



Course: Image Processing
Prof. Mylène C.Q. de Farias
Semester: 2017.1

LIST 01

Submission Date: 30/03/2017

Part 1 – Theory

Question 1.1: When we enter a dark theater on a bright day, it takes an appreciable interval of time before we can see well enough to find an empty seat. Which of the visual processes explained in Chapter 2 is at play at this situations?

Question 1.2: Image subtraction is often used in industrial applications for detecting missing components in product assembly. The approach is to store a “golden” image that corresponds to a correct assembly; this image is then subtracted from incoming images of the same product. Ideally, the difference would be zero if the products are assembled correctly. Difference images for products with missing components would be nonzero in the area they differ from the “golden” image. What conditions do you think have to be met in practice? In what situations the method may fail?

Question 1.3: Show that 2-D transforms with separable, symmetric kernels can be computed:

- Computing the 1-D transforms along individual rows (columns) of the input, and
- Computing 1-D transforms along the columns (rows) of the result of the step 1.

Question 1.4: Suppose that a flat area with center at (x_0, y_0) is illuminated by a light source with intensity distribution

$$i(x, y) = Ke^{-[(x-x_0)^2+(y-y_0)^2]}.$$

Assume, for simplicity, that the reflectance of the area is constant and equal to 1.0, and let $K = 255$.

- If the resulting image is digitized with k bits of intensity resolution, and the eye can detect abrupt changes of eight shades of intensity between adjacent pixels, what value of k will cause false contouring?
- Sketch the image for $k = 2$.

Problema 1.5:

- Develop an algorithm for converting a one-pixel-thick 8-path to a one-pixel-thick 4-path.

- b) What are the necessary conditions for a D_4 distance between two points p e q to be equal to the shortest 4-path between these points. Is this path unique?

Problema 1.6:

- (a) What effect would setting to zero the lower-order bit planes have on the histogram of an image (in general)?
- (b) What would be the effect on the histogram if we set to zero the higher-order bit planes instead?
- (c) Explain why the discrete histogram equalization technique does not, in general, yield a flat histogram.

Part 2 – Practice

Question 2.1: Visualizing High Dynamic Range (HDR) images:

- a) Read the material on HDR images available at the course homepage. What are the differences between ‘regular’ LDR (Low Dynamic Range) images and HDR images? How are HDR images generated and what are their advantages?
- b) A camera takes HDR photos of a scene with a contrast ratio equal to:

$$RC^{cena} = \frac{I_{max}^{cena}}{I_{min}^{cena}} = \frac{1000}{1}$$

where I_{max}^{cena} and I_{min}^{cena} are the maximum and minimum light intensity values, respectively. The camera has a pre-distortion γ factor equal to 3, i.e.:

$$\frac{V_{max}}{V_{min}} = \frac{(I_{max}^{cena})^{1/3}}{(I_{min}^{cena})^{1/3}} = \frac{1000^{1/3}}{1}$$

where V_{max} and V_{min} are the tension outputs of the camera after pre-distortion factor is applied. The captured image is displayed on a monitor with $\gamma = 2$, i.e.:

$$RC^{disp} = \frac{(V_{max})^2}{(V_{min})^2}$$

What is the contrast ratio necessary to accommodate the dynamic range of the scene without saturating lighter and darker regions?

- c) Download the two HDR images (`hw1_memorial.hdr` and `hw1_atrium.hdr`) from the course homepage. Read the images using the Matlab function `hdrread`. What is the data type of the loaded image? Convert an image to grayscale using the Matlab function `rgb2gray`. Using `imshow`, visualize the image. Are you able to see the details of the image?
- d) Apply a γ transformation to the grayscale images obtained in (c). Vary the values of γ and visualize the images using `imshow`. For each image, what was the improvement provided by the transformation? Repeat the procedure applying the transformation in each of the image color components r (red), g (green) and b (blue). Try, again, to use the same γ values for the 3 color components. Next, experiment using different γ values for each of the 3 color components. Discuss the results obtained.

- e) From an HDR image, we can generate an LDR image with a better quality by using an adequate mapping function. One example of such a function is the following:

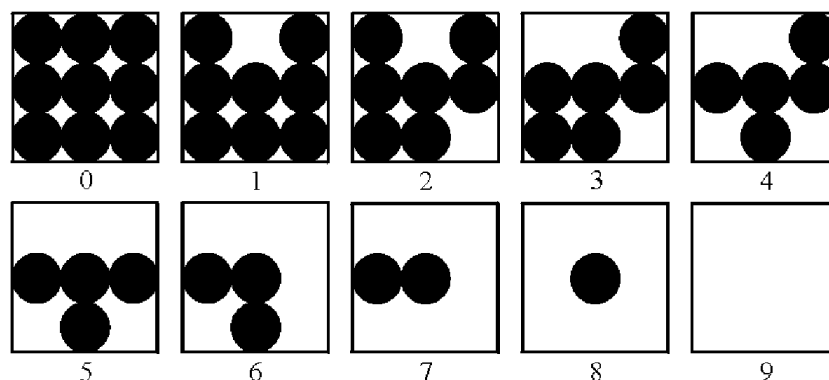
$$y = \left(\frac{x^n}{x^n + \sigma^n} \right)^{1/\gamma}$$

where x is the HDR image, y is the LDR image, and γ is the monitor gamma factor. Varying n and σ we can create an image with a good quality. (Hint: Start with $n = 1, 2$ and $\sigma = 2$). Investigate how this mapping works when we vary n and σ , describing the effects of each parameter.

Question 2.2: Solve the following question, without using ready Matlab functions (such as `imresize`).

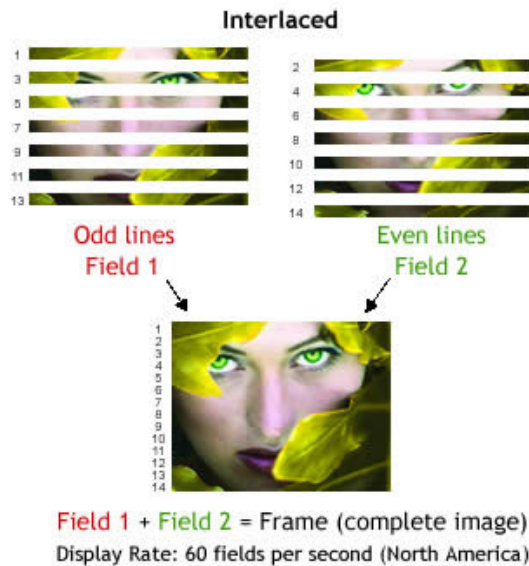
- Write a computer program capable of increasing and decreasing the spatial resolution of an image by pixel replication. Assume that the desired increase/decrease factors (n) are integers and should be given by the user.
- Use your program to shrink (reduce) the image by factors of 2, 4, 8 e 16. Next, use your program to increase the image back to the original resolution.
- Modify your program so that it uses the bilinear interpolation (instead of the replication). Repeat (b) and explain the results.
- Write a program that performs an image rotation, using the developed interpolation programs.
- Rotate an image by 15, 30, 60 e 90 degrees. Perform the rotation using both types of interpolations: (a) replication and (c) bilinear interpolation. Show the images obtained, discussing the differences in the results.

Question 2.3: The following figure shows ten shades of gray approximated by dot patterns. Each gray level is represented by a 3 x 3 pattern of black and white dots. A 3 x 3 area full of black dots is the approximation to gray-level black, or 0. Similarly, a 3 x 3 area of white dots represents gray level 9, or white. The other dot patterns are approximations to gray levels in between these two extremes. A gray-level printing scheme based on dots patterns such as these is called “halftoning.” Note that each pixel in an input image will correspond to 3 x 3 pixels on the printed image, so spatial resolution will be reduced to 33% of the original in both the vertical and horizontal direction. Size scaling as required in (a) may further reduce resolution, depending on the size of the input image.



- a) Write a halftoning computer program for printing gray-scale images based on the dot patterns just discussed. Your program must be able to scale the size of an input image so that it does not exceed the area available in a sheet of size 8.5 x 11 inches (21.6 x 27.9 cm). Your program must also scale the gray levels of the input image to span the full halftoning range.
- b) Write a program to generate a test pattern image consisting of a gray scale wedge of size 256 x 256, whose first column is all 0's, the next column is all 1's, and so on, with the last column being 255's. Print this image using your gray-scale printing program.
- c) Print book Figs. 2.22(a) through (c) of the textbook (download from the course homepage) using your gray-scale printing program. Do your results agree with the conclusions arrived at in the text in pgs. 64-65 and Fig. 2.23 (isopreference graph)? Explain.

Question 2.4: An *interlaced* video camera transmits one image with complete (full) resolution by transmitting first the odd lines (first field) and, then, the even lines (second field). The image with the full resolution can be recovered by combining the even and odd fields – see figure below.



Write a program to obtain the image with full resolution from the consecutive fields composed by odd and even lines. But, consider that the camera has a defect that degrades only the even fields. In the even fields, the intensity values of the pixels in each line have are shifted by the value of the average intensity value of the pixels for this line (i.e. the average intensity value of a line was subtracted from the pixels original intensity values). Your program must correct this degradation in a way that the reconstructed image (full resolution) does not show any visible degradations. The two fields (even and odd) are stored in 'mat' format (Matlab) and can be downloaded from the course homepage. To load the fieds use the Matlab "load" function.