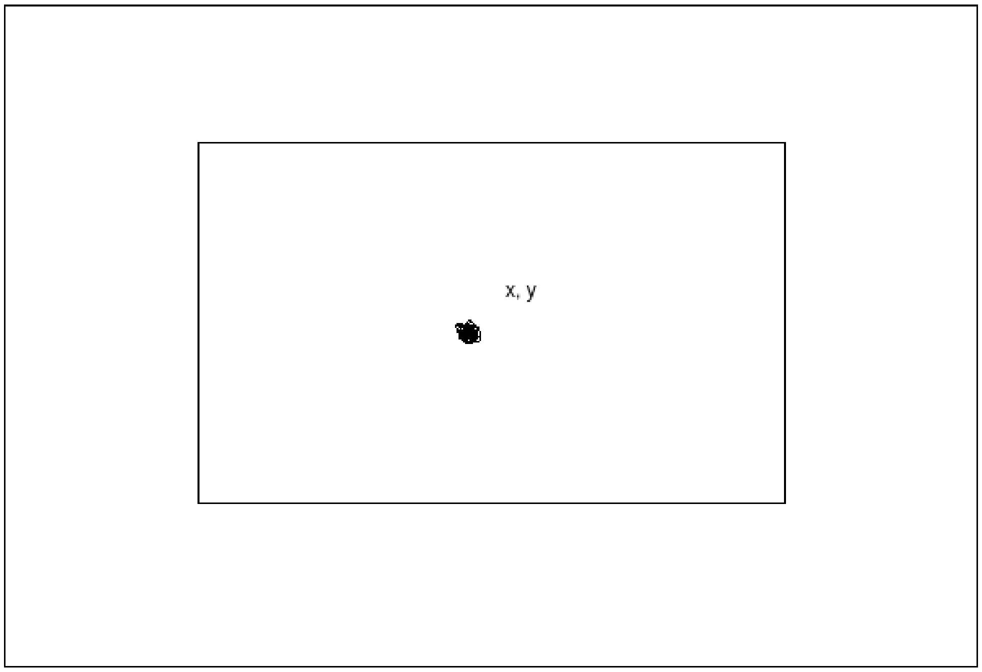
vusal ismayilov - problem set 1

problem 1)

(a)

in the first part of the assignment, the center points of the frame (x,y) were found. after which, the height and the width were divided by 4. the founded value by using the center points' coordinates, adding and subtracting to those values, new rectangle, which is exactly 50% in width and the height of the frame as 25% up and 25% down coordinates were taken on both height and width.



b)

there were 3 channels in the image i have chosen. rgb (red, green, blue). to consider that, then it is obvious that in the list [0] should have been used, however, opencv reads the images in bgr (blue, green, red) whereas red is the last element in the list. either [2] or [-1] could be used to extract the red filter of the image/frame. in the output frame which points are brighter compared to the other pixels/points, then that means those are the parts where red has more exposure and domination accordingly. if other channels would liked to be extracted, the same process and the logic would be applied also.

c)

unique function of opencv converting the image to grayscale exists. however, it is also possible to be done manually by decreasing the dimension of the image to 1, which will cause the color range to change only between 0-255 (white-black).

d)

firstly, using opencv's built-in grayscale converting function, the given image is being translated to grayscale image. then, the magnitude function was used within the threshold as an argument, providing 4 unique colors to represent the agile lines in different degrees accordingly. in this particular case, 360 degrees was divided into 4 90 degree parts, and each 90 representing its own according color.

e)

at this point, 3 different kernels (custom) were created and applied in the filter2D function. each kernel while looking similar to each other, however, resulted in almost completely different results.

problem 2)

a)

while having multiple cameras connected to the device, looping through the range starting from -1 to to the positive infinity, the correct camera was found and used within the code. usually, -1 and 0 representing the same camera was also the case here. “q” pressing also breaking the loop condition was added. while taking the live cam stream from the device, also another frame copying that frame, converting to gray scale and applying the same gradient magnitude logic done in problem 1 was added on top. so, 2 different video streams are running at the same time, both being the same, but one the gradient magnitude of the first one.

b)

here, unfortunately, concatenating the gradient magnitude and the usual stream was failed in the first tries. the reason being the magnitude frames were grayscale, they had no dimension besides the usual stream, which is rgb, hence, having 3 dimensions. however, as the gradient magnitude algorithm is based on the grayscale images, after having the algorithm done completely, it was converted to rgb instantly. while converting it to rgb, concatenating both of the frames to each other was succeeded. in order to save the recording, .avi method was used. .mp4 extension video type could also be an option, however, .avi is more friendly to the user in terms of missing the last frames do not usually break the whole video up, but those parts are just being missing besides .mp4, which completely corrupts your video even if 1 frame is missed. for this case, .avi extension video type was used in action. another 15 seconds of recording was also added as a condition, so whenever the code is being executed, the first 15 seconds starting from the exact second the code has been launched, it is being recorded at the same time and also saved automatically to the specified /output folder, which is being told in the description of the assignment. however, only ending the recording condition was used for that condition, which means, even 15 second ends, video records successfully, and also saves it, whole process in the background of fetching the frames from the cam as long as ‘q’ is being pressed continues.