREVIATION

GUI	–	Graphical User Interface
DCT	–	Discrete Cosine Transform
DFT	–	Discrete Fourier Transform
FFT	–	Fast Fourier Transform
IDCT	–	Inverse Discrete Cosine Transform
IDFT	–	Inverse Discrete Fourier Transform
DWT	–	Discrete Wavelet Transform
CWT	–	Continuous Wavelet Transform
JPEG	–	Joint Photographic Experts Group
MB	–	Mega Bytes
KB	–	Kilo Bytes
Mb	–	Mega Bits
Kb	–	Kilo Bits

© Universiti Teknikal Malaysia Melaka

CHAPTER I

PROJECT OVERVIEW

1.1 Introduction

Nowadays, digital technology enables users to achieve various accomplishments, even reaching a world beyond imaginations. One of these digital technologies is digital imaging.

Digital imaging is being rapidly developed in order to help people to complete their work easier. In other words, digital imaging helps a lot in a person's life. Its applications are numerous such as online gaming, 3D modeling, video conferencing, video on demand and others. Even the Window-based operating systems such as Microsoft Windows itself display disk's file directory graphically.

Basically, many programs provide graphical user interface (GUI) which makes it easier to be used and to interpret the displayed results. The graphics are being used in such applications tend to be big in size which depends on its pixels and resolution.

Therefore, there will be a need for high storage and high bandwidth requirement for each image. Thus, this is why image compression is needed and important in solving the problem.

© Universiti Teknikal Malaysia Melaka

1.2 Problem Statement

One of the important aspects of image storage is its efficient compression. To make this fact clear let's see an example.

An image, 1024 pixel x 1024 pixel x 24 bit, without compression, would require 3 MB of storage and 7 minutes for transmission, utilizing a high speed, 64 Kbit/s, ISDN line. If the image is compressed at a 10:1 compression ratio, the storage requirement is reduced to 300 KB and the transmission time drops to under 6 seconds. Seven 1 MB images can be compressed and transferred to a floppy disk in less time than it takes to send one of the original files, uncompressed, over any network.

In a distributed environment large image files remain a major bottleneck within systems. Compression is an important component of the solutions available for creating file sizes of manageable and transmittable dimensions. Increasing the bandwidth is another method, but the cost sometimes makes this a less attractive solution.

Platform portability and performance are important in the selection of the compression and decompression technique to be employed. Compression solutions today are more portable due to the change from proprietary high end solutions to accepted and implemented international standards.

1.3 Objectives

The main objective of this project is to compress an image's size while maintaining its quality. Currently, problems associated with an image are:

- High storage requirement

  To maintain its original form and data, usually an image tends to be big in size.
- High bandwidth channel requirement

© Universiti Teknikal Malaysia Melaka

If an image is big size, it requires a high bandwidth channel for its data to be transmitted from a place to another, e.g. Internet.

Therefore, this project will be implementing Discrete Cosine Transformation algorithm by using C++ as its medium. Thus, the image could be compressed with its quality sustained. Other objectives are:

- To develop an image compression algorithm by using C++.
- To understand and learn the concepts of image compression.

1.4 Scope of Work

Scopes of work are divided into three stages:

- Stage One: Literature Review
  This is the most basic level in completing the project. Papers, journals and books are being studied in order to help to decide on which algorithm should be used.

- Stage Two: Algorithm Programming
  Stage two is where the real work being done. By applying the formulas that were studied and researched in the earlier stage, a source code is being programmed. This is most crucial stage of all.

- Stage Three: Implementation
  The last stage is where we can find out the results whether the program is correct and successful or not. Plus, maybe in this stage, probably new features could be added to the code.

