

CSC442/542 Image Processing – Spring 2015 Programming Assignment #1: Point Processing
--

For your first image processing assignment, write a program to read in an image, display it, and apply a variety of image enhancement techniques. Offer the user the following menu options:

1. read an image from disk and display it (*File, Open*)
2. write an image to disk (*File, Save* and *File, Save As*)
3. restore the original image (*File, Revert*)
4. exit the program (*File, Exit*)
5. convert a (color) image to gray scale (use 30% red + 59% green + 11% blue)
6. negate an image
7. binary threshold an image (prompt user for threshold)
8. posterize an image to a specified number of intensity levels (prompt user for number)
9. increase/decrease image brightness (prompt user for amount)
10. increase/decrease image contrast using a linear ramp (prompt user for endpoints)
11. adjust image gamma using a power transformation (Equation 3.2-3) (prompt user for gamma)
12. compress the image dynamic range using a log transformation (Equation 3.2-2)
13. convert an image to discrete 8-level pseudocolor (e.g., Figure 6.20)
14. convert an image to continuous pseudocolor (e.g., Figures 6.22 and 6.25)
15. display an image histogram (use the ImageLib *displayHistogram* function)
16. perform an automated contrast stretch on an image (between min/max intensities)
17. perform a modified contrast stretch on an image (prompt user to enter percentages of dark and light pixels to ignore)
18. perform histogram equalization on an image
19. perform histogram equalization with clipping on an image (prompt user to enter the clipping threshold, as a percentage of the total number of pixels)
20. some other point process of your choosing

Apply the selected image processing action to the image window that currently has focus. Your routines should function correctly with color and grayscale images.

Program usage: *pal file1 file2 . . .*

When your program begins execution, open all image files supplied as (optional) command-line arguments. This should happen without any coding effort on your part. Operations 1-4 are also provided automatically by the QImageLib menu and toolbar.

Notes

- Do not use any predefined QImageLib functions to perform image modification in your program! You may use QImageLib functionality to access image data (and display a histogram), but code the image processing routines yourself.
- Work in teams of two on this assignment.
- When you are finished writing, testing, and debugging your program, turn it in using the MCS website submit program. Submit only your source code, not your executable file or test data. Splitting up your source code into multiple files for separate compilation is highly recommended. At a minimum, submit a zip or tar archive that includes the Qt project file (pa1.pro), the interface file (pa1.h), and the implementation files (pa1a.cpp, pa1b.cpp, etc.).
- The MCS website submit program is accessed via the MCS Department Web page (<http://www.mcs.sdsmt.edu>), by selecting the list item on the left entitled “Submit it!”. Usage is self-explanatory: enter your name, choose the instructor and click “Select Instructor”, choose the appropriate course (CSC442 or CSC542), type in (or browse to) the filename you wish to submit, and click “Upload”. Submissions will be date- and time-stamped. You must submit your program by the due date (Friday February 6) in order to receive credit for this assignment. Late programs will not be accepted for partial credit unless prior arrangements have been made with the instructor. If you have any problems with the submit program, report them to your instructor and submit your program by email instead.
- To receive full credit, your code must be readable, modular, and well-documented, as well as correct. Try to make it reasonably efficient in terms of both execution time and space utilization. Your program must compile successfully under g++ with QImageLib on Linux (Fedora) and Windows (using MinGW, *not* MS Visual C++). If your program does not run correctly, indicate why. This will make it easier to give you partial credit.
- You may use sample code given in class or found on the course website as a basis for your program, but be sure to give credit for any code that you did not write. (As a general rule, *always* credit the author of any code that you use.)