Assignment 4: Image Enhancement

Harry Kim

Handed out: September 19, 2022
Due: 11:59pm, October 7, 2022
Handed in: 11:59pm, October 7, 2022

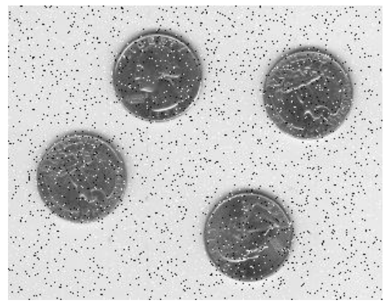
Important Notes:

- Feel free discuss the homework with the instructor or the TAs.
- Handwritten solutions will not be accepted.
- Turn in a PDF report and .m/.py files through Canvas as a compressed (.zip) file; turn in a hardcopy of PDF printout in class)

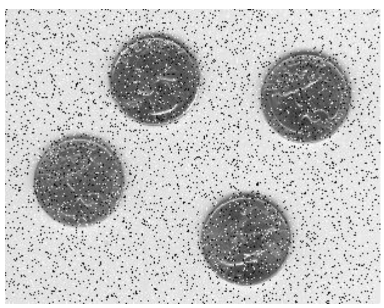
Question 1: Noise removal

(a) Write a function that adds different levels of "salt pepper" and Gaussian noise to a given image. This function should be generic enough to add noise to both grayscale and colored images. Display noisy images when having salt pepper noise with 5%, 10% and 20% of noisy pixels. Display the noisy images when adding Gaussian noise with zero mean and standard deviations of 0.1, 1.0, and 10, and 10. Report your observations for each noise type as you increase the level of noise.

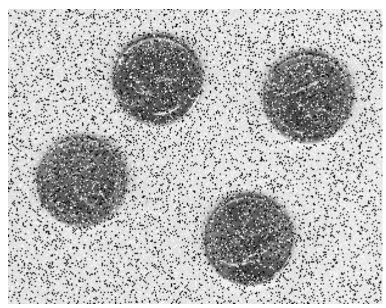
(a) Original image



(b) Salt Pepper 5%

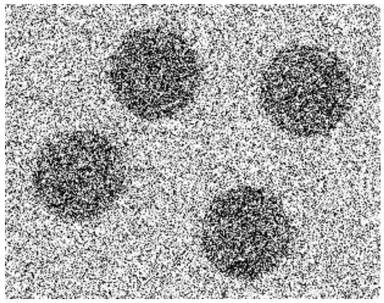


(a) Salt Pepper 10%

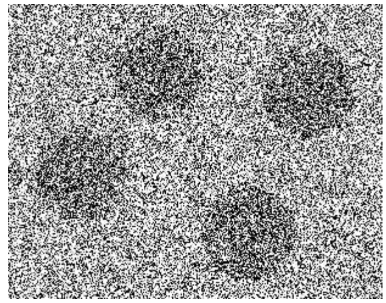


(b) Salt Pepper 20%

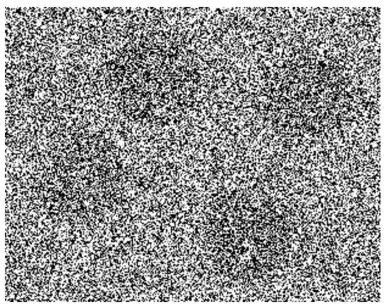
As the density of salt pepper noise is increased, the amount of noise is increased with more and more dots with extreme values (0 and 255) appearing on the image.



