Design of Video/Image Compression System

EE596 – Image and Video Coding

Mini Project

Dasanayake D.M.M.T.
E/15/056

Introduction

Hybrid video coding is an advanced coding technique which is derived from both predictive coding and transform coding. Hybrid video coding framework is commonly used in modern video coding standards, e.g., H.26x, MPEG2/4, AVS, HEVC etc.

In this mini project, A simplified hybrid video codec is implemented with coding tools like Discrete Cosine Transformation, Quantization, and Entropy Coding etc. and the impact of each coding tool on the codec's performance will also be investigated.

This mini project is contained with an image compression system and a video compression system. The video compression system is designed with the aid of image compression system. Using the image compression system, a RGB image can be compressed in Gray Scale. A RGB video can be compressed in Gray Scale using the video compression system.

Here, it will be able to compress a square shape or a rectangular shape image and for video compression, it will be able to decide the number of frames used to compress a given video.

Basic concepts used for Compression System

Basic concepts used for image and video compression system are;

Discrete Cosine Transform (DCT)

The Discrete Cosine Transform helps separate the image into parts of differing importance. The Discrete Cosine Transform is similar to the Discrete Fourier Transform, it transforms a signal or image from the spatial domain to the frequency domain.

Quantization

Quantization is a lossy compression technique achieved by compressing a range of values to a single quantum value. A typical video codec works by breaking the picture into discrete blocks. These blocks can then be subjected to Discrete Cosine Transform to calculate the frequency components, both horizontally and vertically. The resulting block is then pre-multiplied by the quantization scale code and divided element-wise by the quantization matrix, and rounding each resultant element.

Huffman Coding

Huffman Coding is a lossless algorithm for doing data compression and it forms the basic idea behind file compression. In this algorithm, a variable-length code is assigned to input different characters. The code length is related to how frequently characters are used. Most frequent characters have the smallest codes and longer codes for least frequent characters.

Motion Estimation and Motion Vectors

Motion Estimation examines the movement of objects in an image sequence to try to obtain vectors representing the estimated motion.

