UNIVERSITI MALAYSIA PERLIS

DCT Image Compression Implemented on Raspberry Pi to Compress Image Captured by CMOS Image Sensor

by

Ibrahim Saad Mohsin
(1832322627)

A dissertation submitted in fulfillment of the requirements for the degree of Master of Sciences (Embedded System Design Engineering)

School of Computer and Communication Engineering
UNIVERSITI MALAYSIA PERLIS

2019

ACKNOWLEDGMENT

First and foremost, alhamdulillah, the creator the cherisher the wisest the lord of the world. I am extremely thankful to almighty Allah for giving me the chance strength and courage to complete this work, and without his willing this thesis would not have been possible.

I would like to express my utmost and deepest gratitude to my advisor, **Dr. Muhammad Imran Ahmad**, for his guidance in academic research and his support in daily life. he has widened my view in research areas, especially in biometrics and image and signal processing. his motivation and support have guided me towards the successful completion of my thesis.

I would like to convey my deepest gratitude and thanks to my family **(my parents, my brothers and sisters)** for their support spiritually and financially in order to finish this project.

I would like also to extend my gratefulness to my entire friends and staff in the **School of Computer and Communication Engineering,** Unimap for their full support to me all these whiles. Last but not least, my heartfelt thanks to all those who I forgot to mention, but who never the less deserve to be thanked

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

DCT	Discrete Cosine transform
JPEG	Joint Photographic Experts Group
MPEG	Moving Pictures Experts Group
IAAS	Infrastructure as a Service
PAAS	Platform as a Service
SAAS	Software as a Service
EAAS	Everything as a Service
CMOS	Complementary Metal-Oxide Semiconductor
PSNR	Peak Signal-to-Noise Ratio
MSE	Mean Square Error
DWT	Discrete Wavelet transforms
GUI	Graphical User Interface

LIST OF SYMBOLS

F(u,v)	Original image
f(x,y)	Reconstructed Image
E(x,y)	Error between Image

Imej Mampatan DCT Menggunakan Raspberry Pi Untuk Memampat Imej Yang Diambil Oleh Penderia CMOS

ABSTRAK

Imej mengandungi sejumlah besar data digital dan perlu untuk mengurangkan jumlah data digital bagi penghantaran dan pemeliharaan dengan menggunakan pemampatan imej. Projek ini memfokuskan pemampatan imej menggunakan Raspberry Pi, untuk mengurangkan saiz besar imej tetapi mengekalkan kualitinya. Pemampatan isyarat imej adalah teknik pemprosesan isyarat yang penting dalam banyak bidang penyelidikan, seperti perubatan, biometrik, telekomunikasi dan automotif. Tujuan pemampatan adalah untuk mengurangkan jumlah data dan pada masa yang sama untuk mengekalkan kualiti imej dan isyarat. Transformasi DCT adalah pemampatan imej di mana imej asal diubah ke domain lain untuk menghasilkan saiz data yang lebih kecil. Transformasi DCT mempunyai kerumitan pengiraan yang rendah dan algoritma pemprosesan yang cepat. Dalam projek ini, transformasi DCT akan dilaksanakan menggunakan Raspberry Pi SBC yang dijalankan menggunakan pemproses berasaskan ARM. Raspberry Pi mempunyai kelebihan pelaksanaan pemprosesan imej kerana pembangunan perisian sedia ada menawarkan ciri-ciri yang sesuai untuk pemprosesan imej seperti OPENCV. Projek ini terdiri daripada beberapa peringkat reka bentuk seperti pra-pemprosesan imej, pembangunan algoritma DCT, pengiraan kadar ralat dan pengukuran PSNR dan MSE. Algoritma ini dibangunkan menggunakan bahasa pengaturcaraan Python dengan pemprosesan tambahan seperti OpenCV dan NumPy. Perkembangan teknologi internet dan multimedia yang berkembang dengan pesat, mengakibatkan jumlah maklumat yang dikendalikan oleh komputer diperlukan. Ini menyebabkan masalah serius dalam penyimpanan dan penghantaran data imej pada masa sebenar. Oleh itu, cara untuk memampatkan data perlu ditimbangkan supaya kapasiti storan yang diperlukan akan menjadi lebih kecil. Dalam kajian ini pengaruh DCT kepada nisbah mampatan dan PSNR (Nisbah Puncak kepada Nisbah Kebisingan) telah diukur. Kemudian nisbah mampatan dan PSNR pada masa nyata memberikan kualiti visual yang sangat baik. Hasil daripada penggunaan algoritma pemampatan DCT pada imej dengan enam tahap lebih laju pemampatan yang 10, 20, 50, 100,170 dan 200. Prestasi terbaik dapat dicapai dengan tahap pemampatan 200. Walaubagaimanapun, untuk menurunkan tahap mutu pemampatan, pengukuran ralat mula menjadi lebih teruk sehingga apa yang dicapai, di mana perbezaan persepsi dari imej asal dapat mudah dicatat.