© Universiti Teknikal Malaysia Melaka

1.5 Overall Project Flow

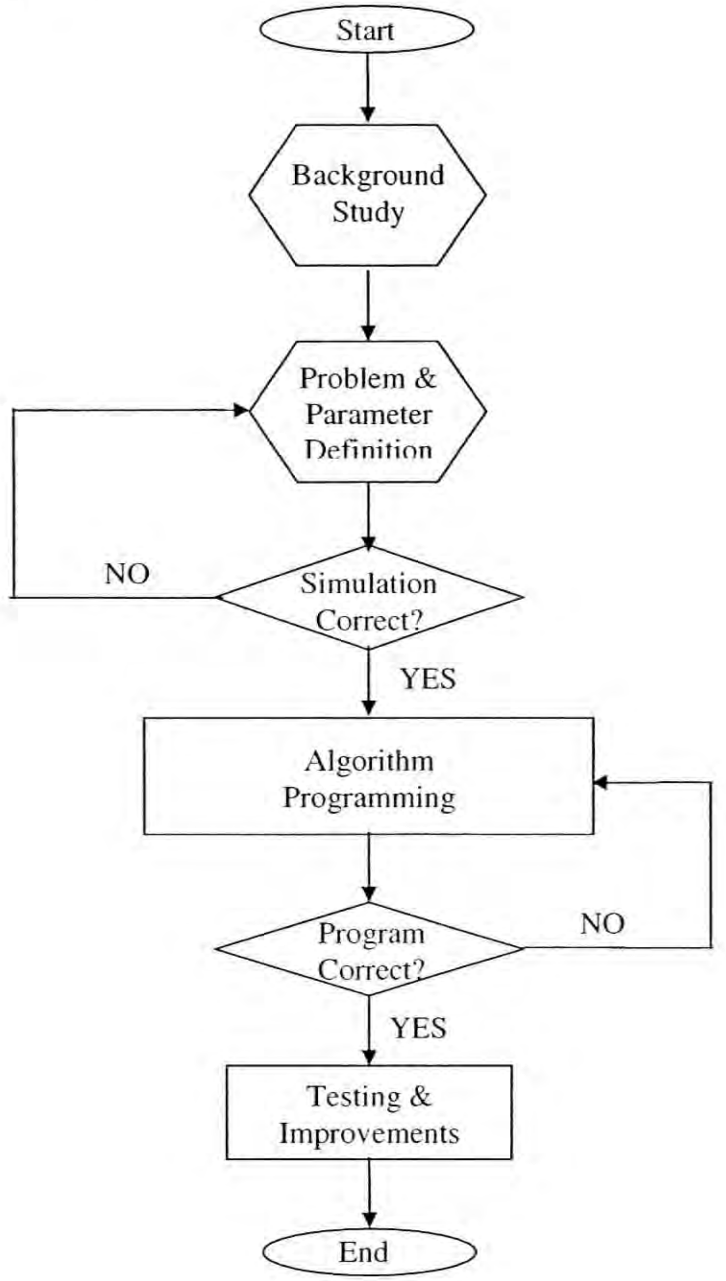

Figure 1.1: Overall project flow

© Universiti Teknikal Malaysia Melaka

Figure 1.1 shows the overall project flows. The project started with background study or literature review. In order to know the in-depth knowledge on image compression, this part took some time to be completed and followed by simulations process. This process is to define and recognize the parameters that are used in the algorithm. Then, the project continues with algorithm programming which this was the crucial part of the process. Lastly, the program is tested and modified in some parts in order to make it better.

1.6 Thesis Outline

Chapter 1 covers the project overview. The objectives, the purpose are all stated in this chapter. Plus, a flowchart which resembles the overall project flow is included.

Chapter 2 will tell about the theories and equations that are being used in order to complete this project. Introduction on image compression and its available techniques are being covered in this chapter.

Chapter 3 is about the project methodology. Project methodology is about what processes are being done to complete the project. It emphasizes the details on the processes that are used.

Chapter 4 is the results and findings chapter. This chapter consists of the results and its findings which most of them are figures. Alongside with some explanation, this chapter covers the output for this project.