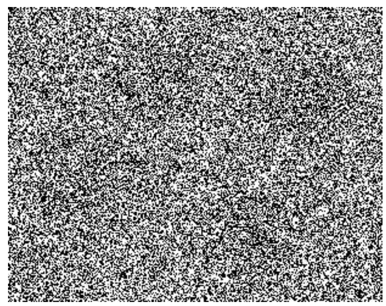
(a) Gaussian std: 0.1



(b) Gaussian std: 1.0



(a) Gaussian std: 10



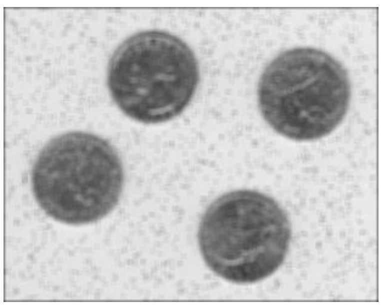
(b) Gaussian std: 100

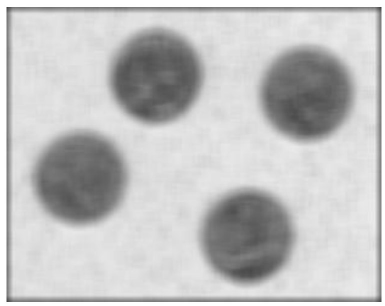
As the standard deviation for the Gaussian noise increases, the image becomes more distorted with "static" noise and it makes the original image harder and harder to see.

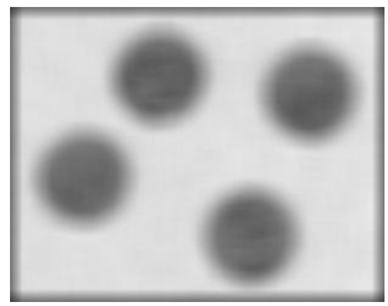
(b) Using images that you have generated from (a), write a function that will remove the noise using mean, median, and Gaussian filtering. Experiment with different filter sizes (5x5, 11x11, and 21x21). Report the results of each experiment. Compare and comment on the results, including the impact of the filter size. Explain which filter is better for which image and noise type. You can use image difference to highlight the difference between the results of two different filters using the same size and input noisy image. Use this difference to comment on if you see difference between mean and Gaussian filters. Explain why or why not you see a difference.

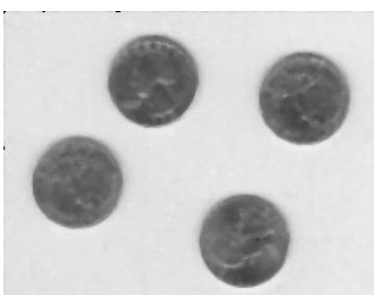
For this document, I chose salt&pepper noise at 5% density and Gaussian noise with a standard deviation of 0.1 for visuals. All other images have been generated and are included within the output-images folder.

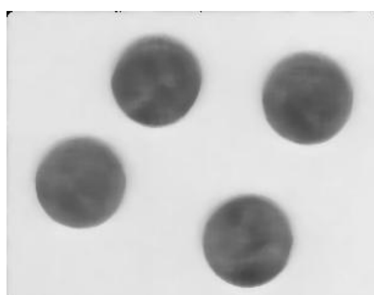
S P noise:

