CHAPTER 1
INTRODUCTION

1.1 Background

Advances in information technology, particularly in the field of telecommunications, are currently accelerating at a breakneck pace. One of the advances is digital video viewing. Video is like a basic need for every individual in today's modern era because video technology plays a role in many daily activities, such as finding entertainment, getting information, watching documentaries, or simply finding out the latest news at this time. Anyone can do videography easily because of today's sophisticated technology, for example people can use a smartphone easily to generate high-quality video results but generate high resolution too.

In general, video files have a specific size which is resolution expressed in length x width. The higher the video resolution, the larger the image region can be accessed, but the more significant the video's file size. Aside from resolution, the bandwidth factor is crucial because it specifies how much data is required to play a video file per second. The better the video quality, so higher the bandwidth is. The high bandwidth combined with the high resolution results in a huge video file [1]. These factors contribute to the need for digital data storage media. On the other hand, large amounts of data are extremely difficult to compute and store. Consequently, we need a large storage device or one that reduces the amount of data sent. As a result, video compression techniques are needed in digital video matters in order to save data storage space.

Compression is a technique for reducing the scale of a digital signal's representation. The terms source coding, data compression, bandwidth compression, and signal compression are all used in the literature to refer to the compression process [2]. Minimizing the bit rate of a signal's digital representation is a common compression issue in cases where the signal is represented as an audio, video, or picture signal. A wide range of applications benefits when compressed image, video, and audio signals are usable. Those applications would be impossible to implement without compression [2]. This compression technique aims to compress data so that the bandwidth needed for data transmission in the network is reduced. Compared to data that has not gone through the compression stage, this results in smaller storage space, making it more effective in storing data and shorter data exchange times.

Compressive Sensing (CS) is a modern technique for data compression and data sampling that uses the signal's sparsity as a CS prerequisite to sample the data [3]. In CS, the signal must be sparse to make compression simpler, as it will be difficult to compress if it is not infrequent. This compressed data acquisition protocol (CS) is based on the idea of discarding signal components that provide little information for reconstruction. It focuses on obtaining signals from random measurements, and it reduces the measurements by using the signal's compressibility and sparsity in a specific transform domain [4].

Study on [5], two novel Adaptive Block-based Compressive Sensing (ABCS) algorithms which both use a deterministic 2 Dimension-Discrete Cosine Transform (2D-DCT) dictionary are used to apply CS to greyscale images or intra-coded video frames. The first computes the Block Boundary Variation (BBV) around each image block using a small number of compressive measurements. In contrast, the second estimates the total number of transform coefficients to capture from each block using a small number of DCT domain measurements. Furthermore, the study revealed that the iterative compressive sensing reconstruction algorithm (IDA) could be used as a technique for post-empathy enhancement. IDA exchanges real-time operations for GPU-assisted efficiency gains, resulting in a PSNR of 1.31dB and a Structural Similarity Index Measure (SSIM) of 0.0152. According to [5] study analysis, this outperformed the PSNR output of a sophisticated deep neural network with a value of 0.4dB and SSIM of 0.0126.

Study on [6], utilizing DCT and DWT techniques for video compression which is implemented on six different RGB videos that have a resolution of 750x750. The primary purpose of this study is to analyze the performance of planned DCT, DWT and Arithmetic coding algorithms for a wide variety of video and image compression levels and applications. The main benefit of using DCT-DWT and Arithmetic coding as a hybrid scheme of transform techniques is that it delivers a high compression ratio without sacrificing video quality. According to a study on [6], the results of this compression video are successful in terms of data compression, indicated by a high compression ratio with the value of more than 35dB and PSNR results with the value of more than 70%.

Study on [7], proposing an adaptive block-based framework for compressed video sampling that explores the temporal redundancy in order to maximize sampling performance. Adaptive blocks are designed to allow different regions to move independently, classifying blocks into different categories based on interframe correlation and adjusting sampling and reconstruction strategies accordingly. According to the simulation results, the proposed framework reduces the number of sampled measurements by 52% to 80% while still meeting the consistency constraint on the reconstructed frames. Compared to previous works, the proposed scheme increases the accuracy of the restored frames and achieves an average PSNR gain of 0.8 dB to 5.4 dB.

Study on [8], using a fractal method to compress offline video with *.Avi format. Split the video frame, convert the RGB color to YUV, and then compress it with the fractal process. According to a study on [8], this compression video with the fractal method is very successful in terms of data compression since it produces a compression ratio greater than 50% of the original file size, but it takes a long time to compress and decompress it. In the implementation of the [8] study, a video with a duration of 17 seconds with 30 frames per second (fps) was successfully compressed using the fractal method with a duration exceeding 2 hours for one video.

