Ikhlas Attarwala / Fall 2019
EECS 331 / Computational Photography
Group #9 (w/ Yunhao Li, Deanna Dimonte)

Reducing Noise via Flash/No-Flash Image Pairs

~~~Before I begin, I'm really sorry but the images have very tiny titles.. I couldn't control this…~~~

Okay, this was an interesting assignment. This week we looked at synthesizing new images with higher quality than the pictures that we initially took. Lighting has a powerful effect on the visual richness of an image, but at times we're required to take pictures in darker environments, or those that are exposed to more natural light. While increasing exposure time can cause motion blur and widening the aperture can limit depth of field, additional light sources such as flash can often distort the true colors and ambiance of the image. One unique approach is merging two identical images with different lighting sources to make a new image that combines the qualities of both, for a better image.

First we'll take an image without a flash to capture the environment's true color and atmosphere. Then we'll take a flash image to capture the crisper details of the image that we couldn't do without additional light. It's important to note that illumination of flash increases our SNR (signal-to-noise ratio), thus providing better estimates of high-frequency details. Unfortunately, we couldn't take any pictures with real flashes, so we programmed our android tablet to take one picture with a "flash" where the screen facing the scene is completely white, and a second picture where the screen is completely black. This Android code is the same as my partner's (Deanna Dimonte).

After we followed the correct procedures to capture the images, we processed them in MATLAB by first segmenting the image into different color channels (R, G and B), then applying a bilateral filter to denoise the channels of both the flash image and no-flash image. An example of the differences between our color channels is portrayed by **Fig. 1** below.

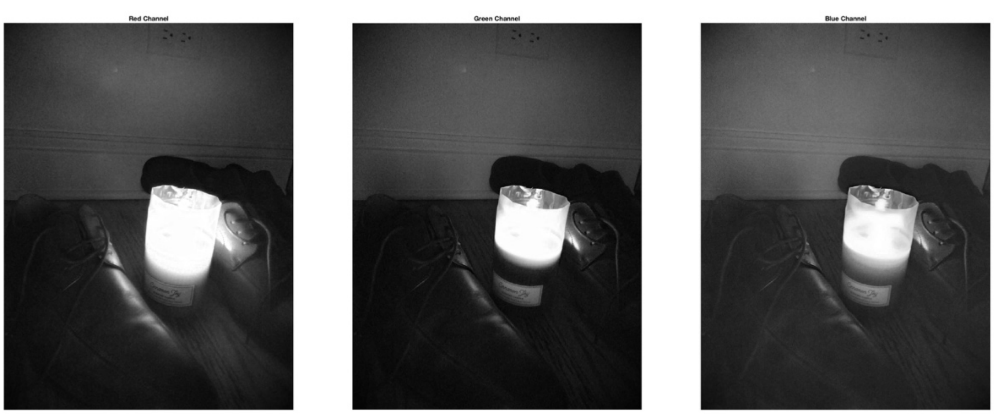

Figure 1: The 3 RGB color channels (Red, Blue, and Green, from left to right)

As mentioned earlier, the flash image is assumed to be a good estimator of high-frequency content. A modified bilateral filter is used because it averages spatially-near pixels that have similar intensity values. As such, this function of denoising an image takes in a spatial (σ_s) domain and intensity (σ_r) domain as parameters to the filter. I experiment with the spatial domain on a range from 1-64, and the intensity domain from 0.05-0.25, in order to see what denoising results are ideal. My results for passing a bilateral filter over my no-flash and flash images are tabulated on the next page in **Fig. 2** and **Fig. 3**.

