Assignment 1 Report

Harsh Vora
CWID : A20445400
CS512 – F20

October 08, 2020

1. Problem Statement

- In this assignment we had to process the image, that image should be read from a file if a file name is specified in the command line or should capture it from the camera if a filename is not specified in the command line.
- The image should be read as a 3-channel color image.
- Special keys on the keyboard should be used to modify the display image as follows:
- 'i': Reload the original image.
- 'w': Save the current image into the file 'out.jpg'.
- 'g': Convert the image to grayscale using the OpenCV conversion function.
- 'G': Convert the image to grayscale using your implementation of conversion function.
- 'c': Cycle through the color channels of the image showing a different channel every time the key is pressed.
- 's': Convert the image to grayscale and smooth it using the OpenCV function. Use a track bar to control the amount of smoothing.
- 'S': Convert the image to grayscale and smooth it using your function which should perform convolution with a suitable filter. Use a track bar to control the amount of smoothing.
- 'd': Downsample the image by a factor of 2 without smoothing.
- 'D': Downsample the image by a factor of 2 with smoothing.
- 'x': Convert the image grayscale and perform convolution with an x derivative filter. Normalize the obtained values to the range [0,255].
- 'y': Convert the image to grayscale and perform convolution with a y derivative filter. Normalize the obtained values to the range [0,255].
- 'm': Show the magnitude of the gradient normalized to the range [0,255]. The gradient is computed based on the x and y derivatives of the image.
- 'p': Convert the image to grayscale and plot the gradient vectors of the image every N pixel and let the plotted gradient vectors have a length of K. Use a track bar to control N. Plot the vectors as short line segments of length K.
- 'r': Convert the image to grayscale and rotate it using an angle of theta degrees. Use a track bar to control the rotation angle.
- 'h': Display a short description of the program, its command line arguments, and the keys it supports.

2. Proposed Solution

- When the program starts it will first check the argument passed with the program if the image name is present in the argument then it will take that image or else it will automatically start the camera and capture the image.
- If the image is 1 channel only (it is gray) it is converted to 3 channel image.
- If the image is too big to fit the screen it is reduced until it fits it.
- Special keys functions:

The image is converted to gray using openCV function cvtColor()

The image is converted to gray converting each pixel to gray from the bgr values using the formula: g = 0.299*b + 0.587*g + 0.114*r

To cycle through the channels bgr each time the key is pressed out of the three two of the channels value is changed.

The image is converted to gray using openCV function cvtColor() and track bar that controls a smoothing filter is created in the window to control the amount of smoothing.

I convolved the image with gaussian filter by giving hyperparameter values to it and and track bar that controls a smoothing filter is created in the window to control the amount of smoothing.

To downsample the image without smoothing its dimensions are divided by 2.

To downsample the image with smoothing the openCV function is used and its dimension are divided by 2.

The image is converted to gray using openCV function as before then a Sobel filter of size 5 is applied to image only in x direction. Finally, it is normalized.

The image is converted to gray using openCV function as before then a Sobel filter of size 5 is applied to image only in y direction. Finally, it is normalized.

Both Sobel filters in x and y directions are applied to calculate the magnitude doing the square root of the sum of them to the power of 2. The magnitude is normalized afterwards.

Image is converted to gray and Sobel filter in both directions is applied. For every N pixel the angle of its gradient is calculated as the inverse of the tangent of Sobely divided by Sobel x. Then knowing the angle, both x and y components can be calculated, and the arrow drawn.

Image is converted to gray and rotate through the track bar using OpenCV functions getRotationMatrix2D() and warpAffine().

Information about keys' functionally is printed on terminal.

3. Implementation Details

- I was having trouble getting light while taking capture of the image, so I make it record for 15 times and get the last one.
- When use waitKey(0), it does not closes the window manually. It will require reopen the CMD and recall the python.
- Use return to break from multiple loops.
- I first created the window and then add the image to it.
- While calling the SlideHandler() function, the createTrackbar() method of cv2is called again and again that change the filter kernel and render the picture with new filter.