Study on [9], Implementing CS for video processing and reconstruction, which video frames are reconstructed using the basis pursuit. This study uses a grayscale video which has a frame rate of 15 fps, 80x60 pixels resolution, and a duration of 8 seconds. The results indicate that a larger measurement matrix size results in higher quality. The experiment with $M = 80\%$ can achieve a PSNR of 36.11 dB on average.

This bachelor's thesis applies CS to video based on difference block frame using the DCT algorithm to achieve video compression performance with a high compression ratio. The video compression is processed with the help of Matlab software. A video is a moving picture made up of a series of images, each of which is referred to as a frame. The number of images taken in a second is expressed in fps. There are more than 20 frames per second in the videos used. The frames are processed through the difference frame method and then divided into 8x8 blocks. The blocks are then converted into sparse signals using the DCT algorithm. The DCT is a lossy compression scheme that transforms NxN blocks from the spatial domain to the frequency domain [10]. Next, threshold value is applied in order to maximize compression and achieve a high compression ratio. Then, after the signal becomes sparse, as indicated by the number of 0 values, the signal is compressed using CS and reconstructed using the OMP algorithm.

1.2 Problem Formulation

The following are the formulations of the problem carried out to complete this bachelor's thesis:

1. How to design and simulate a compression system on uncompressed *.avi

video using CS method based on difference block frame?

2. How is the effect of the number of CS acquisition samples (*M* parameter) on the performance parameters, namely compression ratio, SSIM, and PSNR?

3. How is the effect of the threshold value on the performance parameters, namely compression ratio, SSIM, and PSNR?

4. How is the size of the output video file generated by the designed system?

1.3 Objective

The following are the objectives of this bachelor's thesis:

1. Design and simulate compression system on uncompressed *.avi video using CS method based on difference block frame to be compressed *.avi video.

2. Analyze the effect of the number of CS acquisition samples (*M* parameter) on performance parameters, namely compression ratio, PSNR, and SSIM.

3. Analyze the effect of threshold parameter on performance parameters, namely compression ratio, PSNR, and SSIM.

4. Obtain the file size of the compressed video output with a high compression ratio.

1.4 Scope of Work

Related to the problem formulation, the scope of problem can be identified as follows:

1. The input data is a RGB video that convert to grayscale video in an uncompressed *.avi format file without audio and the CS process only for the images.

2. The performance results compared are between the original uncompressed grayscale *.avi video and the compressed grayscale *.avi video.

3. There are 10 grayscale videos to be processed as the video hosts.

4. The frame difference method is used as the interframe technique. The frames are then processed using the blockproc function and threshold value is applied, allowing CS to be performed on that difference block frame.

5. The compression system uses the DCT as the sparse method and reconstructed using CS method with OMP algorithm.

6. Parameters used for determining the performance of video compression are by calculating performance based on compression ratio, SSIM, and PSNR.

1.5 Research Method

The research methodology carried out on this thesis are:

1. Literature Study
   Obtaining pieces of information based on reading materials from the internet, books, journals, master's thesis, and articles that can support the topics discussed in this thesis for learning materials in understanding the concept of methods to be used.

2. Research and Discussion of Guidance
   Conducting discussions and guidance about this bachelor's thesis with the supervisors in order to gain direction and ensure that research proceeds as planned.

3. Data Collection
   Collecting data in the form of video as input data and source code for designing simulation systems to aid system design.

4. System Design Simulation
   Implementing CS with the DCT method using the Matlab programming language.

5. Analysis
   Analyze the program's success by examining its output. The program is successful if it is able to compress the video with a change in the original video storage size, which is reduced to a smaller size and becomes the output video compression size.

1.6 Bachelor's Thesis Organization

The rest of this thesis is organized as follows:

- Chapter 2 BASIC CONCEPT
  This chapter discusses the basic theory of digital video, CS, DCT algorithm, and test parameters used to determine the quality of the video compression.
- Chapter 3 SYSTEM DESIGN
  This chapter provides the workflow, experimental model diagram, and the steps of applying CS on video based on adaptive block used DCT algorithm.
- Chapter 4 PERFORMANCE EVALUATION
  This chapter conducts system testing and analysis of the results obtained from the system method and implementation chapter.
- Chapter 5 CONCLUSIONS AND SUGGESTIONS
  This chapter provides the conclusions and suggestions for this thesis.