**EE475 Compression**

**Fall 2019**

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| **SUB-SAMPLING** | **PCM** | **DPCM** | **VQ** | **DCT** | **JPEG** | **Total** |
|  |  |  |  |  |  | **/200** |

1. **Try to give concise one or two sentence answers, explanations**
2. **Copy and past images from the VC Demo program to enhance your answers**

**IMAGE COMPRESSION SOFTWARE**

In the following, study the image processing function experimenting with various parameters, report your experience briefly in a few sentences. Copy and paste the images and graphs to accompany your commentaries.

**SUBSAMPLING**

* Inspect the spectrum of the image Build512B.bmp. The DC component is in the center. Give explanations for the observed line structures. Note: *SS Button 🡪 Spectrum 🡪 Apply*

1. Why there are strong horizontal and vertical spectral lines passing through the origin?
2. Plot the image and its spectrum side by side, and identify two main image structures corresponding to two strong lines in the spectrum.
3. Which details in the image would cause the highest frequency components in the spectrum?

* Subsample by a factor of 2, once without anti-aliasing filter and then with 17-tap aliasing filter. Note: *SS 🡪 Factor 🡪 2 , check Apply Subsampling box, check Blow-up Subsampled Image box*

*SS 🡪 Filter 🡪 Check or uncheck Apply anti-alias filter, Set the number of taps to 17.*

1. Plot the images and their spectra side by side: Can you pinpoint the aliasing events in the subsampled image without anti-aliasing filtering? Where would you expect these distortions to occur?
2. Does anti-aliasing filter improve it? Explain the improvements in the image and the changes in the spectrum.
3. In turn, does the anti-aliasing filter cause any distortions? Explain the difference in the spectra of the subsampled images with and without anti-aliasing filters.
4. Subsample the Clown image by factors of 2, 4 and 8, and plot the spectra, side-by-side.

**PCM**

* Study the images Lena256B, Clown256B, Odie256B. Note: *PCM button 🡪 Select bitrate 🡪 Apply. Do for bit rates 7 to 1.*

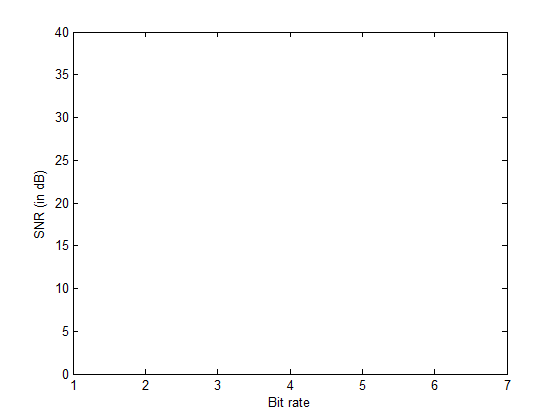
1. At which rate do the coding artifacts become objectionable? Where do you observe these artifacts most clearly? At this rate how many gray values are available to represent the image?

|  |  |  |
| --- | --- | --- |
|  | Bit-rate at which artifacts become objectionable | Gray values at this rate |
| Lena256B |  |  |
| Clown256B |  |  |
| Odie256B |  |  |

For Lena256B, Fill in the table.

|  |  |
| --- | --- |
| Bit rate | SNR (dB) |
| 1 |  |
| 2 |  |
| 3 |  |
| 4 |  |
| 5 |  |
| 6 |  |
| 7 |  |

1. Draw an SNR-versus-bit-rate plot for Lena256B. Explain the reason why you are getting such a slope for this curve.
2. Explain the behavior of Odie under PCM, which is quite different than that of Lena. What distortions occur in Odie? What distortions occur in Lena?



**DPCM**

* Select the Lena256B image and the 1-D predictor. Note: *DPCM button 🡪 Model 🡪 Select the first prediction model. Select Bit Rate from 1 to 6.*

1. Carry out compression at bit rates 6 to 1 bpp and obtain the SNR – Observe the gain over PCM.

For Lena256B, Fill in the table.

|  |  |  |
| --- | --- | --- |
| Bit rate | DPCM SNR (dB) | PCM SNR (dB)  Copy from previous table |
| 1 |  |  |
| 2 |  |  |
| 3 |  |  |
| 4 |  |  |
| 5 |  |  |
| 6 |  |  |
| 7 |  |  |

1. Compare visually the images with their PCM version and state at what rate DPCM achieves a performance equal to that of PCM?
2. Use four types of prediction region for DPCM at bit rate level 5 and comment on the performance differences. Sketch the prediction context (the orange diamond is the current pixel) and write the coefficient values. Plot the Lena image and its three prediction error images in a quadriptych.
3. Observe the correlation matrix. What is the theoretical gain over PCM in terms of bit rate reduction at the same quality? Does the experimental result match the theory? Recall the prediction gain for one-tap predictor:  where r(1) is the normalized correlation coefficient.
4. If DPCM is used on the noise image below, what do you expect the coding gain to be? What would be the prediction coefficients?



