1 point

1. Suppose we have an image with 256 different gray levels. All the gray values appear an equal number of times.

Will variable-length coding lead to any compression in this image without additional processing?

- ○ Yes
- ● No

If all gray levels have equal probability then all the codes will have equal length and no compression is obtained.

1 point

2. How can lossless image compression be achieved for the image in Question 1?

- ● Via predictive coding.
- ○ Erasing pixels.
- ○ Performing a DCT.
- ○ Lossless compression will never be achieved for such image.

Predictive coding can exploit spatial redundancy, leading to non-uniform distributions of the prediction error and then compressing via variable length coding.

1 point

3. How many unique sets of Huffman codes can you construct for an image with only 3 different pixel values (e.g., all the image is composed of 0s, 255s, and 128s)?

- ● 2
- ○ 5
- ○ 3
- ○ Infinity

The two codes are: (1) 0, 11, 10 and (2) 1, 00, 01. The codes are complements of one another. They are constructed by following the Huffman procedure for three symbols of arbitrary probability.

1 point

4. For an image with intensities 21, 95, 169 and 243; and respective probabilities 3/8, 1/8, 1/8, and 3/8; the length of the corresponding variable-length code created by the Huffman coding procedure are

- ○ 1, 4, 4, 1
- ○ 1, 2, 2, 1
- ● 1, 3, 3, 2
- ○ 2, 2, 2, 2

---

21 -> 1 95 -> 010 169 -> 011 243 -> 00

1 point

5. The main source of error (lossy compression) in JPEG is

- ● The quantization.
- ○ The division into 8x8 blocks.
- ○ The variable-length (Huffman) coding.
- ○ The DCT.

While DCT might introduce minor rounding errors, the key source of errors in lossy compression is quantization.

1 point

6. In lossless image compression, prediction can be based on any pixel in the image.

- ● False
- ○ True

Prediction can only be based on pixels already available to the decoder, meaning only on pixels that have already been encoded (this is the causal order of encoding).

1 point

7. A reason for using DCT (instead of Fourier, for example) in JPEG is

- ○ No particular reason
- ○ It is simpler to compute
- ◉ Its favorable periodicity property
- ○ It is real while Fourier is complex

Check the slide "Why DCT?" Its mirror/symmetry periodicity if favorable for working with blocks.

1 point

8. Since we must encode all pixels in the image, JPEG needs at least a bit per pixel and therefore in a 256 levels image (8 bits), it can only achieve up to 8:1 compression.

- ○ True
- ◉ False

Since after DCT and quantization, many coefficients become 0, JPEG can encode all of them together (end of block), achieving higher compression ratios.

1 point

9. In JPEG, if we double the quantization step, then we double the compression ratio.

- ○ True
- ◉ False

There are a number of reasons for this to be false, one of them being the presence of Huffman coding. Doubling the quantization means that each DCT coefficient will become a different symbol, and therefore will be coded differently. There is no way to know in advance the reduction in the number of bits resulting from this. An additional component is the presence of the end of block to code all the zeros.

1 point

10. Without JPEG or a similar compression technique, digital cameras will no be as popular as they are today.

- ◉ True
- ○ False

Compression is an enabling technology and one of the most important contributions in image processing. You will be able to store many fewer images in your cameras and they will take so long to transfer them that their use will be significantly limited.