4. Results and Discussion

Below are the results that I obtained while performing the program:

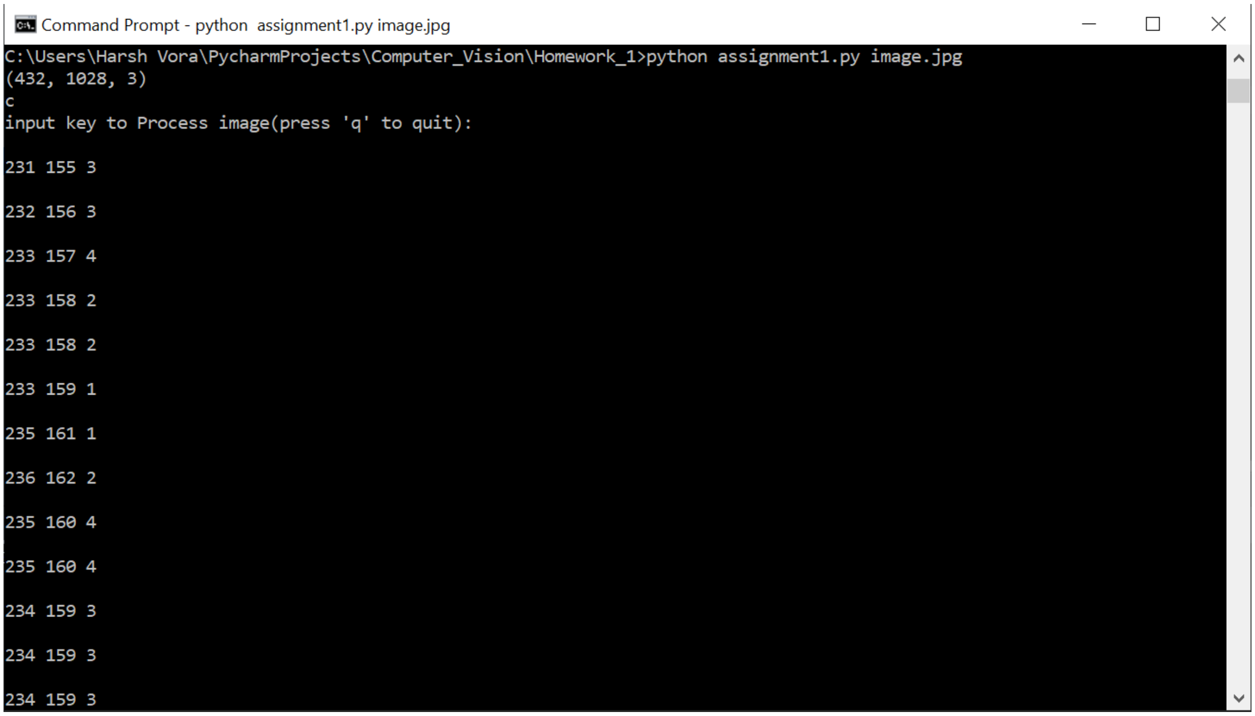
- This is the original image and it is the one that is reloaded when ‘i’ key is pressed.

- All images used in this document have been exported from the program using the key ‘w’ that is the one that saves the image to a file.

- When ‘g’ key is pressed the image is converted to gray using the OpenCV function.

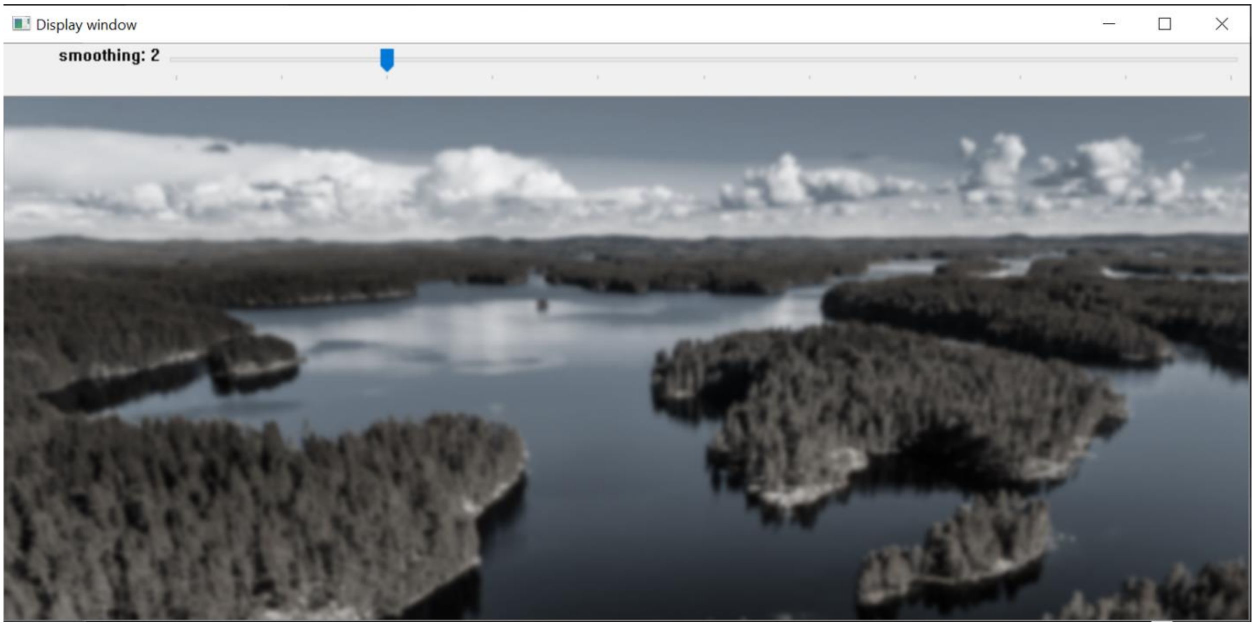
- When ‘G’ is pressed the same conversion to gray is done but using my own code instead of OpenCV function. The result is indistinguishable from the previous one.

