HACETTEPE UNIVERSITY

Department of Computer Engineering

FALL 2018

BBM 415
Image Processing Laboratory

Name and Surname:	Ege Berke Balseven
Number:	21590776
Subject:	Photoshop Battle
Due Date:	02.11.2018

Contents

1 Introduction

This assignment includes 2 parts. At part one i implemented some basic arithmetic operations on the images. I use given data sets which have foreground object masks and original images which includes foregrounds. Firstly, i separate the foreground objects by using the given masks and then i put it on a different background that i decided images to change the background of the foreground object. I set the position of the foreground objects by changing the location of the pixels in order to put it on a meaningful place.
In part two i apply Gaussian filter to the images for blurring and also for sharpening. If you subtract the blur image to original image you obtain edges. Then add that edges with the original image and you obtain sharp image. With the Gaussian kernel i convolve the images and i apply it each channel of the color image. I used built-in function for the convolution but i used my own Gaussian filter function. I add some depth to image by blurring the background and sharpening the foreground. Finally i created the portrait mode images.

2 Implementation Details

At first i was try numpy.bitwise_and and numpy.bitwise_not but after that i realize i was in a wrong way. Then i investigate and found np.where. With np.where i extract the black pixels(0) of the mask and puts the original image to non blacks pixels of mask. After i do that if the pixel is black pass, if not put the original image to background image that i decide. That is how: mask[i,j].all() == 1 is getting the white(1) pixels, then bg[i + rowshift, j + columnshift] = foreground[i, j] do that shift foreground and put to background. Then in second part at first i can only blur the grayscale image and i was not found how to blur all image channels. Also i had darkened image to after applying gaussian filter. I handle it all with doing some formula changes, image normalization, bgr to rgb. I did convolution with gausIm[:, :, i] = signal.convolve(bg[:, :, i], gausskernel, mode='same') for each image channel.