DCT Image Compression Implemented on Raspberry Pi to Compress Image Captured by CMOS Image Sensor

ABSTRACT

An image contains large amount of digital data and it is necessary to reduce digital data volume for transmission and preservation by using image compression. This project mainly concentrates on image compression using a Raspberry Pi processor, which helps to preserve a large number of images and in retaining its quality. Signal, Image and data compression are an important signal processing tool in many areas of research, such as medicine, biometric, telecommunication, automotive and. The purpose of compression is to reduce the amount of data at the same time maintain the quality of image and signal for the other purpose. DCT transform is a family of image compression where the raw image is transformed to the other domain to produce smaller size of data. DCT transform has low computational complexity and fast processing algorithm. In this project, DCT transform will be implemented using Raspberry Pi SBC development board running on an ARM based processor. The raspberry Pi board has an advantage of image processing implementation due to the existing software development tool offered a rich feature for image processing such as OPENCV. The project consists of several design stages such as image pre-processing, the development of DCT algorithm, error rate computation and measurement by PSNR and MSE. The algorithm is developed using Python programming language with the additional image processing library such as OpenCV and NumPy mathematical library. The development of Internet and multimedia technologies that grow exponentially, resulting in the amount of information managed by computer is necessary. This causes serious problems in storage and transmission image data on real time. Therefore, should be considered a way to compress data so that the storage capacity required will be smaller. In this research wanted to know the influence of DCT to the compression ratio and to the PSNR (Peak Signal to Noise Ratio). Then the compression ratio and PSNR results on real time provide excellent visual quality. The result of applying DCT compression algorithm on images with six compression rate level which are 10, 20, 50, 100,170 and 200. The best performance can be achieved with compression rate level 200. However, on reducing the quality level of compression rate, the error measurements start becoming worse until a point is reached, where the perceptual difference from the original image can be easily noted.

CHAPTER 1 : INTRODUCTION

1.1 Background of Study

Image processing is a type of signal processing in which an image, such as a photograph or a video frame, is used as an input, and a corresponding set of parameters related to the image, in the form of a video frame or a photograph, is given as an output. There are numerous reasons that make image processing a necessity, such as unwanted camera shake, random unstable camera movements, or the compression of a large image so that it can be used as input for certain applications.

In various areas of research, such as telecommunication, the automotive industry, biometrics, medicine and so on, image, signal and data compression are regarded as essential tools for signal processing. The primary objective of compression is to reduce the amount of data required to represent the signal or image while maintaining the quality of the original. A Haar transform refers to a group of wavelets for image compression, via which the raw image is transformed into another domain while creating data with a smaller size. A wavelet transform, unlike a discrete cosine transform (DCT), is characterised by a fast processing algorithm and low computational complexity(Sahitya et al., 2017). This project consists of a number of design stages, which include a DCT algorithm, image pre-processing, measurement of the MSE and calculation of the error rate. The Python programming language is used to develop the algorithm, together with

the NumPy mathematical library and an additional image processing library called OpenCV.

The latest advancements in digital technology have resulted in a transformation in communication media, and visual information has evolved to play a significant role. A few of the applications in which visual information is becoming important include conferencing, medical imaging, high-definition TV, virtual reality, telephony, servers, CD and CD-ROM archiving, and wireless transmission. Since a raw signal or digital image generally contains a large amount of information, a large storage space and high-capacity communication channels are required to save and transmit the information. Despite recent developments in terms of storage and communication channels, the cost of implementation often becomes a limitation on capacity. In general, as the bandwidth requirement increases, the cost of transmission or storage increases accordingly. In order to meet the requirements for storage capacity or transmission channels, it is imperative to use compression techniques that reduce the data rate while retaining the subjective quality of the decoded signal or image.