- When ‘c’ is pressed several times the image displays the 3 channel colors bgr.



- When ‘s’ is pressed image is converted to gray and smooth is controlled with the track bar.

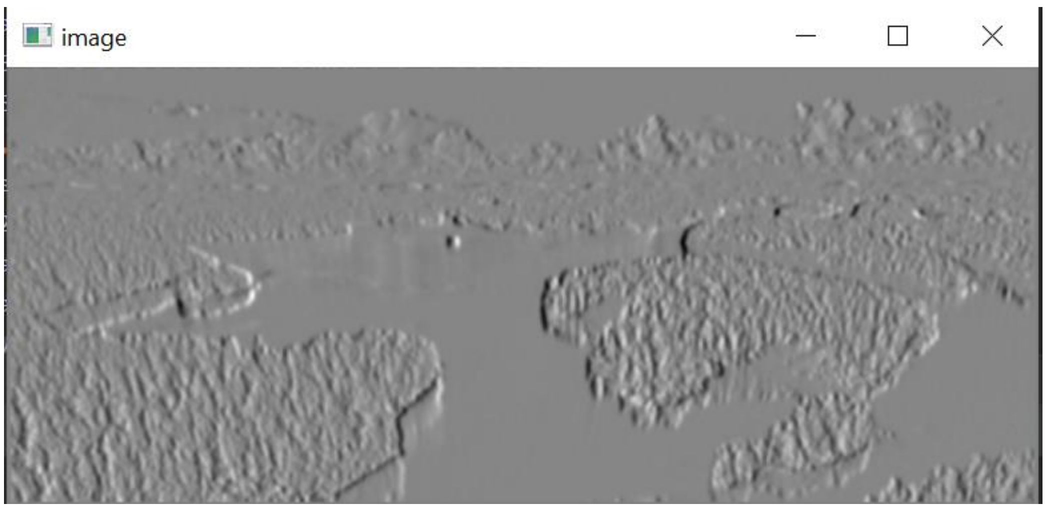
- When ‘S’ is pressed it was supposed to do the same as the previous one but I could not get it done.



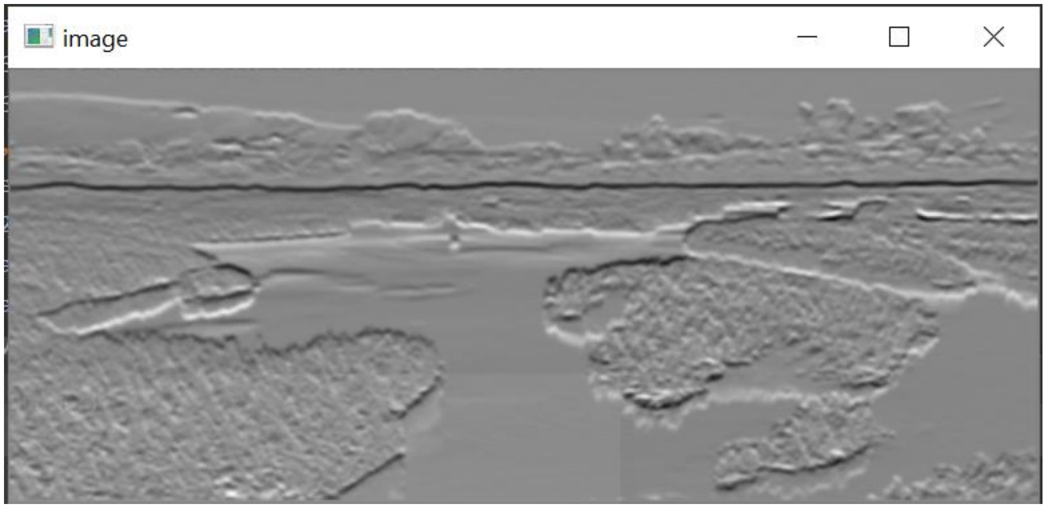
- When ‘d’ is pressed the image is downsampled by 2 without smoothing. Since I am reducing the size of the image I cannot show that it is downsample by 2.