Chapter 5 is to discuss and concludes on the overall project. This chapter emphasizes on advantages and disadvantages. Plus, it includes future modifications and suggestions.

© Universiti Teknikal Malaysia Melaka

CHAPTER II

LITERATURE REVIEW

2.1 Introduction

An image is represented as a two-dimensional array of coefficients, each coefficient representing the brightness level in that point. When looking from a higher perspective, it is hard to differentiate between coefficients as more important ones, and lesser important ones [8]. But thinking more intuitively, it is possible [8]. Most natural images have smooth color variations, with the fine details being represented as sharp edges in between the smooth variations. Technically, the smooth variations in color can be termed as low frequency variations and the sharp variations as high frequency variations.

The low frequency components (smooth variations) constitute the base of an image, and the high frequency components (the edges which give the detail) add upon them to refine the image, thereby giving a detailed image [4]. Hence, the smooth variations are demanding more importance than the details.

Current digital technology enables users to reach a world beyond imaginations. There are many of its applications in this world nowadays. One of it is digital imaging. Digital imaging is being rapidly developed in order to ease our daily routines. Its applications are numerous ranging from videos applications such as video on demand, video conferencing to online gaming or 3-Dimensional graphics imaging. Basically, many programs provide graphical user interface (GUI) to its users which makes it easier to use and to interpret the displayed results [1]. These graphics are usually tended to be

© Universiti Teknikal Malaysia Melaka

big in size which depends on its pixels and resolution. Therefore, there will be a need for high storage and high bandwidth requirement for each image. Thus, this is where the role of image compression comes in to help solving the problems.

2.2 Image Compression

Image compression is a technique to compress an image while maintaining its quality. Image compression and decompression operations are used to reduce the data-content size of a digital image. The goal of these operations is to represent an image, with some required quality level, in more compact form.

Mainly, there are two types of image compression method which are widely used in the world these days. They are lossless compression and lossy compression. Lossless image compression preserves the exact data content of the original image. Lossy compression scheme is known as a concept that namely compressing by removing irrelevancy. In other words, the image has some loss of information but it is acceptable as the reception quality is good and can be neglected.

A full image compression step consists of image compression and decompression and can be pictured as below:

© Universiti Teknikal Malaysia Melaka

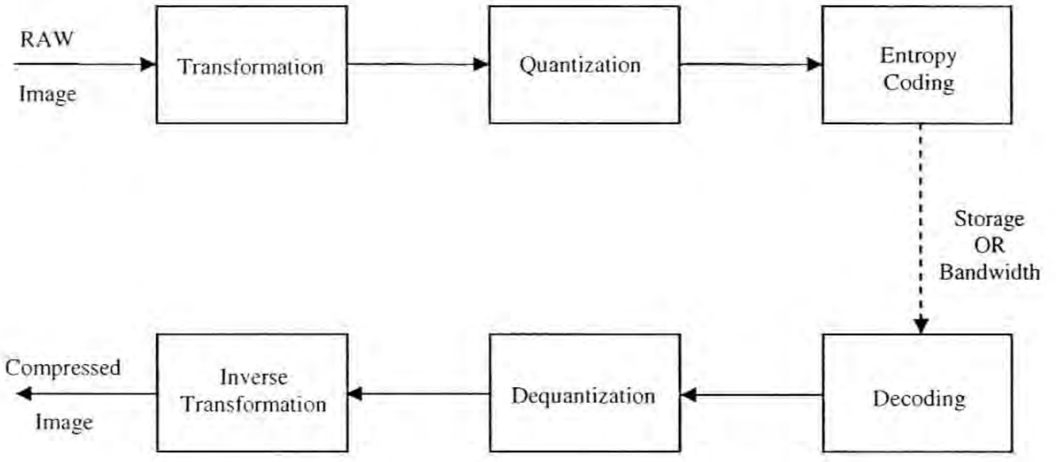

Figure 2.1: Image compression algorithm

Any basic image compression method contains these three steps:

- Image Transformation
- Image Quantization
- Image Entropy Coding

But for some cases, entropy coding is ignored because the two main steps in an image compression process are transformation and quantization. Entropy coding is just a step to measure how much compression is obtained from a quantization matrix.

