COL783 Assignment 1
Due date: 29 August 2024

This assignment deals with basic image operations, intensity transformations, spatial filtering, and colour image processing.

The assignment is in two parts. Part 1 is being released on 8 August. Part 2 will be released once we have covered spatial filtering and colour image processing.

PART 1

1. You are given a small 100×100 image of the IITD CSE department's logo: cse-logo.png. Take another $M \times N$ image, for example a photograph you have taken yourself (resize it using any tool, e.g. GIMP, so that it is of reasonable size, $1000 \leq \max(M, N) \leq 1500$). Your task is to rescale the logo so its height and width are $0.25 \max(M, N)$, and then draw it in the bottom-left corner of the other image, making the white background of the logo transparent.

   a. Let the logo image be f. To determine the background pixels, find the most common intensity r^*, and create a new image α in which $\alpha(x, y) = 1$ if $|f(x, y) - r^*| \leq t$ and 0 otherwise. Choose a tolerance $t \geq 0$ which gives a reasonable separation of the foreground and background region. Report both r^* and t.

   b. Implement image rescaling using two interpolation algorithms: nearest neighbour and bilinear interpolation. Perform rescaling of both f and α to the target size mentioned above, and show the results for both algorithms.

   c. Drawing the logo f on top of g amounts to taking a weighted average of the corresponding pixels, $\alpha f + (1 - \alpha)g$, so that the logo shows up where $\alpha = 1$ and the other image shows where $\alpha = 0$. Note that this operation, taken literally, can only be applied when the two images are the same size, but here f is smaller even after rescaling. Figure out what to do in order to place the rescaled f at the bottom-left corner of g, and explain it in your report.

2. Choose one of the following high-dynamic-range images to work with: groveC.hdr, groveD.hdr, memorial.hdr, nave.hdr, vinesunset.hdr. (These were acquired by Paul Debevec in 1997.) The HDR file format is supported by most image I/O libraries. Loading it should give you an $M \times N \times 3$ array of floating-point numbers; convert it to a 1-component $M \times N$ image by averaging the 3 components of each pixel.

a. Report the maximum and minimum intensities in the image, and the contrast ratio $r_{\max}/r_{\min}$. Let us verify that it is difficult to adequately display this range simply by linearly scaling the intensities, $s = cr$ for some constant c. Produce one image in which c is chosen to map $r_{\max}$ to 255, and another in which c maps $r_{\min}$ to 1; in both cases, remember to clip the computed intensities to the valid range $[0, 255]$.

b. One way to visualize a very large range is to perform a log transformation, $s = a \log r + b$. Choose a, b so that the output intensities span $[0, 255]$, and display the resulting image.

c. Suppose we take logs, apply a linear transformation, and then undo the log: $s = \exp(a \log(r) + b)$. Choose a, b to map the intensity range $[r_{\min}, r_{\max}]$ to $[1, 255]$. In your report, show that this is equivalent to a gamma transformation.

d. Implement histogram equalization and test whether it works well to display the HDR image. Since the HDR image may have a very large number of intensity levels spanning many orders of magnitude, I suggest applying a log transformation, then choosing 256 histogram buckets spanning the range $[\log r_{\min}, \log r_{\max}]$. In your report, explain whether the log transformation should significantly affect the results of histogram equalization or not (apart from quantization issues).

e. Now that you have histogram equalization, you can also implement histogram matching. Demonstrate this by matching the HDR image's histogram to that of a real photograph of your choice (which need not be taken by you).

PART 2

3. To better preserve local detail, we can try using spatial filtering to reduce only the contrast of large-scale intensity variations. In this part, let us work in the "log domain", similar to problem 2(c): apply a log transformation, perform whatever spatial filtering operations, and in the end apply the inverse of the log transformation (i.e. exp) to the final image.

a. Implement Gaussian filtering with a user-specified σ. Use a separable kernel so that the time complexity is linear in σ rather than quadratic. Demonstrate the results on an image of your choice, with two significantly different choices of σ.

b. Take the log-HDR image, $\hat{f} = \log f$, and decompose it into a lowpass-filtered image $\hat{g}$ and a highpass-filtered image $\hat{h}$ using your Gaussian filter. Apply your choice of contrast reduction from problem 2 to only $\hat{g}$, recompose the image by adding the lowpass and highpass images, and finally undo the log. Show all intermediate images (with intensities of each suitably rescaled to the displayable range) as well as the final result.

c. Implement a bilateral filter with two user-specified parameters σ_s and σ_r. Apply it to the same image as in part (a), with parameters chosen to clearly demonstrate that

low-contrast variations are smoothed while high-contrast edges are preserved.

    d. Repeat part (b) using a bilateral filter instead of a Gaussian filter. Again, try to choose parameters to get the best image you can, in terms of enhancing local details while avoiding artifacts such as saturation and halos.

4. Now, we will process the original RGB values of the HDR image. We can use the HSI colour model to apply contrast reduction to the intensity component while preserving the original chromaticity.

    a. Visualize the structure of the HSI model by creating an image where hue is constant, saturation increases horizontally from 0 to 1, and intensity increases vertically from 0 to 1. Highlight colours which are outside the standard RGB gamut (i.e. any of R, G, or B is less than 0 or greater than 1) by assigning them a different "error" colour. Similarly, create one image where saturation is constant, and one where intensity is constant.

    b. Show that the HSI formulas in the textbook are meaningful even if we have HDR colours where R, G, B are arbitrarily large. In particular, show that if a colour $\mathbf{c} = (r, g, b)$ has HSI components (h, s, i), then for any multiple $\mathbf{c}' = (kr, kg, kb)$ we have $h' = h$, $s' = s$, and $i' = ki$ no matter how large k is.

    c. Process the colour HDR image by converting to HSI, applying contrast reduction from problem 3(d) to the I component, and converting back to RGB. Compare the results with applying contrast reduction to the R, G, and B components independently.

    d. The values in the original RGB image are in a "linear" colour space (i.e. proportional to the actual physical intensity of light), but your display probably expects gamma-corrected RGB values. Apply a gamma transformation of $\gamma = 1/2.2$, corresponding to the typical sRGB gamma, to the results of part (c), and display the images again. In your report, comment on the change in chromaticity and explain why it occurs.

SUBMISSION

Submit a zip file that contains (i) all your code for the assignment, and (ii) a PDF report that includes your results for each of the assignment problems. In the report, each output image must be accompanied with a brief description of the procedure used to create it, and the values of any relevant parameters.

All images should ideally be saved in PNG format, which is lossless and so does not cause any information loss (other than quantization if the intensities are not already in 8-bit format). JPEG is permitted if your submission PDF is becoming too big for the upload limit.

Your assignment should be submitted on Moodle before midnight on the due date. Only one person in a group needs to submit. Late days are counted with a quantization of 0.5 days: if you cannot finish the assignment by midnight, get some sleep and submit by noon the following day.

Separately, each of you will individually submit a short response in a separate form regarding how much you and your partner contributed to the work in this assignment.