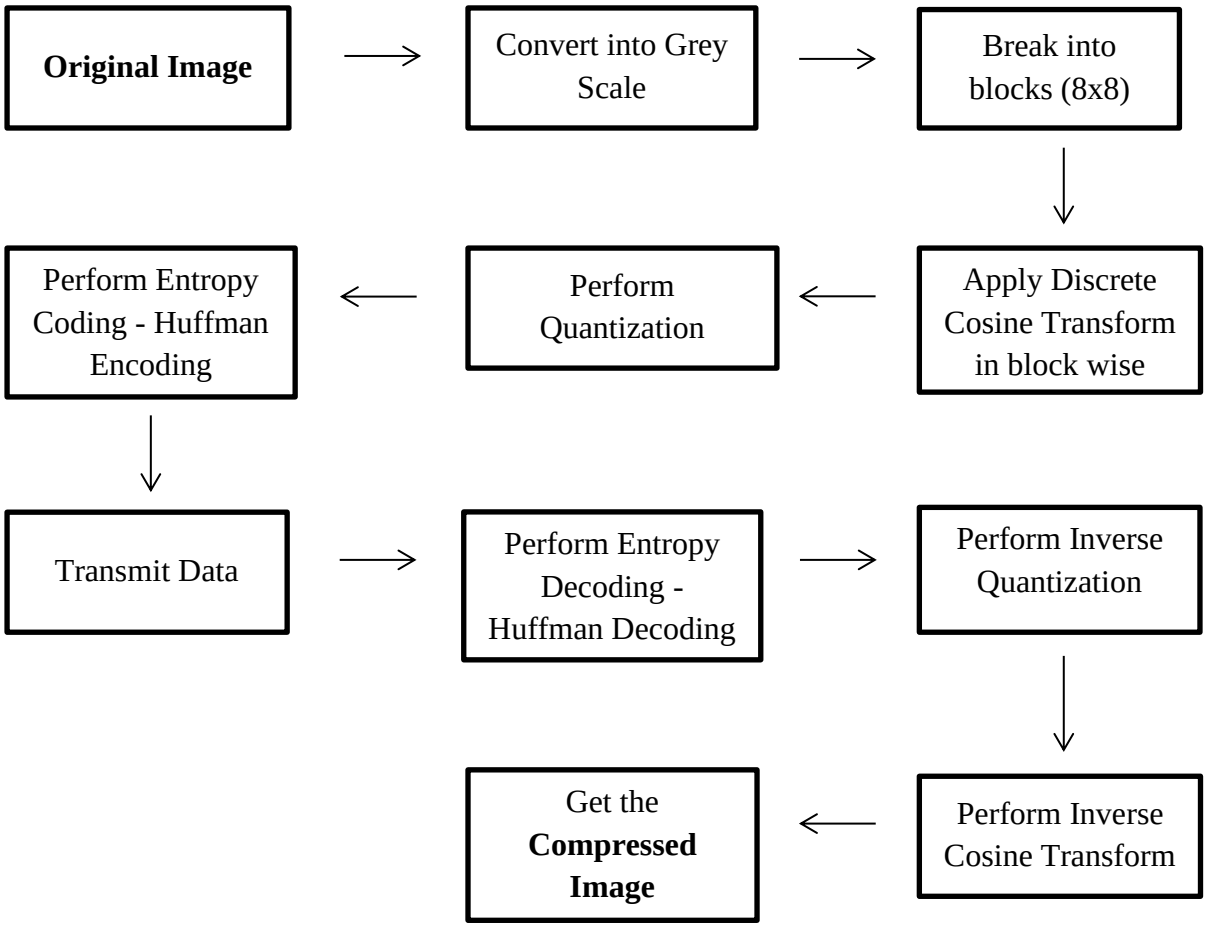
Motion Compensation

Motion Compensation is an algorithmic technique used to predict a frame in a video, given the previous and/or future frames by accounting for motion of the camera and/or objects in the video.

Image Compression System

This image compression system consists of concepts like Array Padding, Discrete Cosine Transform, Quantization, and Huffman Coding etc.

Flow Chart



Pseudo Code

1. Get the original image, read it and convert to Gray Scale.
2. Break the image into 8X8 macro blocks. If the image is rectangular shape, perform Array Padding to obtain a square shape.
3. Apply Discrete Cosine Transform block by block to that image.
4. Apply Quantization block by block to the image.
5. Apply Huffman Encoding for the quantized image and save the bit stream to a text file and obtain the Huffman dictionary.
6. Read the text file at the decoder and extract the bit stream.
7. Perform Huffman Decoding with the aid of obtained Huffman Dictionary and get the output image data.

8. Perform Dequantization to that image data.
9. Apply Inverse Discrete Cosine Transform to the dequantized results.
10. Save the reconstructed image and output the Compressed Image.

