COMP 546 Assignment 1

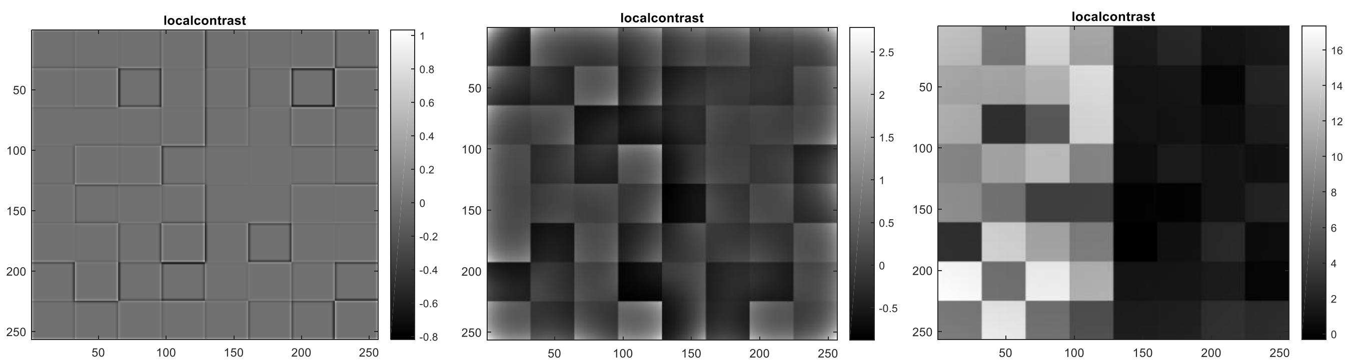
260674503

seong.an@mail.mcgill.ca

Christopher An

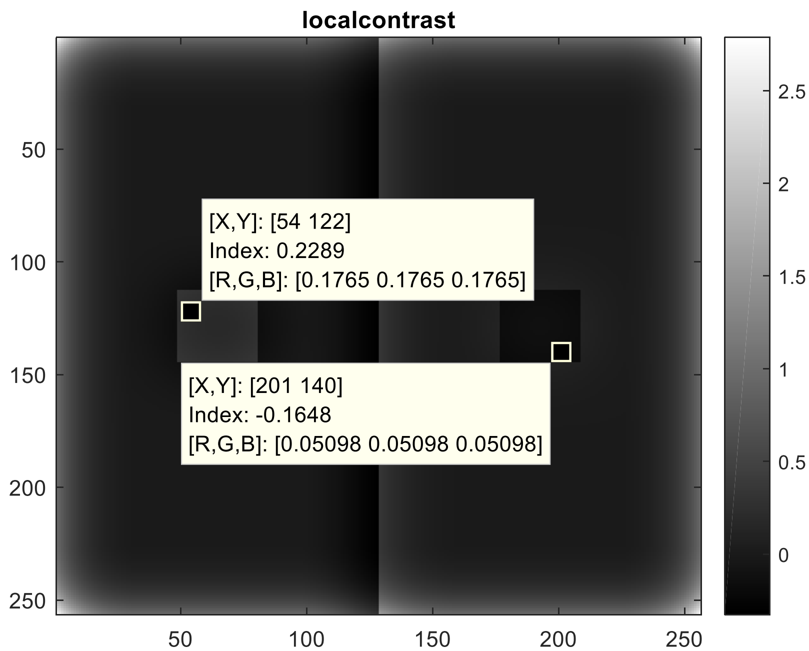
1. Local Contrast

a. After testing multiple sigma values in MATLAB, I found out that the sigma value of around 15 is optimal for comparing the original values of the unshadowed verses shadowed squares. Mathematically speaking, this value makes perfect sense. The function *makeImageCheckerboard* generates a 256 pixel by 256 pixel square and divides the whole square into 64 equal squares. In order to be able to compare the original values of the image well even after applying the shadow, one must apply a gaussian filter with the width being approximately the size of each smaller square. This filter would then reduce the effect of the shadow and amplify the original colours. Heuristically, width of the gaussian can be estimated with 2σ, which would indicate that the optimal width of the gaussian for this image would be 32, in turn implying that the most optimal σ value should be around 16. If the sigma value becomes too small, you will not be able to compare the original values of each square properly. On the other hand, if the sigma value becomes too large, gaussian filter becomes meaningless as the effect of the shadow will be displayed without any filtering.