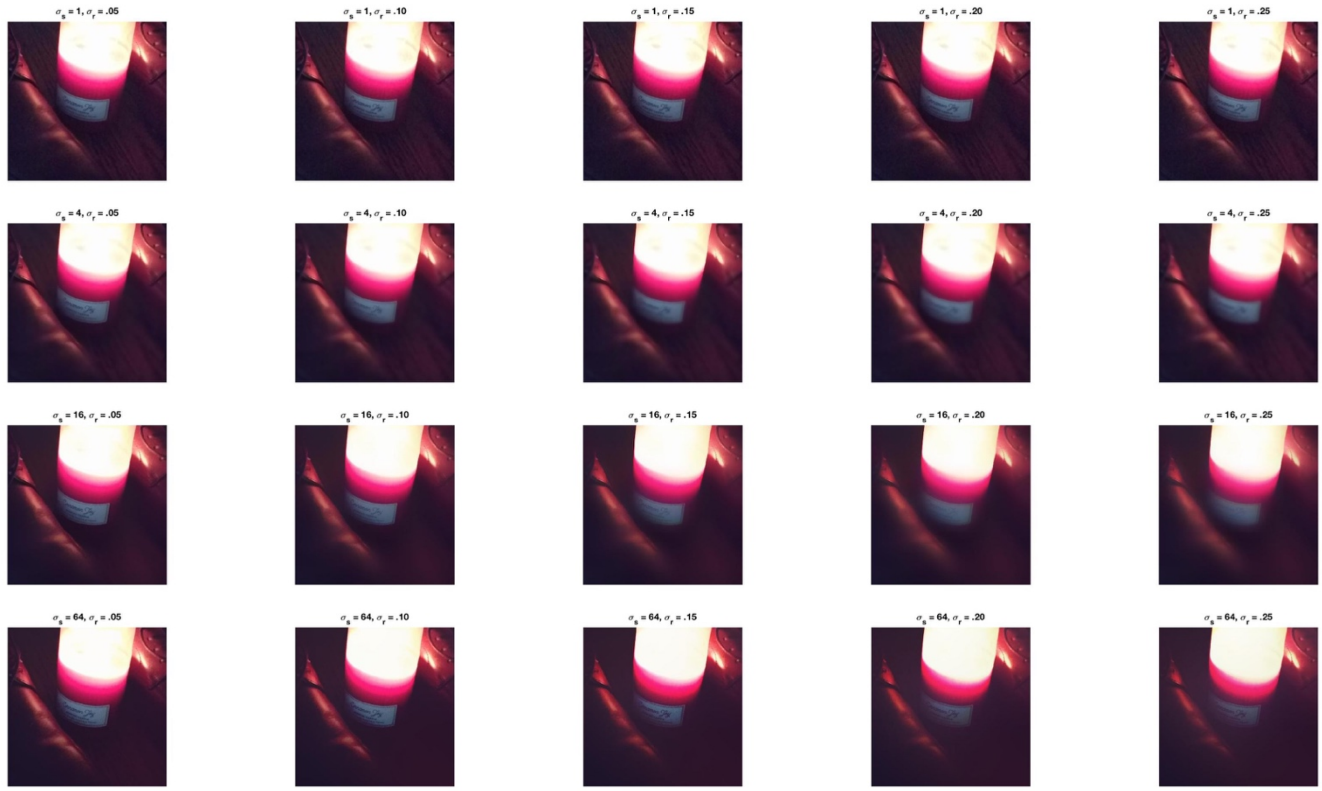

Figure 2: Bilateral filter passed over the **no-flash** image with changing spatial and intensity domains. (.05, .10, .15, .20, .25 σ_r from left to right; 1, 4, 16, 64 σ_s from top to bottom)

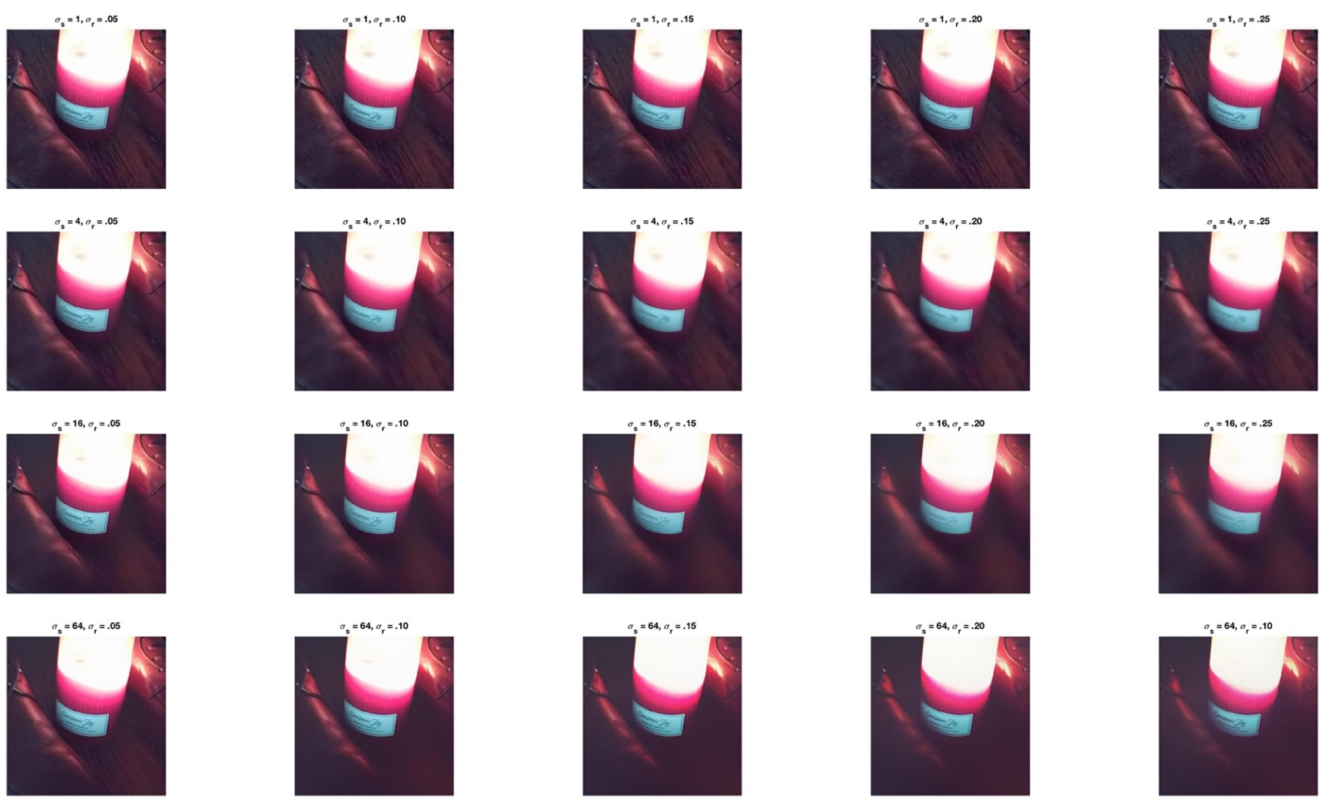

Figure 3: Bilateral filter passed over the **flash** image with changing spatial and intensity domains. (.05, .10, .15, .20, .25 σ_r from left to right; 1, 4, 16, 64 σ_s from top to bottom)

According to my results and observation of all 20 bilaterally filtered versions of the no-flash image, the filter setting of <u>σ_s = 4 and σ_r = .05</u> had the best denoising. I didn't plan for this, even though Ollie's website suggested they would be similar, but the best denoised image from the 20 bilaterally filtered versions of the flash image also had the same filter setting of <u>σ_s = 4 and σ_r = .05</u>. The original images, denoised images, and fused result are arranged below in **Fig. 4**. (I attached the non-cropped version of Figure 4 as **Fig. 8**, at the very end).

The fusing is quite interesting. We want to transfer detail from our denoised flash image to our denoised no-flash image, and this can be done easily because the higher SNR of the flash image retains nuances that were removed by filtering the no-flash image. Filtering essentially preserves strong edges and blurs away most of everything else. We use the formula below to compute the fused image.

$$N_{final} = N_{denoised} * \frac{F_{original} + \epsilon}{F_{denoised} + \epsilon}$$

In the above equation, N represents the no-flash image, F represents the flash image, and ϵ is a value that Ollie, Yunhao and I determined is a small positive value used to keep the denominator from reaching 0. I will discuss testing this epsilon value in further detail at the end of this report.