2.3 Image Transformation

There are a lot of image transformation techniques. A few of them are highlighted on this thesis. All of these techniques served the purpose for converting a signal into elementary frequency components. The techniques are listed below:

© Universiti Teknikal Malaysia Melaka

- Discrete Cosine Transform
- Discrete Fourier Transform
- Wavelet Transform

2.3.1 Discrete Fourier Transform (DFT)

In mathematics, the discrete Fourier transform (DFT), sometimes called the finite Fourier transform, is a Fourier transform widely employed in signal processing and related fields to analyze the frequencies contained in a sampled signal, solve partial differential equations, and to perform other operations such as convolutions. The DFT can be computed efficiently in practice using a fast Fourier transform (FFT) algorithm.

The sequence of N complex numbers $x_0, \ldots, x_{N-1}$ are transformed into the sequence of N complex numbers $X_0, \ldots, X_{N-1}$ by the DFT according to the equation 2.3.1.1:

$$X_k = \sum_{n=0}^{N-1} x_n e^{\frac{-2\pi ikn}{N}} \quad k = 0,\ldots, N-1 \qquad (2.3.1.1)$$

where e is the base of the natural logarithm, i is the imaginary unit ($i2 = -1$), and π is Pi. The transform is sometimes denoted by the symbol F, as in $X = F(x)$.
The inverse discrete Fourier transform (IDFT) is given according to equation 2.3.1.2:

$$x_n = \frac{1}{N} \sum_{k=0}^{N-1} X_k e^{\frac{2\Pi i}{N} kn} n = 0,\ldots, N-1 \qquad (2.3.1.2)$$

Note that the normalization factor multiplying the DFT and IDFT (here *1* and *1/N*) and the signs of the exponents are merely conventions, and differ in some

© Universiti Teknikal Malaysia Melaka

treatments. The only requirements of these conventions are that the DFT and IDFT have opposite-sign exponents and that the product of their normalization factors be 1/N. A normalization of $1/\sqrt{N}$ for both the DFT and IDFT makes the transforms unitary, which has some theoretical advantages, but it is often more practical in numerical computation to perform the scaling all at once as above (and a unit scaling can be convenient in other ways). (The convention of a negative sign in the exponent is often convenient because it means that X_k is the amplitude of a "positive frequency" $2\pi k / N$. Equivalently, the DFT is often thought of as a matched filter: when looking for a frequency of $+1$, one correlates the incoming signal with a frequency of -1.)

2.3.2 Discrete Cosine Transform (DCT)

The discrete cosine transform (DCT) is a Fourier-related transform similar to the discrete Fourier transform (DFT), but using only real numbers [7]. It is equivalent to a DFT of roughly twice the length, operating on real data with even symmetry (since the Fourier transform of a real and even function is real and even), where in some variants the input and/or output data are shifted by half a sample. (There are eight standard variants, of which four are common.) [5]

The most common variant of discrete cosine transform is the type-II DCT, which is often called simply "the DCT"; its inverse, the type-III DCT, is correspondingly often called simply "the inverse DCT" or "the IDCT". Types of DCT:

- DCT-I
- DCT-II
- DCT-III
- DCT-IV

© Universiti Teknikal Malaysia Melaka

2.3.2.1 DCT-I

Referring to the equation 2.3.2.1 below,

$$x_k = \frac{1}{2}(x_0 + (-1)^k x_{N-1}) + \sum_{n=1}^{N-2} x_n \cos\left[\frac{\Pi}{N-1} nk\right] \qquad (2.3.2.1)$$

Some further multiply the x_0 and x_{N-1} terms by $\sqrt{2}$, and correspondingly multiply the X_0 and X_{N-1} terms by $1/\sqrt{2}$. This makes the DCT-I matrix orthogonal (up to a scale factor), but breaks the direct correspondence with a real-even DFT.