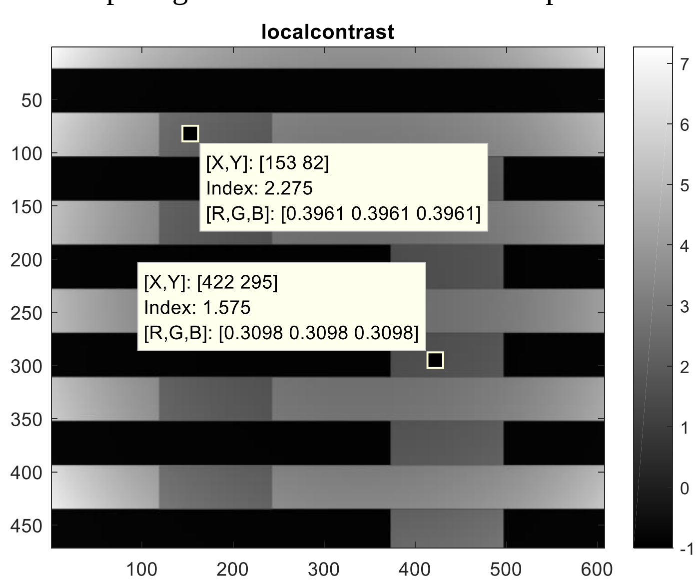
From left to right: sigma = 1, sigma = 15, sigma = 256 (you can see the effect of sigma values well)

b. Yes, local contrast can explain the perceived brightness.



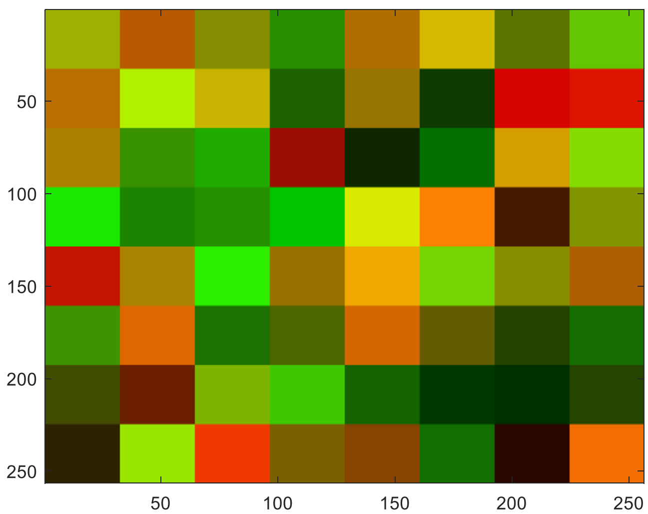
Using the same sigma value as question a (15), computing the [R,G,B] values of each squares show that the square on the right has [R,G,B] values closer to [0,0,0]. This implies that the square on the right side is darker than the square on the left side. If you compute the [R,G,B] values of these squares on the original image, they actually display exact same values. We can explain this illusion using local contrast because computing local contrast of a grey square on a darker background makes the local contrast of square higher (becomes brighter to our eye), while computing local contrast of a grey square on a lighter background makes the local contrast of a square lower (becomes darker to our eye).

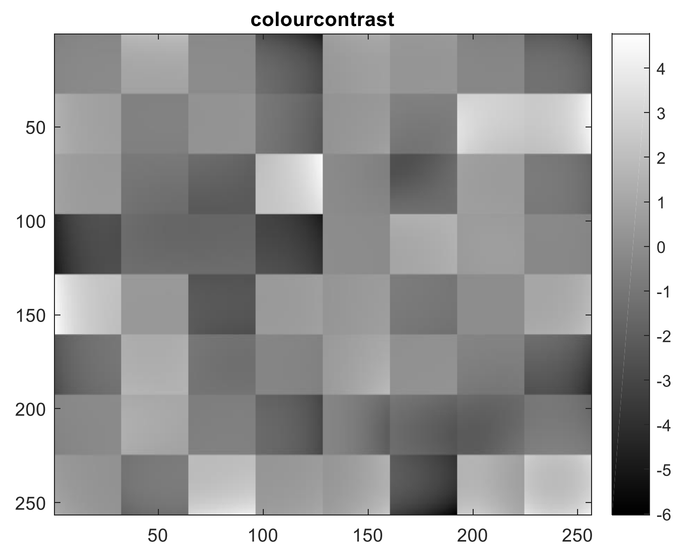
c. Computing the local contrast cannot explain White's illusion.



