Ryerson University

Department of Electrical & Computer Engineering
Faculty of Engineering & Architectural Science

Course Title:	ELE725
Course Number:	ELE725
Semester/Year (e.g.F2016)	W2020

Instructor:	Dr. Naimul Khan

Assignment/Lab Number:	3
Assignment/Lab Title:	lab3

Submission Date:	20/03/2020
Due Date:	20/03/2020

Student LAST Name	Student FIRST Name	Student Number	Section	Signature*
Zou	Yun Peng	500628304	2	

Reset Form

*By signing above you attest that you have contributed to this written lab report and confirm that all work you have contributed to this lab report is your own work. Any suspicion of copying or plagiarism in this work will result in an investigation of Academic Misconduct and may result in a "0" on the work, an "F" in the course, or possibly more severe penalties, as well as a Disciplinary Notice on your academic record under the Student Code of Academic Conduct, which can be found online at: http://www.ryerson.ca/senate/current/pol60.pdf

ELE725 LAB 3 (Intra- and Inter-Frame Coding)

Name: Yun Peng Zou
FEAS
Toronto, Canada
ypzou@ryerson.ca

Abstact- In this lab, core princpinciples of coding images frames of jpeg encoding is though the use of MATLAB and a grayscale image. The two image encoding is intra-frame coding and interframe coding using DPCM. The two encoding were successful implemented and compression was achieved. The trade off between compressing the data is distortion rate. As the compress rate goes up, the distortion rate goes up as well. Also, encoding and decoding the data will take a noticeable amount of time based upon system specifications.

I. INTRODUCTION

In the modern era, image data has become very large as the resolution and bit per pixel goes up to create a better quality image. The need for compact storage is on demand size increases. The jpeg compression encoder is the most popular encoder for image compression. Two ways jpeg does compression is though intra-frame and interframe coding. Intra-frame is compression based on within a single frame while interframe is between frames and will be the main focus of the lab.

II. THEORY

In intra-frame coding, the image is spilt into 8 pixel by 8 pixel blocks. The pixel information is encoded using the Discrete Cosine Tranformation (DCT). [1] The Discrete Cosine Tranformation evluates the spectral properties of the signal. [2] The DCT is given by the equation below :

(1)

$$F(u,v) = \frac{2}{\sqrt{MN}} C(u)C(v) \sum_{x=0}^{M-1} \sum_{y=0}^{N-1} \cos\frac{(2x+1)u\pi}{2M} \cos\frac{(2y+1)v\pi}{2N} f(x,y)$$

$$where: C(\xi) = \begin{cases} \frac{\sqrt{2}}{2} & if \xi = 0, \\ 1 & otherwise. \end{cases}$$

The signal is quantized by dividing by the jpeg quantization table and round. These values are sent to a reciever. The reciever will have to dequntize the value and do the inverse DCT tranformation.

In interframe coding, only one frame of the image is saved and the rest of the image is obtained though a predictor and a difference frame. The predictor used in this lab is :

	C	B		
	A	X		

Prediction Index	Prediction
0	No prediction
1	A
2	B
3	C
4	A + B – C
5	A + ((B – C)/2)
6	B + ((A – C)/2)
7	(A + B)/2

Figure 1: Predictor used in lab [1]

The predictor will guess which pixel from the previous frame can be used to make up the next frame. The predictor will not have perfect predictions so a difference frame needs to be stored which contains the value of difference between predicted value and obtained value. This compresses frames based on spatial redundances as the difference between two frames is very little unless a change in scenery or a lot of motion happens.

III. METHODOLOGY

A. Part 1 Intra-Frame Coding

The image was obtained using imread function in matlab.The image was gray scaled using the rgb2gray function. The image was subsection by iterating though the image and getting subsection based upon the number of parts needed for the subsection. The DCT tranformation was appiled to the sample by using the DCT dct2 command from matlab and the values were log and absolute valued and reshaped to 1 row so a hisgram could be obtained though the histogram command. The original signal is reobtained though the idct2 command. To zero the DC, high frequency, low frequency components, the matrix was zigzagged though with a for loop and the values were set to zero if the indexes were 0,0 or the frequency was passed 50 or below 10 depending on the components getting zeroed out.