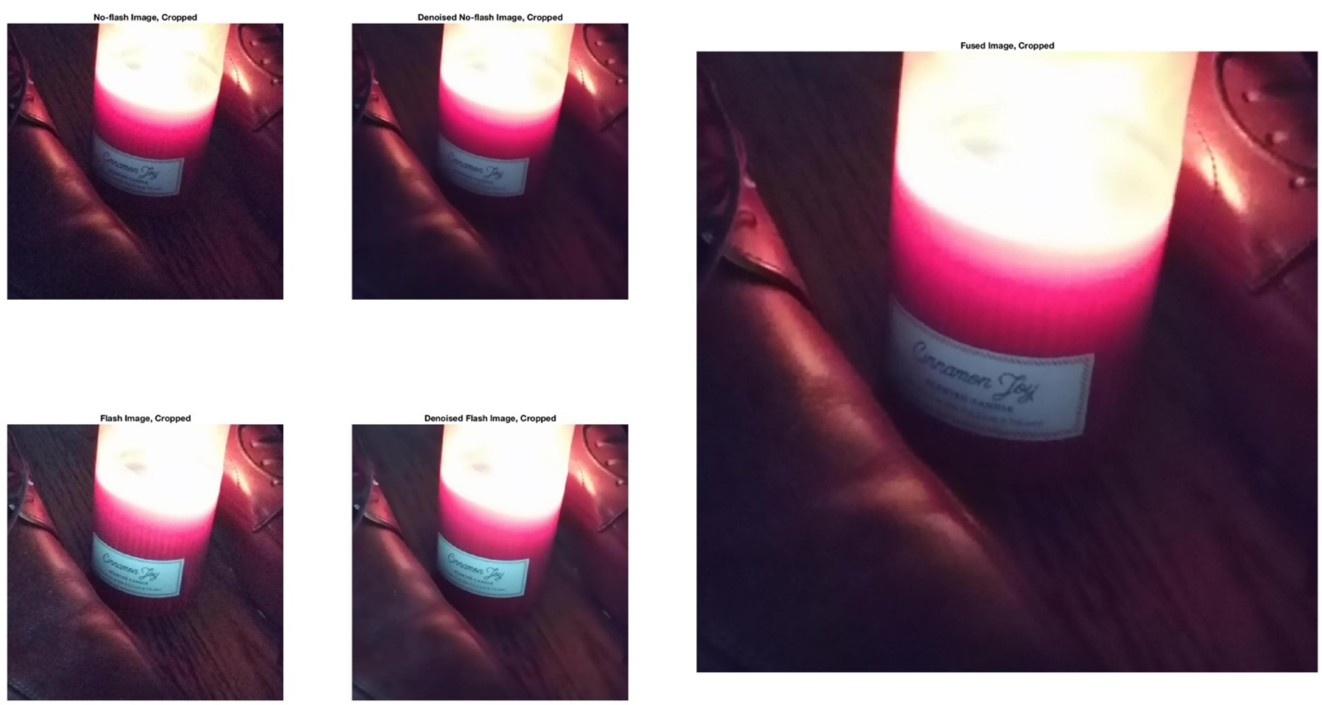

Figure 4: The original images, denoised versions, and fused image (all cropped for observing details)

While a little difficult to tell, all my images are saved in the 'RESULTS' folder of my submission. Above, the denoised no-flash image has reduced noise, but has pretty much wiped out the details of the wood flooring. In the denoised flash image, we still have some of the strong edges like the wood flooring, but the lighting and color is off. The result of the fusion is a version that is denoised, properly lit and colored, and also preserves stronger details. We did it!

Or so I thought. Just to be safe, I decided to fuse every combination of no-flash and flash images with the same spatial and intensity settings (ranging from 1 to 64 for spatial, and .05 to .25 for intensity). **Fig. 5** shows my findings. While it's difficult to tell, I checked each of the 20 images individually and the best two images were #1, $\sigma_s = 4$ and $\sigma_r = .10$, and #2, $\sigma_s = 4$ and $\sigma_r = .05$, our current best. I changed the intensity settings to .075 but that didn't prove to be too useful. After testing more combinations, I found the optimal settings of $\sigma_s = 4$ and $\sigma_r = .05$ for the no-flash image, and $\sigma_s = 4$ and $\sigma_r = .10$ for the flash image to produce the best, newly synthesized image. Still, I was very close before, and it's near indistinguishable anyway, I just like to be a perfectionist.