Grey rectangles on the left has have greater [R,G,B] values than grey rectangles on the right, implying that the grey rectangles on the left should look lighter. However, our eye views the opposite – grey rectangles on the left looks darker than grey rectangles on the right. Therefore, we cannot use local contrast to explain White's illusion.

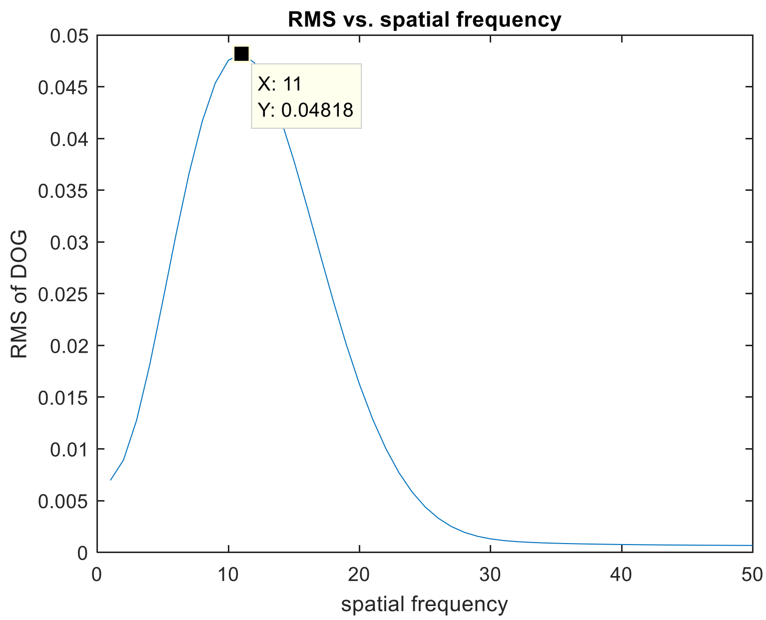
d. The goal of normalizing the R-G opponent values is to make the colour difference much more obvious. This normalization would allow one to ignore the intensity difference and only compare the original colour values. Dividing *singleOpponency* by *intensityLocalMean* would have this effect on the images as this would standardize the RGB values on a new scale, ignoring intensity values.



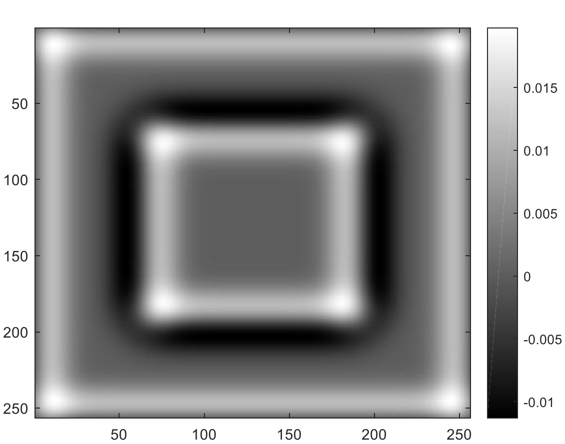


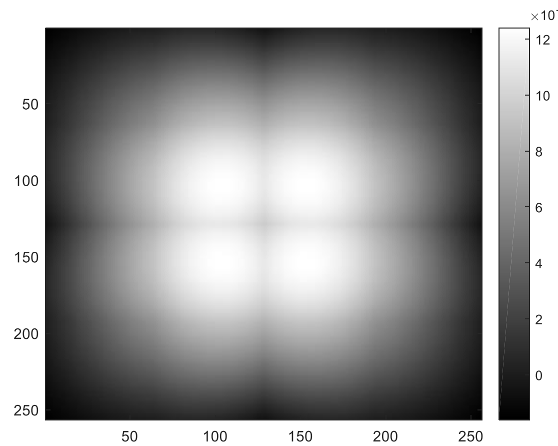
2. **Gaussian and DOG filtering of spatial patterns**