To divide an entire image into 8x8 pixels, first the size of the image was determined using the size function. The number of blocks was determined by divideding the size of image by 64. To simplify coding, the image used was divisible into 8x8 blocks. In a for loop iterate though the entire image, block by block and do the dct2 for the block. The block was then stored in a (8,8, i) array i being the number of block the iteration is at. After the image is spilt into blocks of 8, the blocks are divided by a quantization table. The table was obtained based on jpeg compression table and was set in an array.Each value was divided by its subsequent value and round by a nested triple for loop. To recontruct the signal, the inverse is done. The value are mutiplied by the values in the quantization table. The block is changed from a (8,8, number of blocks) back into a (m,n) matrix which in the case of the lab is (512, 512) though the useage of for loops. The inverse idct2 is applied before the value is stored and the signal is reconstructed with very little data loss. The MSE, SNR, PSNR are found using immse, snr , psnr from matlab.

B. Part 2 Inter-Frame Coding

The video was read into matlab though the videoreader command. Frames were read from the video using readFrame and 11 frames were stored. The frames were grayscaled using rgb2gray. The first DPCM was the previous frame so the difference in the difference between two frame. To contruct the DCMP, the previous frame value was input and the frame value was iterated though. The predictor places the next pixel value based on the prediction index from Figure 1 and corner cases were just filled in based upon the current pixel, if the index goes out of bound. The entropy was caculated though the built in entropy function and histogram was contructed using build in histogram function. To caculate the compression ratio, sum the total number of bits for the original ten frames divide the number by the sum of the entropy caculated. The original bits per pixel for the frame is 8 since uint8 is 8 bits.

IV. RESULTS

A. Part 1 Intra-Frame Coding

163	162	161	160
162	162	162	161
162	162	163	161
162	163	163	161

Figure 2: input sample of 4x4

647.5000	1.9598	-1.5000	0.8118
-1.1152	1.1036	1.1152	-0.2500
0	0.9239	0	-0.3827
-0.0793	0.2500	0.0793	0.3964

Figure 3: coefficent sample of 4x4

1.2852e+03	5.1947	4.0358	-0.1134	-0.2500	0.2337	-7.8954	9.9472
8.2109	-2.0030	2.0033	-5.8583	4.1335	0.1536	-0.0724	0.0366
-5.9424	-0.1653	-3.5695	4.0386	-0.0957	-0.1944	-0.1250	0.3406
1.7874	0.4993	3.9078	0.2936	-0.1381	-0.2134	-0.4229	-0.0374
0	-0.1964	0	0.3468	0	-0.0690	0	-0.2940
-0.1587	0.0322	-0.7083	0.2134	-0.1181	-0.0562	0.4694	-0.6346
0.0260	-0.1483	0.1250	0.4267	-0.2310	-0.2228	0.3195	0.7105
0.4862	-0.0366	0.5246	0.4803	0.0794	0.1469	0.1856	-0.2345

Figure 4: coefficent sample of 8x8

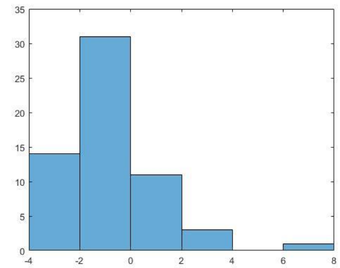

Figure 5: Coefficient histogram 8x8

When the DC component gets zeroed out, the signal can not be reconstructed. The signal cannot be fully reconstructed without the low frequency components, there will be slight error depending on the portion. The signal can be reconstructed with slight error of in each cell of the number is rounded without the high frequency.