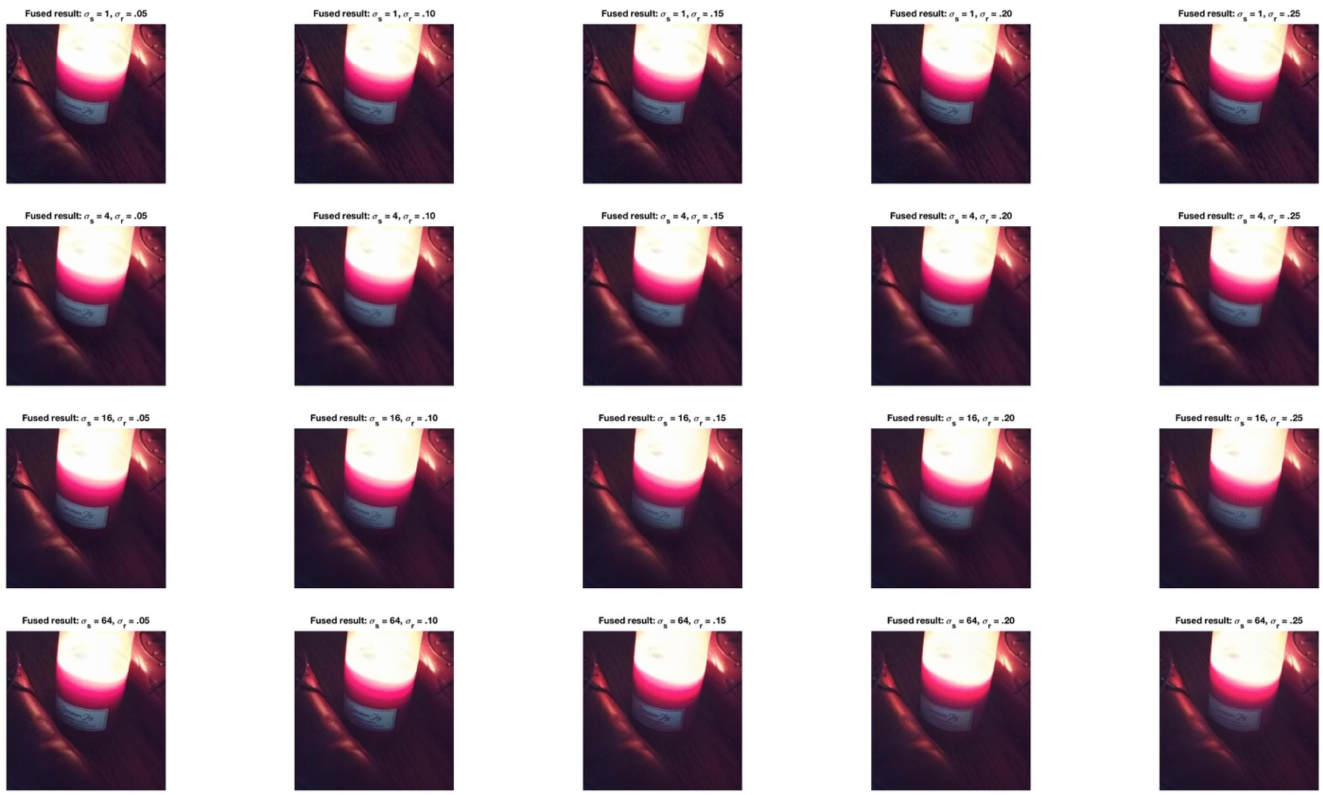

Figure 5: Every combination of fused images, just for fun

The best image is below in **Fig. 6**.

Now back to epsilon. I decided to test what would happen if I increased or decreased epsilon in the fusion equation. It seems it's only there to keeping the function from dividing by 0, but what would happen if we increased it? As you increase epsilon, the value of the fraction approaches 1, and therefore N_{final} approaches $N_{denoised}$, ie. the final result looks almost identical to the denoised, no-flash image. Now what if we were to reduce value of epsilon further down to, but not exactly at 0? As expected, we would slowly be trading back a little noise for sharper details. I decided to arrange an illustration of this as **Fig. 7**, at the very end. In my opinion, going down to $\epsilon = 0.1$ is not a bad idea, so a really great result ended up using the settings $\sigma_s = 4$, $\sigma_r = .05$, $\varepsilon = 0.1$.