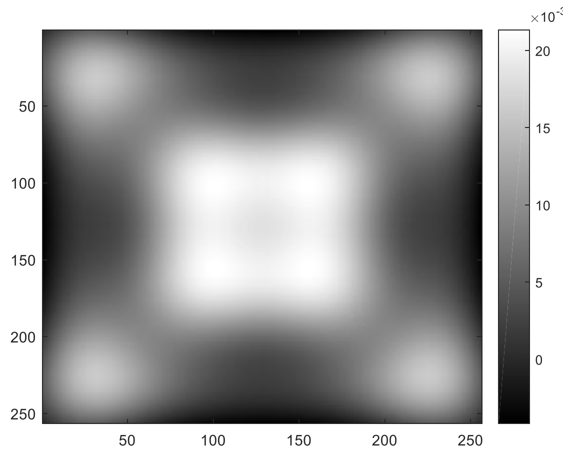
   a. When I used the values sigma_1 = 5, sigma_2 = 5.5, N = 256, KX = 1:50, and KY = 0, maximum RMS output was obtained when spatial frequency (KX) was 11. Mathematically speaking, the maximum RMS value would be obtained if the peak of the gaussian matches with the peak of the cosine curve (when cosine is at a positive value). This would indicate that the wavelength of the cosine curve should be approximately equal to the width of the gaussian (which can be estimated heuristically with 2σ).



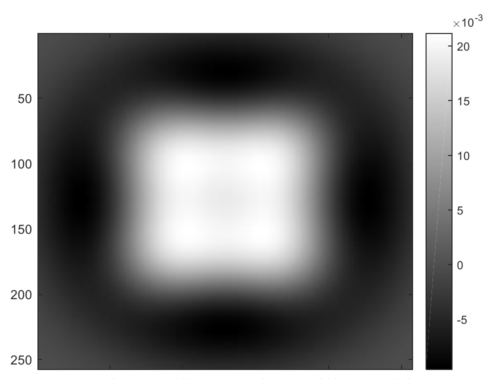
   b. First two pictures show the filter outputs on the image when the square is much smaller or larger than the center region of the DOG, respectively. When the square is much smaller than the center region of the DOG, we see the edges (around the small square and the background) become amplified. This is because the gaussian size is too small to create any effect on a uniformly coloured background, except when there is an obvious edge present between the gray background and the white square. When the square is much larger than the center region of the DOG, you see a large blur around the square region.
   When the square is approximately big as the DOG, you start seeing the image of the inner square more clearly – this might mean that the DOG filter is working to enhance the edges much more clearly.



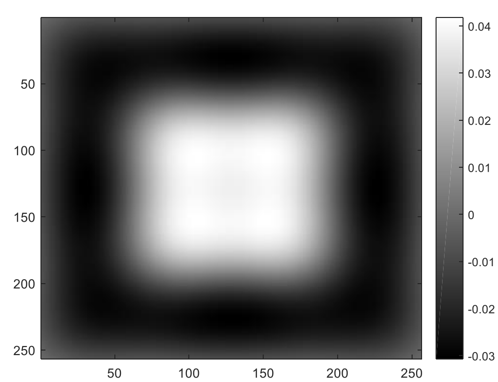




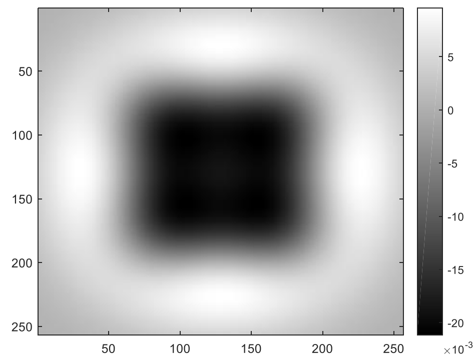
c. Here I set the DOG in the R channel as on-center-off-surround and the DOG in the G channel as off-center-on-surround.