The DCT-I is exactly equivalent (up to an overall scale factor of 2), to a DFT of $2N - 2$ real numbers with even symmetry. For example, a DCT-I of N=5 real numbers *abcde* is exactly equivalent to a DFT of eight real numbers *abcdedcb* (even symmetry), divided by two. (In contrast, DCT types II-IV involve a half-sample shift in the equivalent DFT.)

Note, however, that the DCT-I is not defined for N less than 2. (All other DCT types are defined for any positive N.) Thus, the DCT-I corresponds to the boundary conditions: x_n is even around $n=0$ and even around $n=N-1$; similarly for X_k.

© Universiti Teknikal Malaysia Melaka

2.3.2.2 DCT-II

By referring to the equation 2.3.2.2 below,

$$x_k = \sum_{n=0}^{N-1} x_n \cos\left[\frac{\Pi}{N}\left(n + \frac{1}{2}\right)k\right] \qquad (2.3.2.2)$$

The DCT-II is probably the most commonly used form, and is often simply referred to as "the DCT". This transform is exactly equivalent to a DFT of *4N* real inputs of even symmetry where the even-indexed elements are zero. That is, it is half of the DFT of the *4N* inputs y_n, where $y_{2n} = 0$, $y_{2n} + 1 = x_n$ for $0 \le n \le N$, and $y4N - n = y_n$ *for* $0 < n < 2N$.

Some further multiply the X_0 term by $1/\sqrt{2}$. This makes the DCT-II matrix orthogonal, but breaks the direct correspondence with a real-even DFT of half-shifted input. The DCT-II implies the boundary conditions: x_n is even around $n=-1/2$ and even around $n=N-1/2$; X_k is even around $k=0$ and odd around $k=N$.

2.3.2.3 DCT-III

Refer to the equation 2.3.2.3 below,

$$x_k = \frac{1}{2}x_0 + \sum_{n=1}^{N-1} x_n \cos\left[\frac{\Pi}{N} n(k + \frac{1}{2})\right] \qquad (2.3.2.3)$$

Because it is the inverse of DCT-II, this form is sometimes simply referred to as "the inverse DCT" ("IDCT"). Some multiply the x_0 term by √2. This makes the DCT-III matrix orthogonal, but breaks the direct correspondence with a real-even DFT of half-shifted output.

© Universiti Teknikal Malaysia Melaka

The DCT-III implies the boundary conditions: x_n is even around $n=0$ and odd around $n=N$; X_k is even around $k=-1/2$ and odd around $k=N-1/2$.

2.3.2.4 DCT-IV

The DCT-IV equation is as listed in equation 2.3.2.4 below,

$$X_k = \sum_{n=0}^{N-1} x_n \cos\left[\frac{\Pi}{N}\left(n+\frac{1}{2}\right)\left(k+\frac{1}{2}\right)\right] \qquad (2.3.2.4)$$

The DCT-IV matrix is orthogonal. A variant of the DCT-IV, where data from different transforms are overlapped, is called the modified discrete cosine transform (MDCT). The DCT-IV implies the boundary conditions: x_n is even around $n=-1/2$ and odd around $n=N-1/2$; similarly for X_k.

2.3.3 Wavelet Transform

Wavelet transform itself has many types. There are a large number of wavelet transforms each suitable for different applications. The common ones are listed down below:

- Continuous wavelet transform (CWT)
- Discrete wavelet transform (DWT)
- Fast wavelet transform (FWT)
- Wavelet packet decomposition (WPD)
- Stationary wavelet transform (SWT)

© Universiti Teknikal Malaysia Melaka