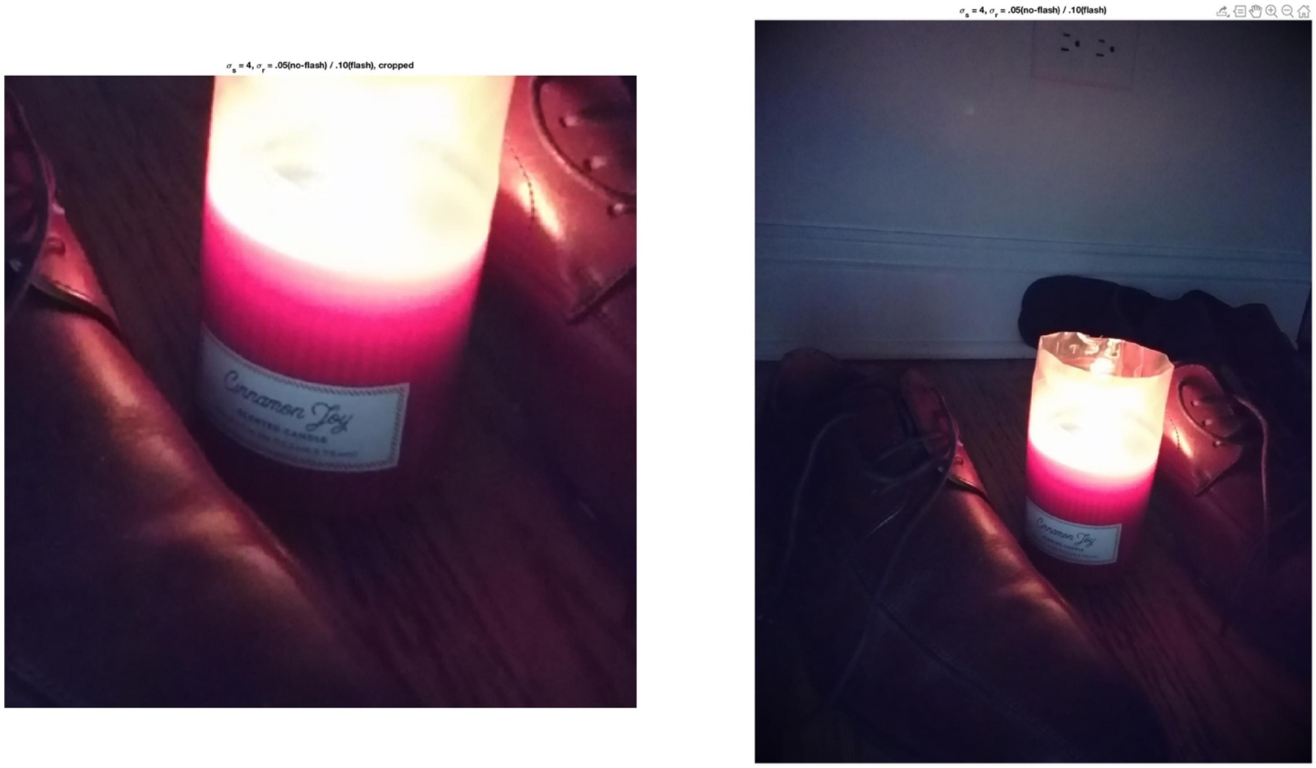

Figure 6: Fusion of $\sigma_S = 4$ and $\sigma_r = .05$ (no-flash) and $\sigma_S = 4$ and $\sigma_r = .10$ (flash)

Figure 7: Testing different epsilons on the fused image.
Values range from 0.2, 0.1, and 0.05 from left to right

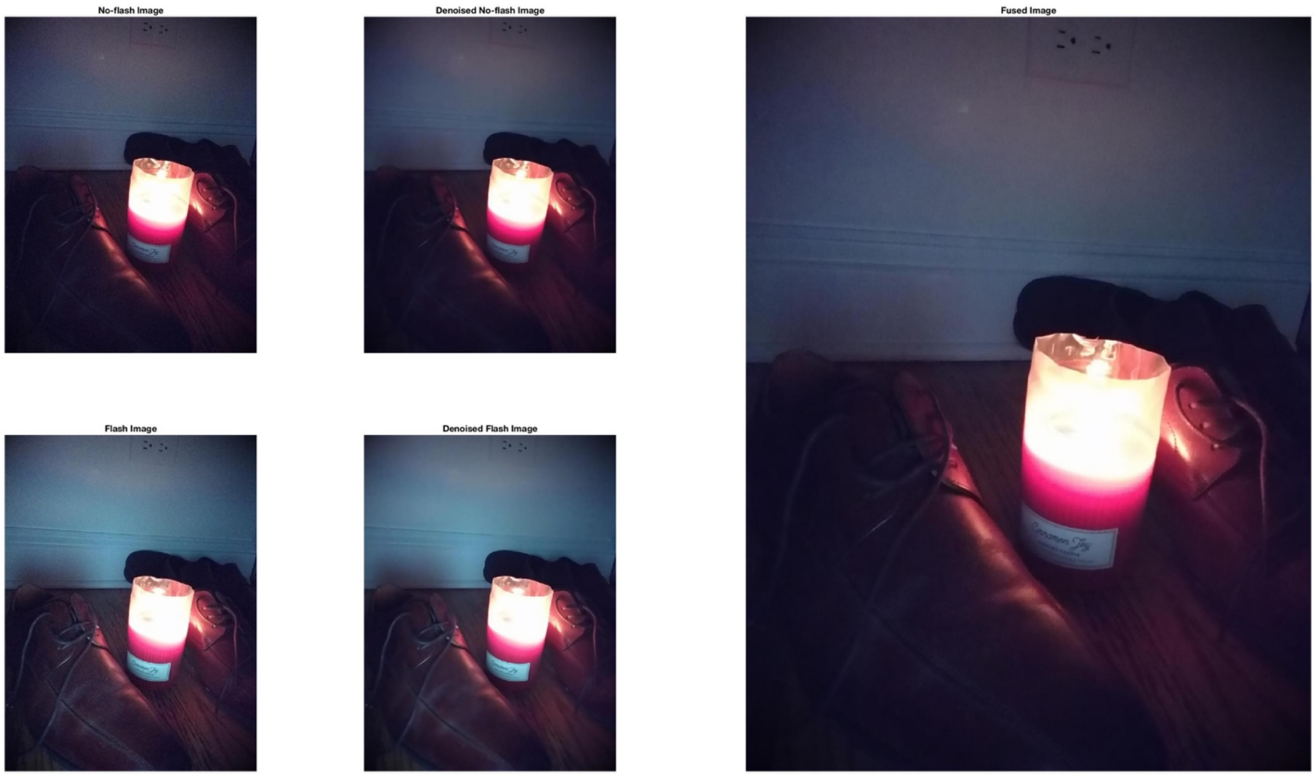

Figure 8: The same as Figure 4, but I didn't crop the images. Figure 4 was more useful just to identify key differences, but I wanted to show I didn't forget to not use cropped images.

Update for Florian

Florian asked me to provide an update with new images because the results were not convincing enough. I repeated the experiment above with the images he provided and found some new findings. All the new results are in the folder titled 'NEW RESULTS', and the new MATLAB file is titled 'main_updated_for_florian.m'.

While the experiment and code remained relatively the same, I decided to use an σ_s of [1, 2, 4, 8, 16], and an σ_r of [.05, .10, .15, .20]. These two ranges are a little smaller than the previously chosen ranges from the first run that included σ_s of 64 and σ_r of .25. I believe the reason for a smaller spatial domain has to do with the fact that the image dimensions are smaller, and the shorter intensity domain was just a preference when I observed all the images. All steps were taken the same as above in the first run, so the results of the 20 combinations of spatial and intensity domain in bilateral filtering for the no-flash image are tabulated in **Figure 9**, below. **Figure 10** depicts the same process done for the flash image.

Figure 9: Bilateral filter passed over the **no-flash** image

Figure 10: Bilateral filter passed over the **flash** image

Selecting the “correct” de-noised images from this image was harder than before. I was more or less confused what constituted the correct level of noise reduction and for what elements. For example, some spatial and intensity level combinations would reduce noise on the pots, while other levels reduced noise on the wood and wall but also further blurred the pots’ art style which I wanted to keep, so maybe it would be wiser going into bilateral filtering by first deciding on what elements we want to focus on de-noising, and which elements we are comfortable with leaving the way they are.

Ultimately, I decided to focus on de-noising the pots and keeping the details of the wood board. The best de-noised results of both no-flash and flash images used the settings $\sigma_s = 8$ and $\sigma_r = .05$. **Figure 11** and **Figure 12** on the next two pages depict the flash and no-flash images, along with their best de-noised results (in my opinion), and the fused image use the equation from Page 3 above. In Figure 12, while the pots look similar to the de-noised no-flash image, you can observe the difference in the details of the wood table.