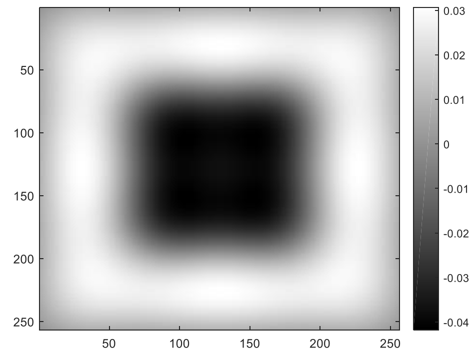
red square on white background



red square on green background



green square on white background



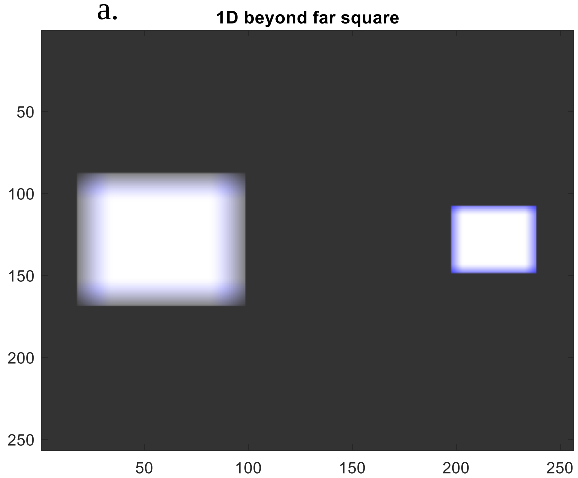
green square on red background

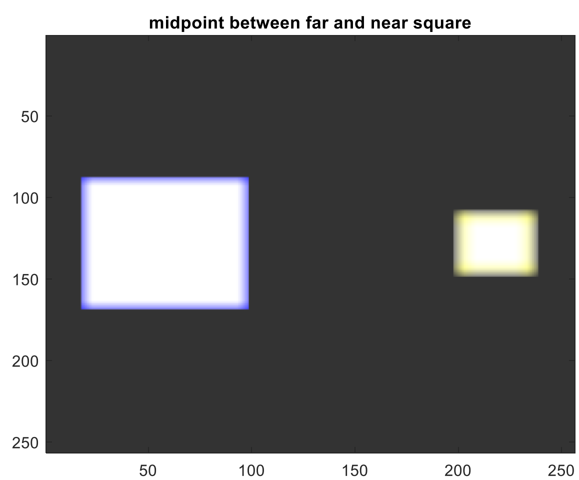
red square on white background

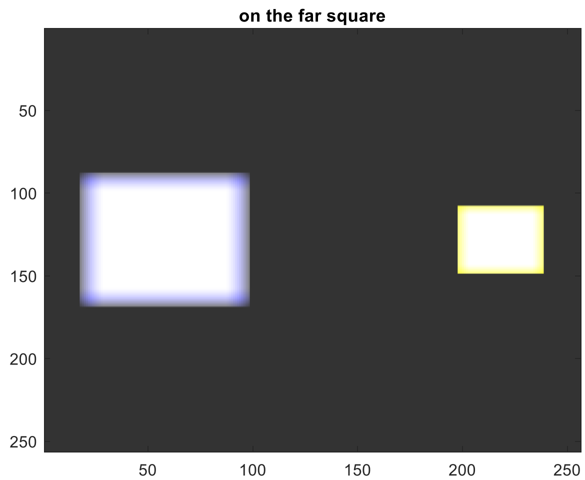
3. Chromatic aberration and depth

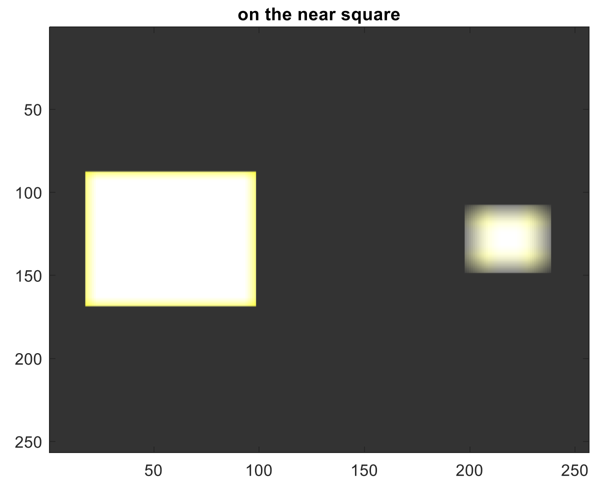
a.