The two most widely used techniques in inter-frame image coding are transform coding and sub-band coding. Both of these coding techniques have the potential to remove statistical redundancies and to exploit certain traits of the human visual system using frequency-weighted distortion measures. There are several compression algorithms that can be implemented to achieve the compression of images, and algorithms that have

been established as compression standards include JPEG, MPEG, H.261 and H.263 (Azuwam et al., 2017).

1.2 Problem Statement

In the multimedia sector, image compression system is indispensable, since it is important in sending and receiving multimedia data from one device to another efficiently (Kaimal et al., 2013) without compromising the quality of the data. It is observed that the use of large images that are not able to be successfully compressed will lead to traffic in the network, resulting in a need to increase the bandwidth in order to avoid these shortcomings. For the Raspberry Pi to be able to work as a camera, its imaging feature needs to be enabled.

The development of Internet and multimedia technologies has been exponential, resulting in increases in the amount of information that needs to be managed by computers. This causes serious problems in terms of the storage and transmission of image data in real time, and compression is needed before the data are transferred. A method of compressing data to reduce the storage capacity required therefore needs to be developed (Al-Ani, 2017).

In this project, the problem definition is dependent on the continuous form in which most of the images of interest generally occur. Due to the proliferation of digital technology, it has become important to work with discrete forms of the continuous image.

Hence, the compression algorithm involves sampling and interpolation, and the image is then treated as a discrete entity or a data file.

In another problem statement, the definition is linked with the practicality of the proposed system in terms of remote monitoring. It is found that in most remote systems, a wireless sensor device is required that captures pictures and sends them across the master node. However, the frequency response when taking the pictures may not be sufficiently fast. In cases where the frequency of taking pictures taken needs to be high, it is necessary to analyse the size of the images and to use compression, so that the original image quality is maintained and the images can be sent rapidly across the network with less traffic (Wadkar and Patil, 2016).). In such cases, the time required for compression becomes significant.

The use of MATLAB software and development in C/C++ has driven image processing and computer vision (CV) algorithms. Even though MATLAB provides an effective, high-level platform for prototyping and testing algorithms, it still cannot match the performance of a well-designed and optimised C/C++ implementation. Recently, numerous potential solutions have been developed for efficient image processing and CV algorithms in Python. Two Python libraries are primarily used to develop image processing and computer vision algorithms: NumPy/SciPy, and OpenCV with a Python wrapper. In this project, the use of basic computer vision routines from OpenCV and SciPy is discussed.

Image compression is very important in several applications in order to reduce data storage and shorten transmission times. There are several popular image compression algorithms such as DCT, the Haar transform, the Walsh-Hadamard transform and the wavelet transform. Currently, most compression algorithms are implemented using PC-based systems. A compression algorithm has high computational cost, and thus implementation in a low-cost processor is a challenging task. This limitation may be observed at the processor level, as a low-cost processor has low features compared to a high-end alternative. The Raspberry Pi processor allows the implementation of most widely used 2D DCT compression methods to give images in the Joint Photographic Experts Group (JPEG) format using the OpenCV platform. A DCT compression algorithm is used to compress the full-colour still images, which gives good human visual perception after compression. The experimental results give high-quality compressed images, and these are compared using different quality factors. The implementation of image compression on a Raspberry Pi has a real time application in micro air vehicles (MAVs). The proposed method helps MAVs to store a large number of images captured in this way. All of the limitations of low-cost processors are also examined in detail in this study, such as limited memory sizes and the high computational cost of the algorithm itself. The project will deal with biometric images such as faces, palmprints, fingerprints, and irises. Biometric data contains unique information, and thus the compression algorithm must produce information that is adequate for use in the next stage.

1.3 Aims and Objectives

The main objectives of the project are as follows:

i. To Implement a DCT algorithm on a Raspberry Pi to compress biometric image

ii. To measure the performance of an embedded system in terms of performing a compression algorithm for different compression ratios.

1.4 Project Scope

The primary objective of this project is to develop an image compression algorithm using a Raspberry Pi device. Code is developed and an algorithm presented which can transfer an image from one domain to another. The use of DCT can enable images to be converted to the time domain from the spatial domain. The compression system can eliminate the redundancy that is seen in content with lower frequency.