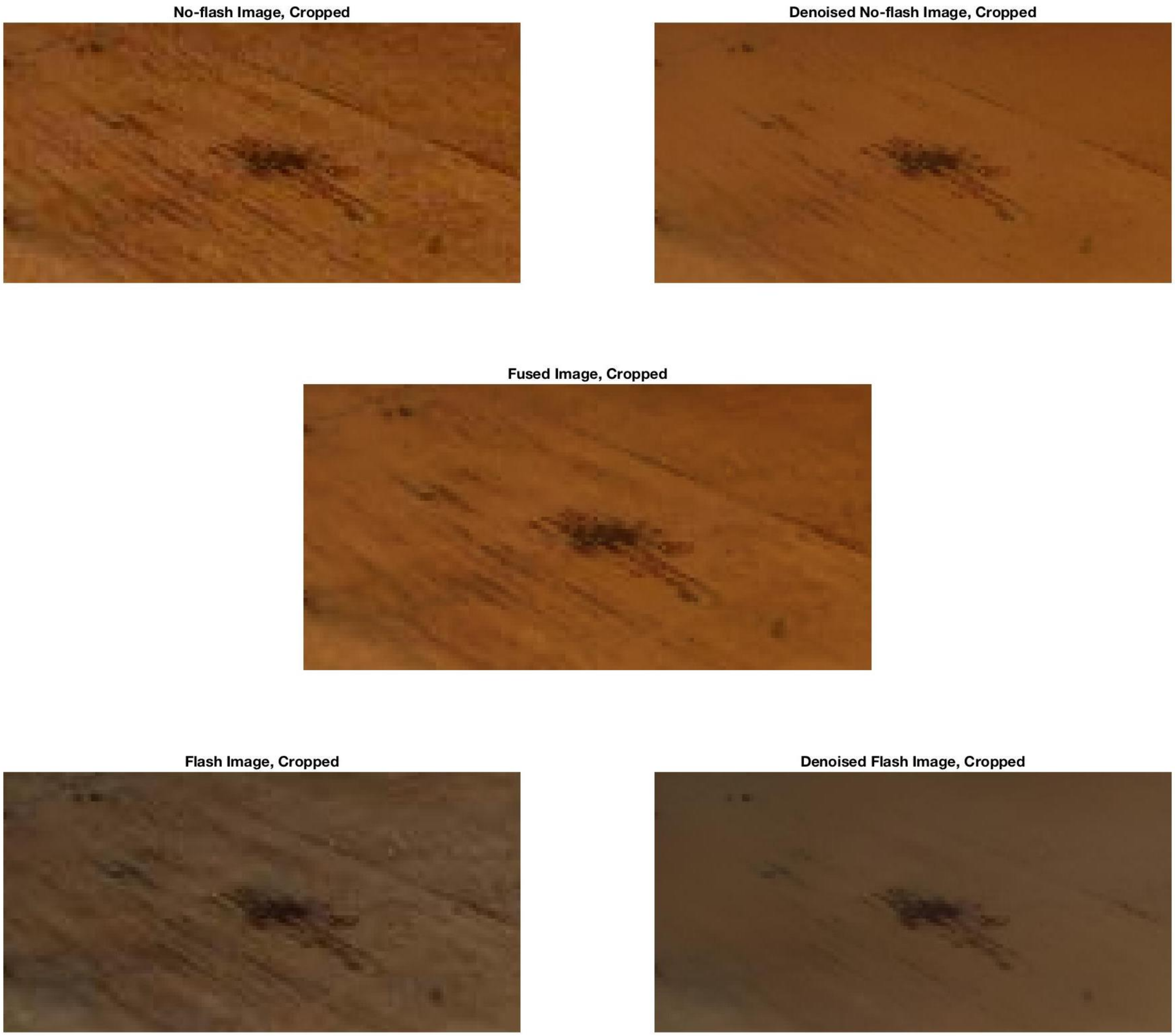

Figure 11: Original images, denoised results of original images using best setting, and fusion result

Figure 12: Same as Figure 11, but the whole image.

Figure 13, below, shows a comparison of 2 cropped segments of the image to show the difference in level of noise from the original to fused result.

Figure 13: Comparing 2 cropped locations to show the fusion results from the original image

Just to be safe, I ran a fusion across all combinations of spatial/intensity domains (**Figure 14**, below), but still found my original settings of $\sigma_s = 8$ and $\sigma_r = .05$ to be one of the best fusions.

Figure 14: Every combination of fused images, just for fun

Finally, and this I didn't expect, I found different results in my epsilon test than what I had determined in the first run. I reran the fusion result with epsilons [.3, .2, .1] and compared it to the original image. In my first run with the image of the candle and shoes, I found $\varepsilon = .1$ (below the .2 setting we were suggested to use) to be the better setting, but for this image, $\varepsilon = .3$ seemed to show better results. While I didn't expect it to change, this still makes sense, as a slightly higher epsilon value provides more weight to the no-flash image's level of detail than the detail extracted from the ratio of the flash image over the de-noised flash image. The comparison is depicted in **Figure 15** on the last page.

Fused image, $\sigma_s = 4$, $\sigma_r = .05$, $\epsilon = .3$

Fused image, $\sigma_s = 4$, $\sigma_r = .05$, $\epsilon = .2$

Fused image, $\sigma_s = 4$, $\sigma_r = .05$, $\epsilon = .1$

Original Image

Figure 15: Epsilon test with ε = [.3, .2, .1] from the top, and original no-flash image at the bottom