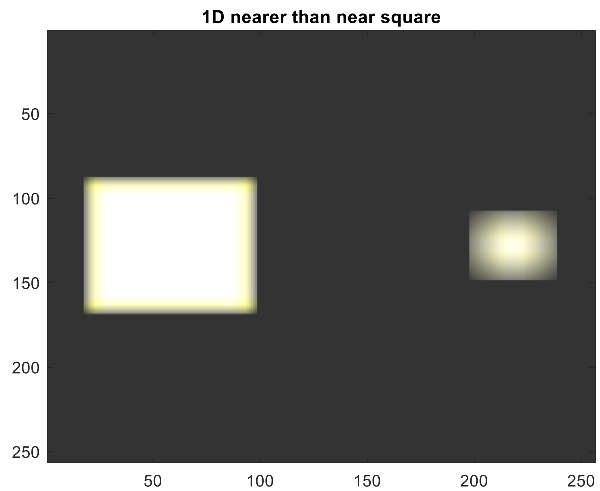
1D beyond far square









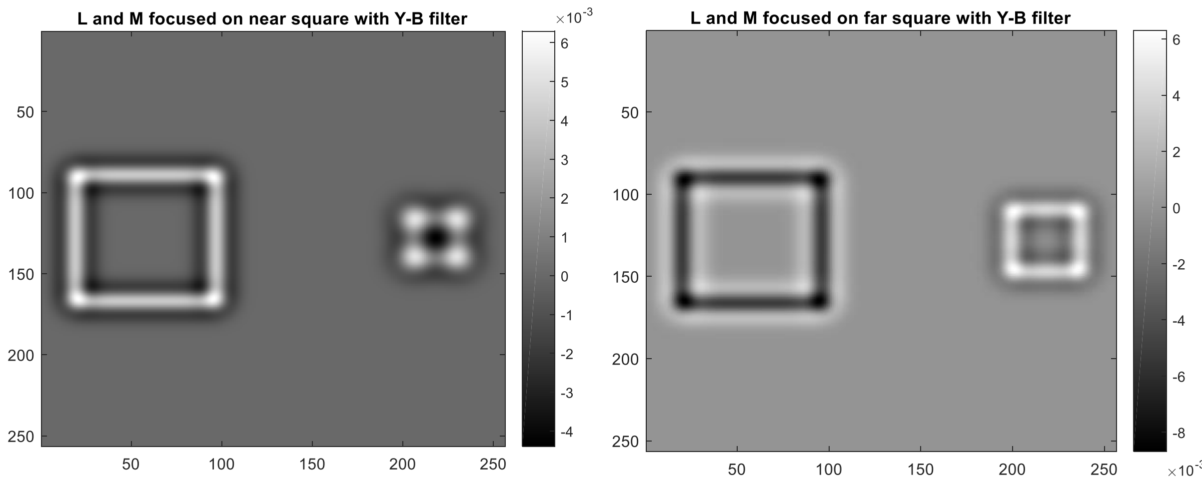


	L,M Large square	S Large Square	L,M Small Square	S Small Square
1D beyond far square	34.36 pixels	22.92 pixels	11.44 pixels	0 pixels
On the far square	22.92 pixels	11.44 pixels	0 pixels	11.44 pixels
At the midpoint	11.44 pixels	0 pixels	11.44 pixels	22.92pixels
On the near square	0 pixels	11.14 pixels	22.92 pixels	34.36 pixels
1D nearer than the near square	11.44 pixels	22.92 pixels	11.44 pixels	45.84 pixels

For when L and M channels are focused on the far square, you get a defined yellow tinge around the small square (far square). This makes sense as only R and G channels are focused on the far square. There is a lot of blue blur on the larger square since none of the channels are currently focused on it. Most of the blue blur comes from the fact that the L and M channels are focused far away from the square, resulting in this type of blue.

For when L and M channels are focused on the near square, you get a defined yellow tinge around the large square (near square). This makes sense as only R and G channels are focused on this square. There is a lot of yellow blur on the far square as all the cones are focused far away from it. The yellow blur comes from the fact that the cone focused the farthest from this square is the S cone.

b.



The double-opponency filter enhances the blur effect. In the case where L and M cones are focused on a near square, yellow tinges are enhanced on both the far square, making the blur more obvious (white colour in the diagram). The blue tinges become darker in the closer square.

Similarly, in the case where L and M cones are focused on far square, yellow tinge on the far square is enhanced (white colour in the diagram). The blue blur on the near square becomes more obvious as well (black colour in the diagram).