The project consists of several design stages, such as image pre-processing, the development of the DCT algorithm, and error rate computation and measurement using PSNR and MSE. The algorithm is developed using the Python programming language, with additional image processing libraries such as the OpenCV and NumPy mathematical libraries.

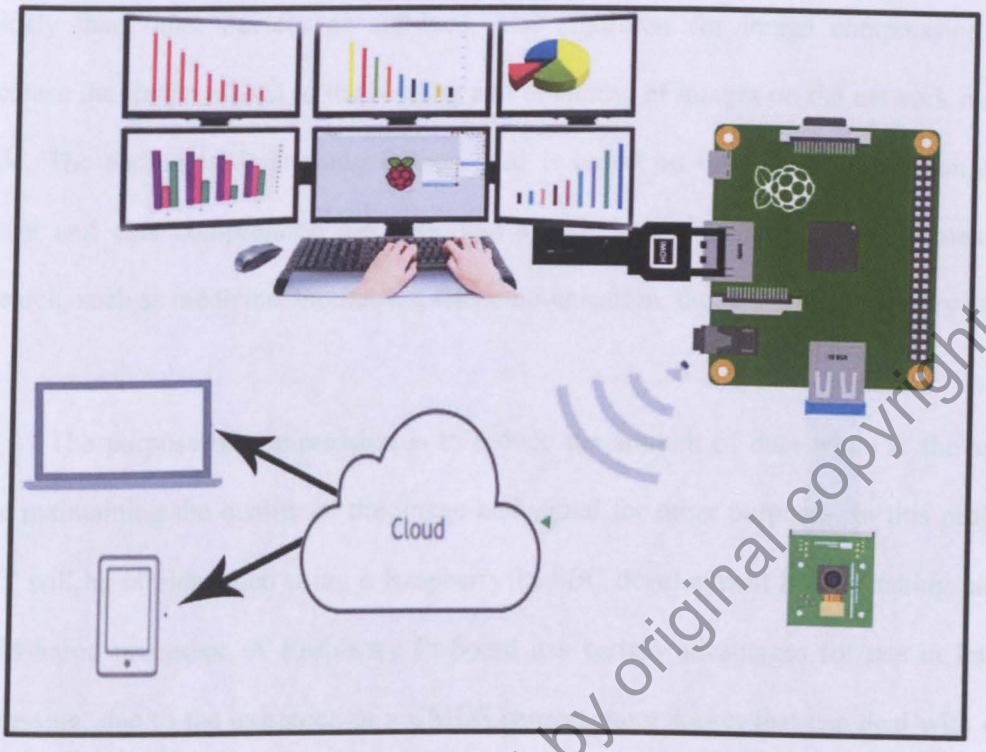

Figure 1.1: Design of the system and devices available with boards and communication methods.

1.5 Summary

Image processing is a type of signal processing in which an image, such as a photograph or a video frame, is used as an input, and a corresponding set of parameters related to the image is given as an output.

The project comprises multiple design phases. Its main objective is to operate the programming code utilising Raspberry Pi, to decrease the image size to that required for transfer from one node to another node. A Raspberry Pi can compress images more

quickly than other devices or software. The algorithm for image compression can decrease the traffic related to the sending and obtaining of images on the network master node. The coding is done using Python, and is based on interpolation and sampling. Image and data compression are important signal processing tools in many areas of research, such as medicine, biometrics, telecommunication, the automotive industry etc.

The purpose of compression is to reduce the amount of data while at the same time maintaining the quality of the image and signal for other purposes. In this project, DCT will be implemented using a Raspberry Pi SBC development board running on an ARM-based processor. A Raspberry Pi board has certain advantages for use in image processing, due to the existence of a CMOS camera input socket that can deal with raw data capture via a camera module. In other SBC boards, the camera module needs to be connected via a USB cable, which produces low-quality images compared to the direct camera input offered by the Raspberry Pi board.

The project consists of several design stages, including image pre-processing, the development of the DCT and inverse DCT algorithms, error rate computation and PSNR measurement. The algorithm is developed using the Python programming language, with the additional image processing libraries OpenCV and Numpy.