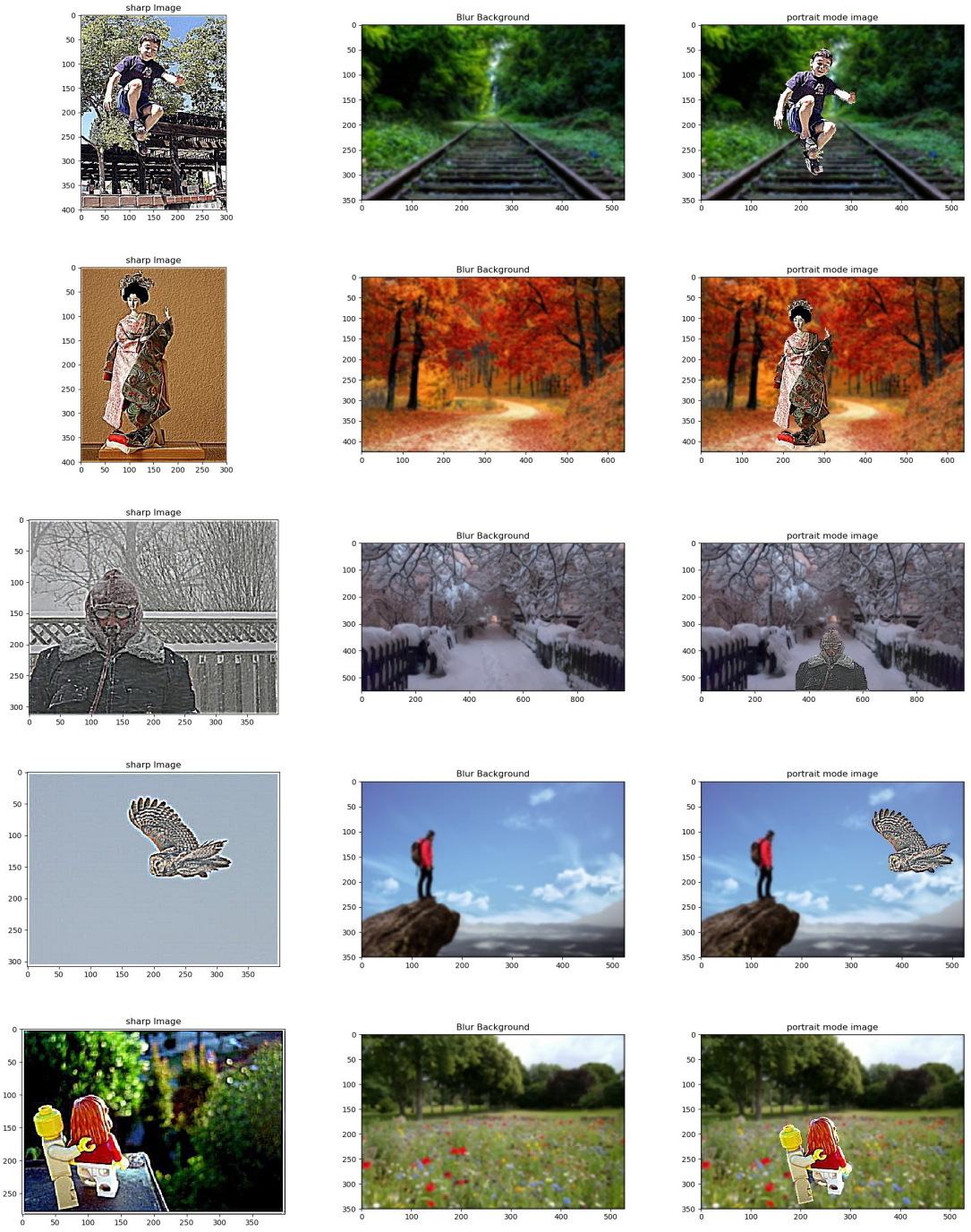
3 Experimental Results

3.1 Part 1: Image Masking and Changing Background