Figure 6: Original Image

Figure 7: Image from DCT transform coefficient

Figure 8: Reconstructed Image

MSE = 18.1070

SNR = 0.0000022544

PSNR = 35.5523

B. Part 2 Inter-Frame Coding

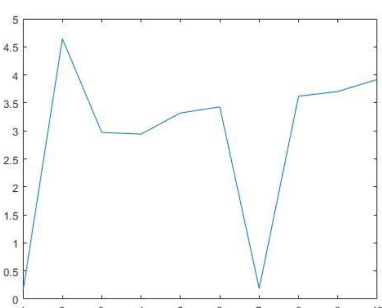

Figure 9: Plot of entropy over 10 frames

Compression Ratio Using ideal = 2.4277

Figure 10: sample frame 2 from video

Figure 11: sample frame 11 from video

Figure 12: Sample difference frame of figure 10 and 11

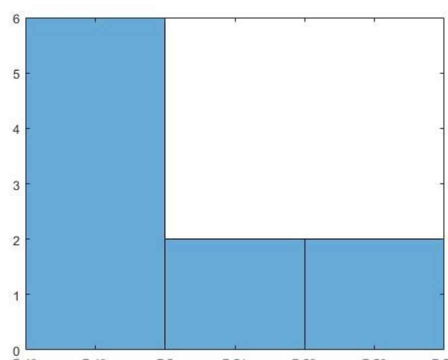

Figure 13: Histogram of raw image entropy

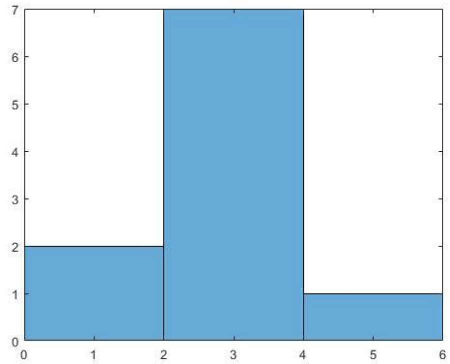

Figure 14: Histogram of DPCM entropy (mode 1)

V. DISCUSSION

A. Part 1 Intra-Frame Coding

The coefficient distribution observer for DCT transformation is mainly on low frequency with a few high frequency components and one extremely high frequency component. The reason for this is that the extreme high frequency is the dc component while the rest is the ac component. When zero out the high frequency components has very little impact on the image, the high frequency component that is because the high frequency has very little effect on the overall signal as seen at figure 3 and figure 4 where the high frequency is less than 1. The low frequency has a significant impact for some signals but no impact for others which is highly dependent on the signal. If the dc component is zeroed out, the signal cannot be reconstructed at all. This can be seen though the quantization table as the higher frequency on the table are larger. The quantization table zero out all high frequency component and some low frequency component to save bits. The reconstruction of the image is pretty much identical as seen from figure 6 and 8 despite zeroing out the majority of frequencies. The zeroing of frequency decreases space because instead of saving the bit as 8 bit, it can be saved using 1 bit. The mse, snr, psnr is very low for the tested image. The distortion value increase when the image is more details because the compression ratio gets larger which means the distortion ration increases.

B. Part 2 Inter-Frame Coding

The difference in the frame is where movement happens in between frames. If the frame has very little motion, the difference is very tiny but with movement it creates differences as shown from figure 12. As higher motion values are detected, the difference is greater. From figure 13, the raw image has an entropy of around 7.5. This can be compressed using a DPCM predictor as shown in figure 14, it will substantially decrease most of the values by at least half. Better predictor will compress the frames better.

VI. CONCLUSION

In conclusion, intra-frame coding and inter-frame coding provides great compression with low distortion. The compression of a video can use both intra-frame and inter-frame coding to achieve greater levels of compression. The developer must balance both compression ratio and distortion rate to effectively create a compression algorithm.

REFERENCE

[1] Faculty of Engineering and Architectural Science. (2020, Winter). ELE725 Lab 3 Manual: Lossless Coding. Ryerson University.

.