Results

Figure 1: RGB image which have to be compressed

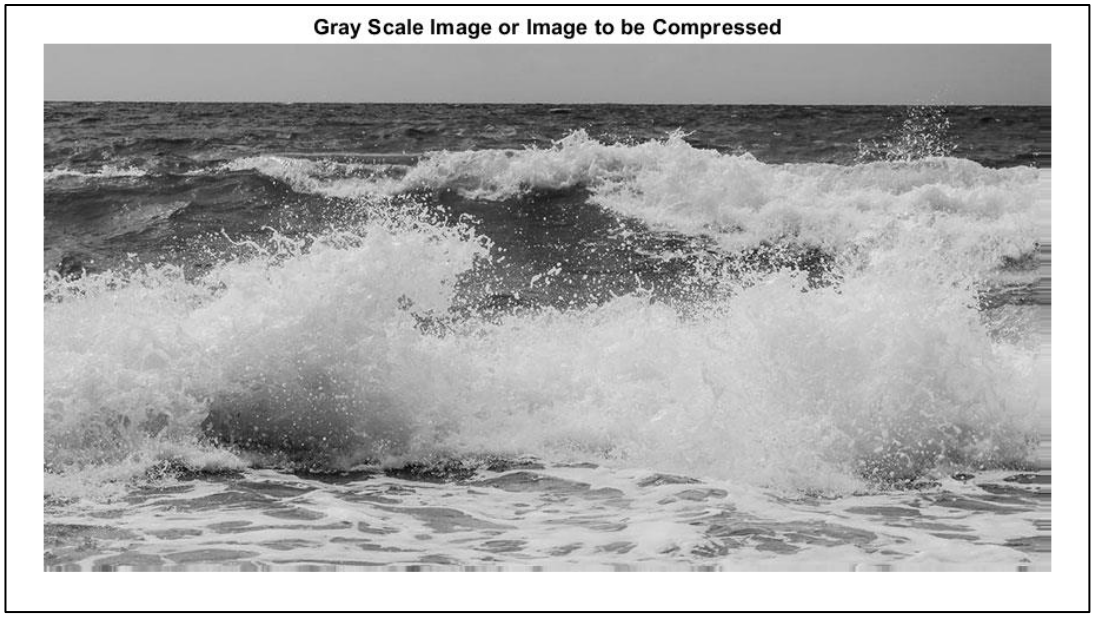


Figure 2: Gray Scale image which have to be compressed

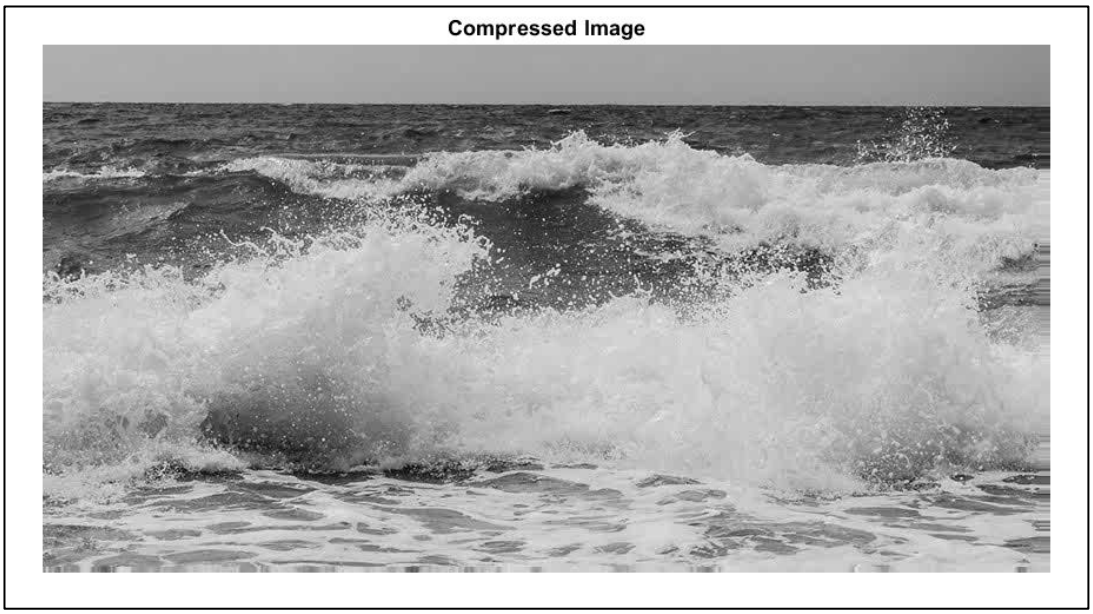
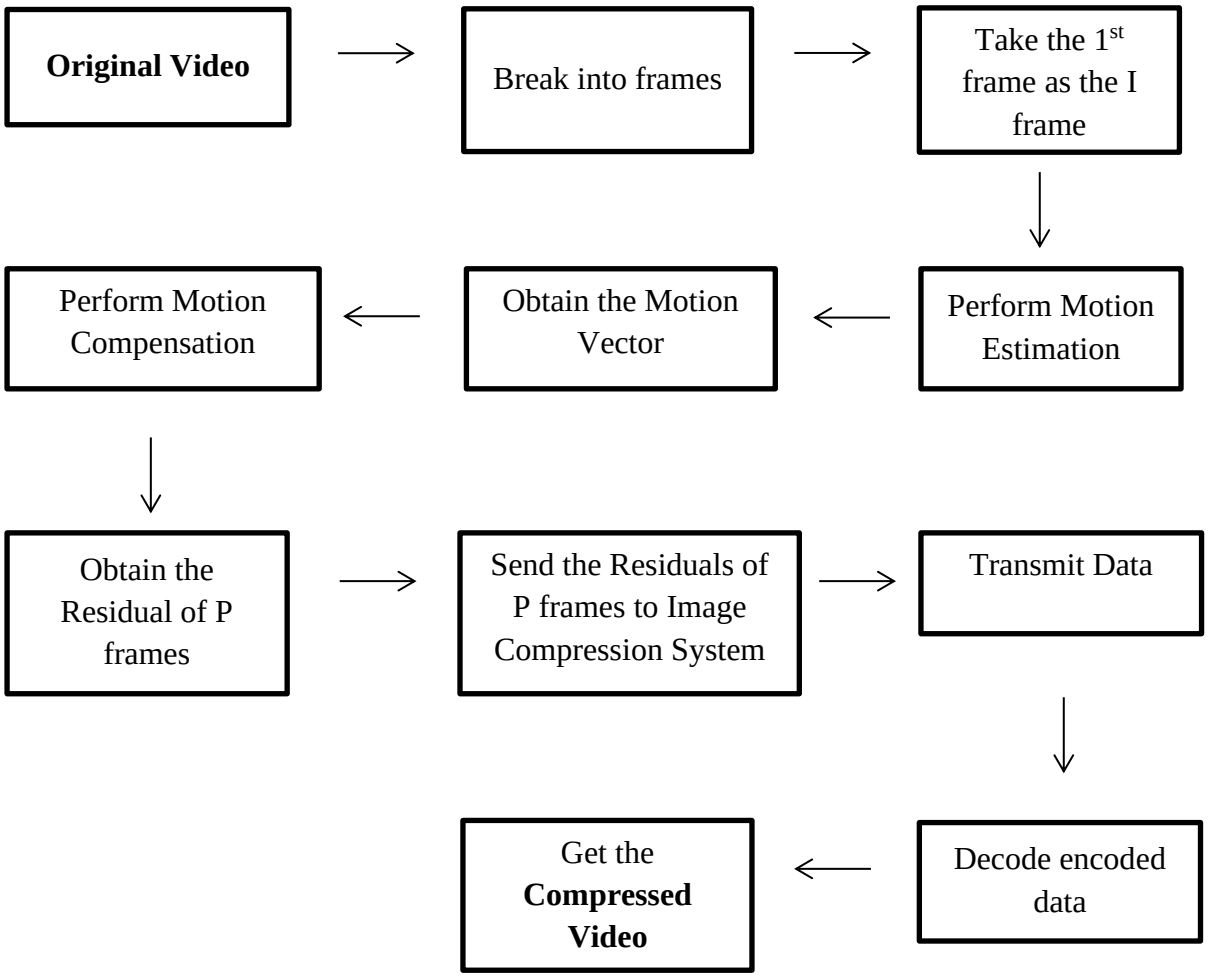


Figure 3: Compressed image

Video Compression System

This video compression system consists of concepts like Motion estimation, Motion Vectors, and Motion Compensation etc. with the help of Image Compression System.

Flow Chart



Pseudo Code

1. Get the original video, read it and break the video into frames.
2. Get the first frame as the I frame.
3. Perform Motion Estimation.
4. Calculate Motion Vectors using SAD values.
5. Perform Motion Compensation to obtain predicted frames.
6. Obtain the Residuals of P frames using predicted frames obtained.

7. Input that to the Image Compression System, obtain encoded Residuals of P frames and save them into a text file.
8. Read the text file and extract the Residuals of P frames.
9. Reconstruct the frames using extracted Residual data and Motion Vectors.
10. Output the Compressed Video.

Results

Original video which have to be compressed	→	Original Video
Frames used for the compression process	→	Frames
Compressed image	→	Compressed Video