CHAPTER 2 : LITERATURE REVIEW

2.1 Introduction

In digital image processing, the image compression algorithm is crucial, especially when there is high capacity or limited resources for the data transmission procedure. The main aim of compression is to decrease the rate of bit size for any image data matrix. This is especially helpful for image processing in microcontrollers such as the Raspberry Pi, where the limited memory available in the microcontroller determines the pattern of images and template sets that need to be stored. However, for calculations of image correlations, this data storage is very helpful due to its processing procedure.

Since it is crucial to reduce the transmission of image bytes from one node to another and to efficiently manage the available memory space that is assigned to the processing of an image, image compression in most cases leads to the permanent removal of part of the original image. Thus, this literature review presents a discussion of the DCT algorithm that is employed in image processing, and describes various works pertaining to microcontroller image processing, with a focus on the common procedures used in image compression to retain the quality of the original image.

2.2 Image Processing on Raspberry Pi

A digital image often suffers from unwanted camera vibrations or unstable camera movements, and image optimisation algorithms are therefore required to remove these unwanted camera vibrations. These image processing concepts are implemented on a Raspberry Pi via the MAV application. A Raspberry Pi is a platform which uses a single credit-card-sized computer, and was developed in the UK.

The Raspberry Pi is based on the Broadcom BCM2835 system on a chip (SOC), which includes an ARM1176JZF-S Core (ARM V6K)700 MHz CPU processor, Broadcom Video Core IV GPU with 17 pins, 3.5 W of power, and 512 MB of RAM memory. The Raspberry Pi system has secure SD card reader (models A and B) or micro SD card reader (models A+ and B+) sockets for boot media and persistent storage. The system provides a Debian Linux operating system Raspbian image for download, and Python is used as the main programming language.

A MAV is a remote-controlled, unmanned aircraft vehicle (UAV) that is significantly smaller than typical UAVs, which have size restrictions. A UAV is an aircraft without a human pilot. Its flight is controlled either by autonomous on-board computers or via remote control by a pilot on the ground or in another vehicle. By mounting a Raspberry Pi camera module on a MAV, the efficiency of this air vehicle can be increased, and new fields of application become available. This approach is needed in military operations in which targets have to be identified, which is often done by a human

on the ground in order to reduce the probability of mistakes. However, a Raspberry Pi camera module is also helpful if a MAV is autonomously flown through an arch (Shilpashree et al., 2015).

2.3 Image Compression Scheme

In this project, the image compression is based on a structured approach using a supporting DCT algorithm, which is applied via a Raspberry Pi-based image compression system. This provides a basic understanding of the concept to obtain a high-complexity and low-memory scheme that matches the requirements of the hardware implementation. The image compression process has several keys, including a transformation stage algorithm, and various algorithms have been developed to make it easier and more straightforward (Gupta et al., 2016).

A typical image processing method is needed to convert into pixels of matrix such as 8×8 pixel, where the encoding for each block is different. In a majority of image compression protocols, the most common 2D 8 point may require intensive computational transformation during compression. Various methods to perform this image transformation can be used, including the DCT method.

2.4 DCT Algorithm

The DCT is a mathematical transformation that takes a signal and transforms it from the spatial domain into the frequency domain. Many digital image and video compression schemes use a block-based DCT, since this algorithm minimises the amount of data needed to recreate a digitised image. In particular, JPEG and MPEG use the DCT to concentrate image information by removing spatial data redundancies in 2D images(Khan et al., 2018). DCT works by separating images into different frequency segments. In a step called quantification, where part of the compression actually occurs, the less important frequencies are ignored. Following this, the most important remaining frequencies are used to retrieve the image in the decompression process. As a result, the reconstructed images may contain some distortion (El_Rahman, 2018).

2.5 DCT Transform in Data Compression

Data compression can be represented in many different forms. In this study, we will focus on DCTs. The coefficients of discrete image transforms can be employed to characterise image data, and those coefficients that make only a small contribution to the information contents can be ignored. Typically, the image splitting is done based on blocks (sub-images) of 8×8 or 16×16 pixels, after which the transformation of each block is done separately. However, this does not consider any correlations that are present between blocks, and produces 'blocking artefacts' that are not desirable if there is a need for a smooth image. However, instead of only the sub-images, the DCT is applied to all