|  |  |
| --- | --- |
| Name: Hiteshi Shah (hss7374) | Foundations of Computer Vision  Due: 02/16/2018 |

**HW05**

1. With the Local\_Smear\_Routine( im ) routine, the resulting image looks a lot smoother

than the original, particularly, the dark parts of the image are less harsh like Chelsea’s hair

and the dark patterns on shirts.



As we can see, the smoothing effect caused a drastic decrease in the number of bright pixels.

2. The results of Local\_weighting\_routine( im, weights ), with the [1 2 1 ; 2 4 2 ; 1 2 1] / 16

filter are almost identical to the previous filter, which a few more darker pixels.



3. The results of Local\_weighting\_routine( im, weights ), with the fspecial(‘Gauss’, 5, 1)

filter are almost identical to the previous filter, with a few more darker pixels and is a little more blurred than the other two images.



4. With the Local\_weighting\_routine( im, weights ), with the fspecial(‘Gauss’, 5, 1)

filter, the resulting image looks a lot smoother than the original as well as the results of the other three filters.



As we can see, this smoothing effect caused a drastic decrease in the number of bright pixels, and increased the number of dark pixels. This routine also took the longest to execute among all the routines at about 5.75 seconds.

5. One issue that I ran into while writing the functions, we maintaining the sum of the pixel

values. Initially the variable was of type uint8, so it’s value never went over 255 which resulted in a very dark image.

I ignored smoothing the borders for this assignment.

6. The goal of this assignment was to demonstrate how the smoothing filter works and to guide

us to be able to write our own filters. Through this assignment, I learned that smoothing causes a decrease in the number of bright pixels, and an increase the number of dark pixels. I also learned that the mod() function in Matlab returns the remainder of a divison and the fix() function returns the quotient.