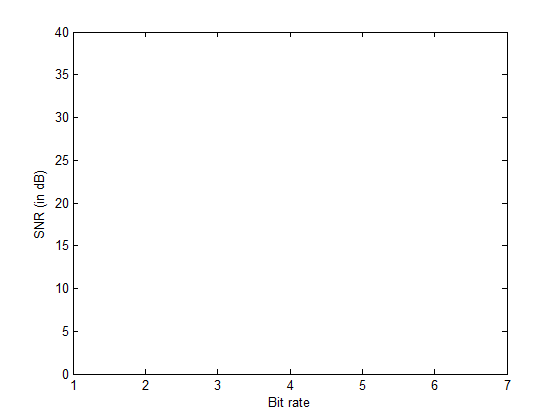
**VQ**

Load the predesigned codebook Standard\_4x4\_min1-max12.cbk. This codebook has been designed on 4 other images other than Lena.

Note: *VQ button 🡪 Codebook 🡪 Load codebook 🡪 Select Standard\_4x4\_min1-max12.cbk 🡪 Open*

*Bits 🡪 Select*

1. Make an SNR-bit rate plot of VQ on the same curve as PCM/DPCM, i.e., copy first the PCM-DPCM plot. Note: Be careful to first calculate the VQ bit rate (bits/pixel) before you plot.



**Start this curve from 0.**

1. What type distortion do you observe at low bit rates? Does it appear to be additive white Gaussian noise? Plot the VQ coded images.
2. Can you use codebooks obtained by Lena-type images on cartoons and maps? If yes, how? If not, why not?

**DCT**

Select the Lena256B image. Select PCM compression for all DCT coefficients. Pick c = 0.75 as the exponent power of the generalized Gaussian distribution.

Note: *DCT button 🡪 Size tab 🡪 Check encoding of DCT coefficients 🡪 Set transform size as 2x2, 4x4 and 8x8 in turn and press apply*

*Coefs tab 🡪 Select PCM for First and Others, Select C value as 0.75*

*Bitrate 🡪 Check the entropy coding. Select each of the bit rates in turn.*

1. Study the block sizes of 2x2, 4x4 and 8x8, 16x16. Explain what you observe in the “DCT: Original Coefficients” window. Plot both the un-coded and coded versions of the DCT coefficients, side by side for the four block size choices, as in the example below.

* *

*Uncoded Coded*

1. Use again entropy coding, but with the wrong signal model. Assume wrongly that the shape parameter of the generalized Gaussian distribution is a = 2 (hence pure Gaussian), and not a = 0.75. What do you observe? How is coding performance affected?

**JPEG**

* Use the Lena256B image and the standard luminance normalization matrix. Note: *JPEG button 🡪 Huffman tab 🡪 Select FLC, Standard VLC, Optimal VLC in turn*

*JPEG button 🡪 Bit rate tab 🡪 Bitrate 🡪 Select bit rate as 0.5, 0.8, 1.0, 1.5, 2.0, 2.5, 3.0 in turn 🡪 Apply*

Write down the Encoded Bit Rate in bpp, Optimized Quality Factor and the corresponding SNR. (FLC: Fixed Length Coding, VLC: Variable Length Coding)

1. Fill the SNR-bit rate tables, one for each entropy coding choice. At the same time, write down the corresponding QF: Quality Factors. How much additional SNR does entropy-coding give?

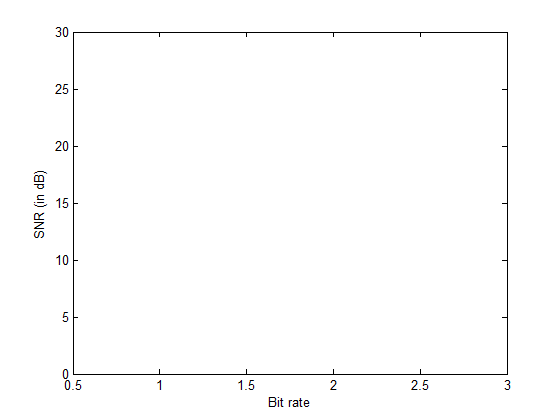
|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Bit rate | FLC | | | Standard VLC | | |
|  | Encoded Bit Rate | Quality factor | SNR (dB) | Encoded Bit Rate | Quality factor | SNR |
| 0.5 |  |  |  |  |  |  |
| 0.8 |  |  |  |  |  |  |
| 1.0 |  |  |  |  |  |  |
| 1.5 |  |  |  |  |  |  |
| 2.0 |  |  |  |  |  |  |
| 2.5 |  |  |  |  |  |  |
| 3.0 |  |  |  |  |  |  |

1. Compare the resulting SNRs from a “flat normalization matrix” against that of the “standard luminance normalization” and comment.

Note: *Repeat above selecting: Quant 🡪 Choose Flat*

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Bit rate | Fixed weighting | | | Standard weighting (psychovisual) | | |
|  | Encoded Bit Rate (b | Quality factor | SNR (dB) | Encoded Bit Rate | Quality factor | SNR |
| 0.5 |  |  |  |  |  |  |
| 0.8 |  |  |  |  |  |  |
| 1.0 |  |  |  |  |  |  |
| 1.5 |  |  |  |  |  |  |
| 2.0 |  |  |  |  |  |  |
| 2.5 |  |  |  |  |  |  |
| 3.0 |  |  |  |  |  |  |

Draw the four bit rate-SNR curves on the same plot:



1. Observe the images with flat versus standard normalization matrix at 0.5 bpp. Their SNRs are equal, yet one looks better than the other one. Explain this dilemma.