- When ‘D’ is pressed the image is downsampled by 2 with smoothing. Again I cannot show the downsample but the smoothing can be seen in the image compared to the previous one.

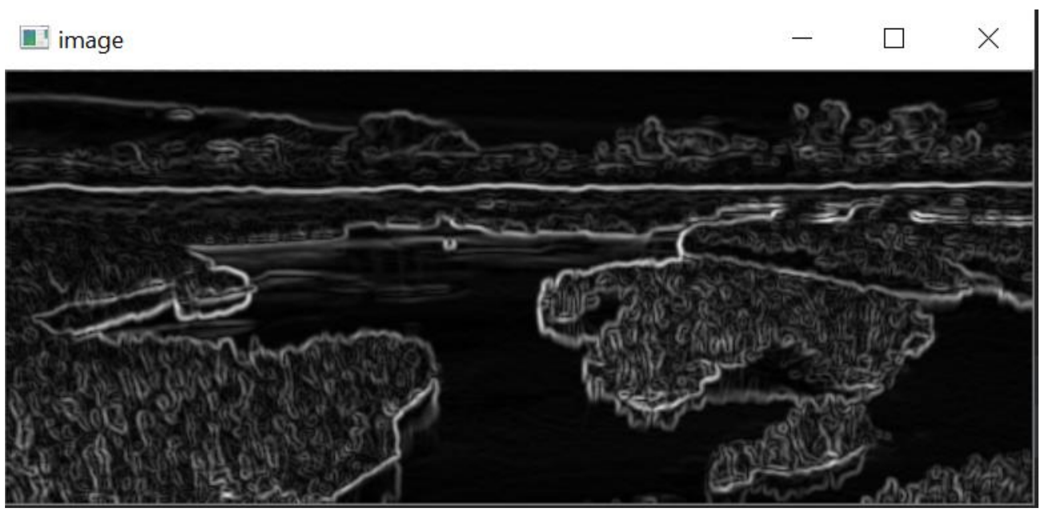
- When ‘x’ is pressed the image is converted to gray and convolve with an x derivative filter (Sobel).



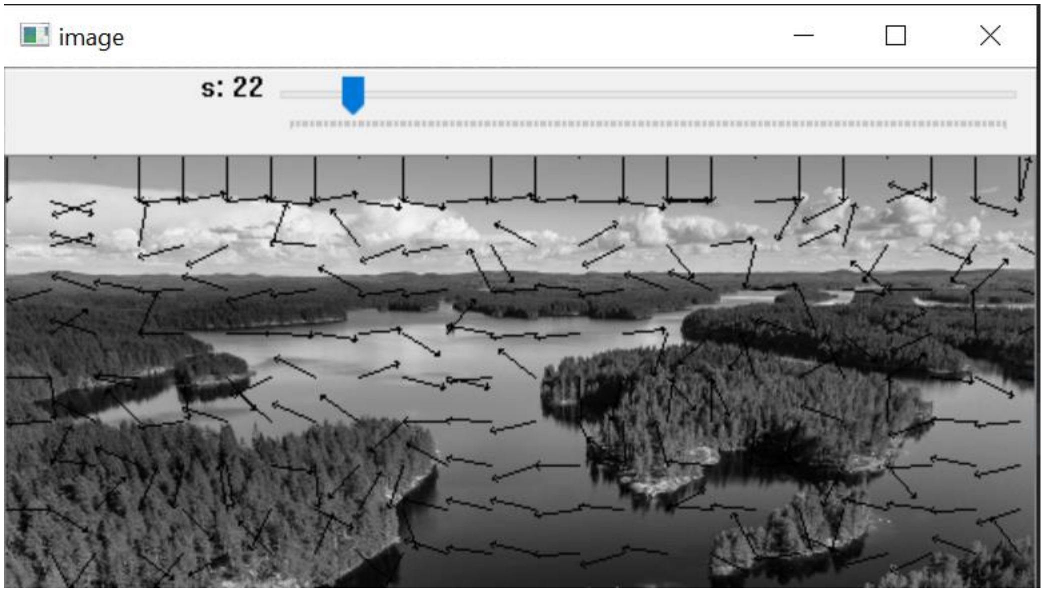
- When ‘y’ is pressed the image is converted to gray and convolve with an y derivative filter (Sobel).



- ‘m’: show the magnitude of the gradient normalized to the range [0,255]. The gradient is computed based on the x and y derivatives of the image.



- ‘p’: convert the image to grayscale and plot the gradient vectors of the image every N pixel and let the plotted gradient vectors have a length of K. Use a track bar to control N. Plot the vectors as short line segments of length K.



- ‘r’: convert the image to grayscale and rotate it using an angle of theta degrees. Use a track bar to control the rotation angle.