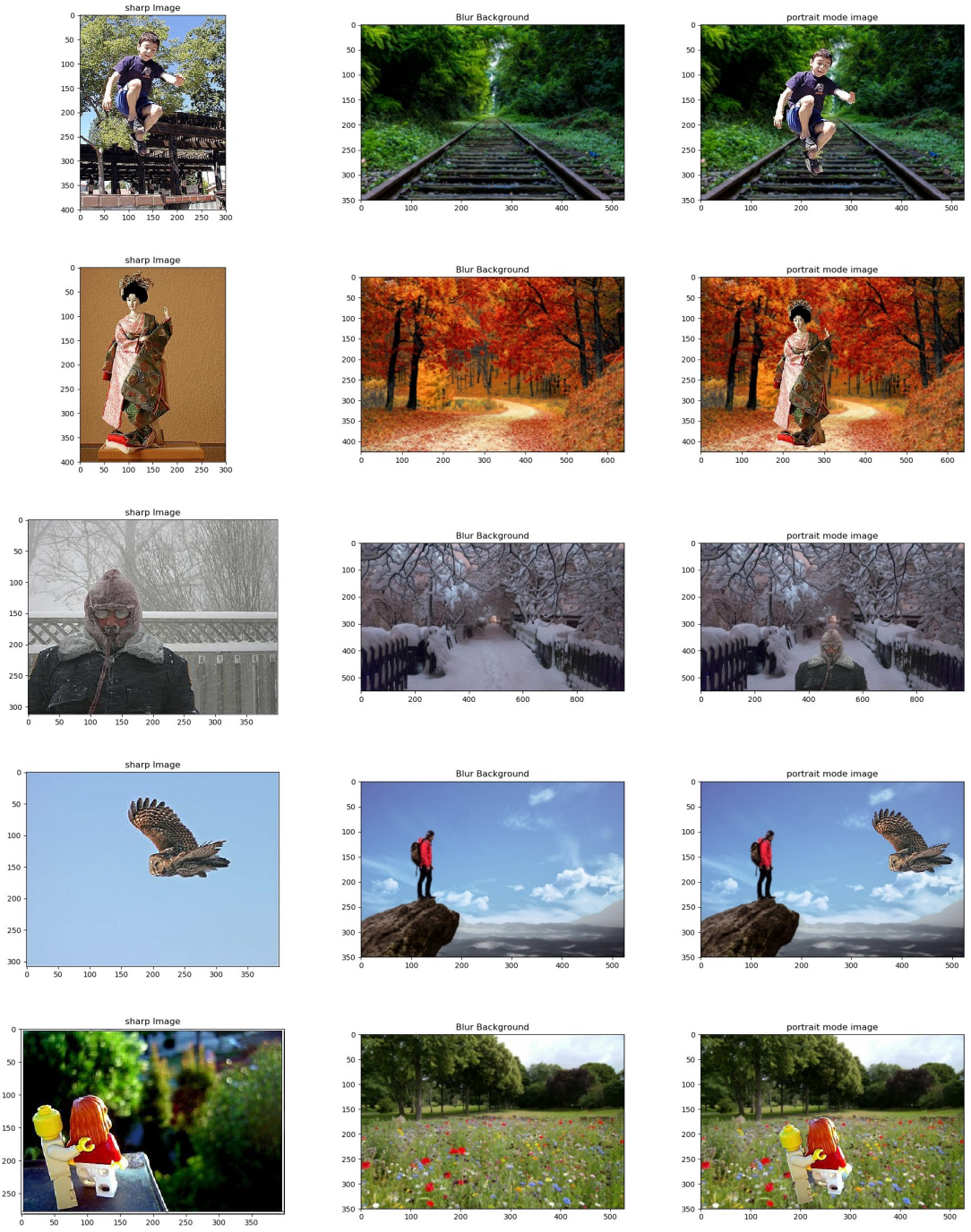
Here are the original images, masks, foreground images and foreground images with a new backgrounds that i chosen:

3.2 Part 2: Background Blurring and Sharpening Foreground

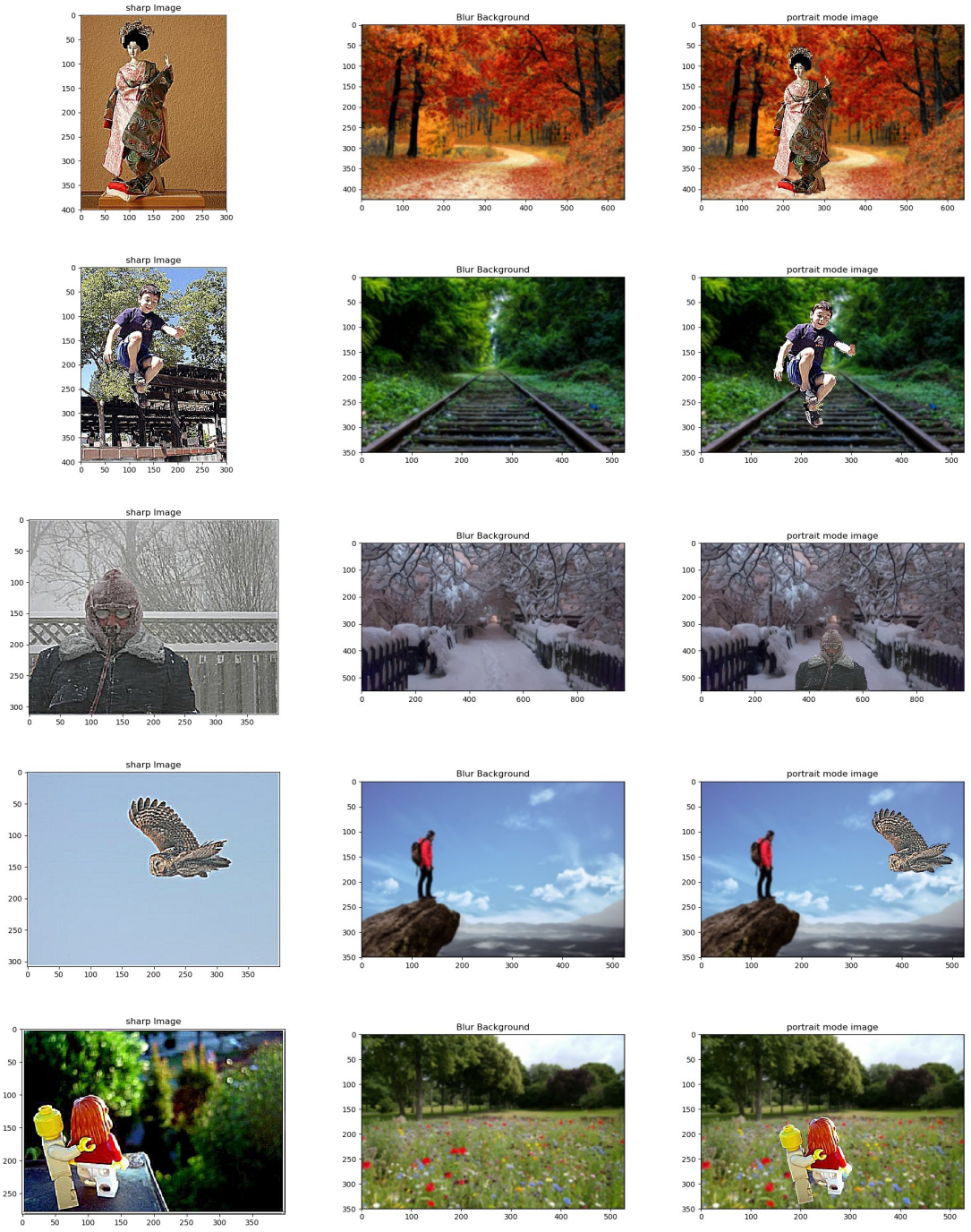
Here are the portrait mode images with according to sigma, alpha values and kernel sizes:



gauss kernel size: 7, sigma: 7, alpha: 4
kernel size, sigma and alpha are very large so background and foreground blur difference is too much and realism gone



gauss kernel size: 3, sigma: 3, alpha: 2
This is the most ideal portrait mode, blur difference less so realism more

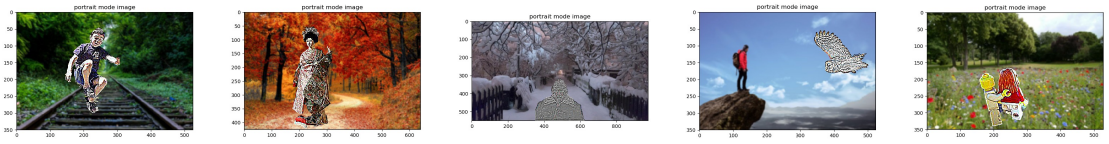


gauss kernel size: 5, sigma: 5, alpha: 3
the results are not bad but the blur difference is a bit too much

Some other results:

gauss kernel size: 2, sigma: 10, alpha: 2

gauss kernel size: 10, sigma: 2, alpha: 2
High kernel size affects blur and sharpness more than high sigma



gauss kernel size: 5, sigma: 5, alpha: 10

gauss kernel size: 5, sigma: 5, alpha: 1
Alpha only affects foreground sharpness. When Alpha is low, sharpness decreases, but when it is too high, the sharpness is not high, the image becomes distorted.

4 Conclusion

The results are successful but unfortunately not like the real photoshop app. I could get better results with things like edge smoothing and doing histogram equalization etc. Although Part1 was not very difficult, I thought I was doing the right research and I got the wrong result. In Part2 i thought long time about how to implement the gassian filter and i have encountered problems with color channels. The process may not seem very difficult, but it wants deep research. In the future, i try to get a

better result by learning better with the right research.