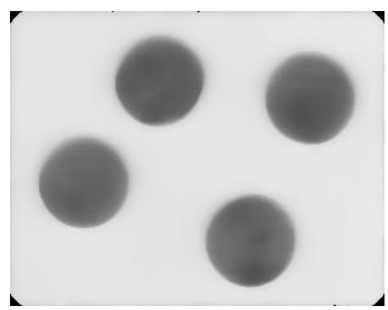

(a) Mean filter 5x5

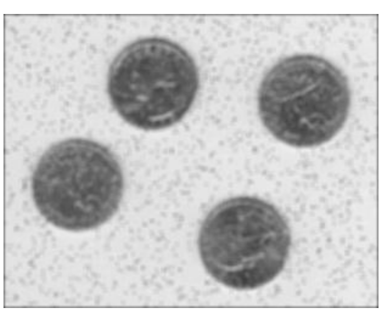

(b) Mean filter 11x11

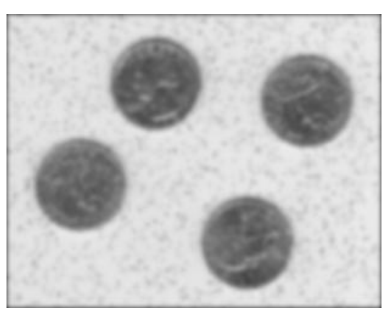

(c) Mean filter 21x21

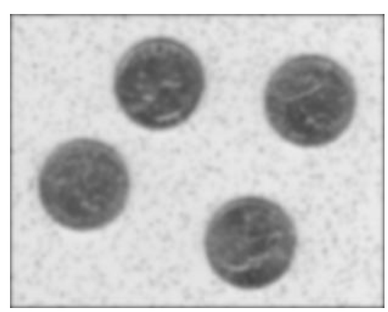

(a) Median filter 5x5

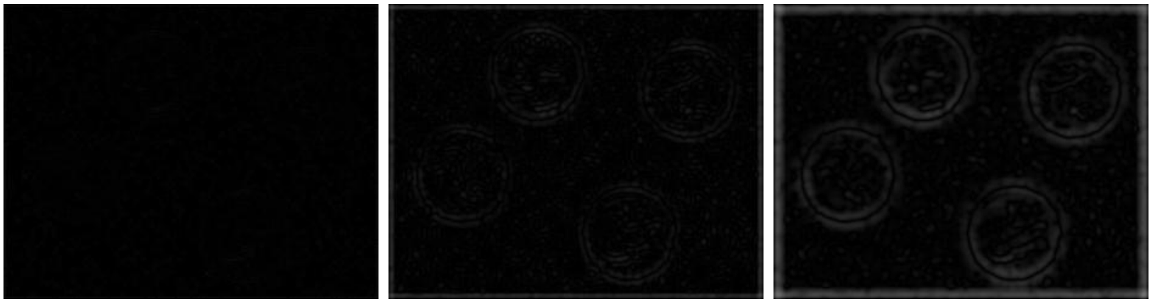

(b) Median filter 11x11


(c) Median filter 21x21


(a) Gaussian filter 5x5


(b) Gaussian filter 11x11

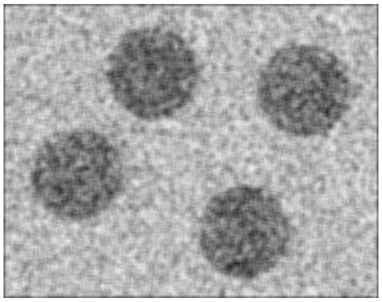

(c) Gaussian filter 21x21



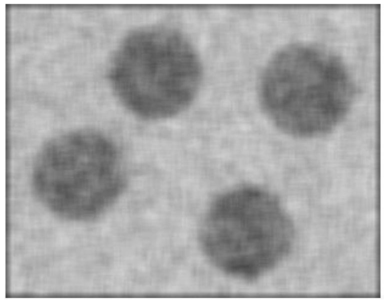
(a) M&G Abs diff 5x5 (b) M&G Abs diff 11x11 (c) M&G Abs diff 21x21

For salt and pepper noise, it would seem that the mean filter smooths out some noise, but blurs the image as the filter size increases. The median filter retains the edges of the coins very well, and removes salt and pepper noise very well also, but all fine details are completely smoothed over and the image takes on a milky glass effect. The Gaussian filter just blurs the entire image to attempt to smooth out the s&p noise. As we increase the density of noise, however, each respective filter has a harder and harder time getting rid of the noise. The absolute difference between the mean and the Gaussian filtered images shows that they do work differently especially at the edges of the coins.

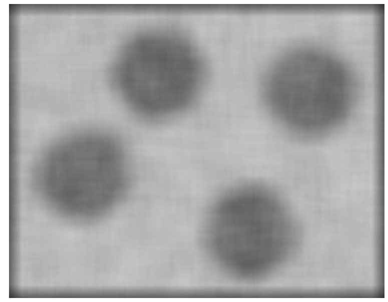
Gaussian noise:



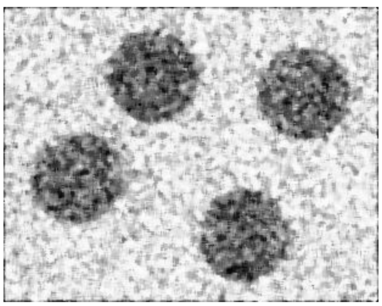
(a) Mean filter 5x5



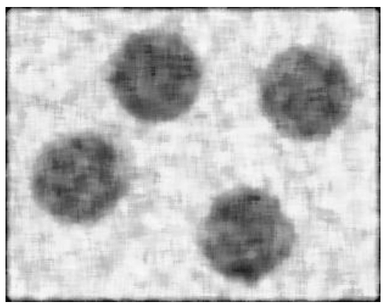
(b) Mean filter 11x11



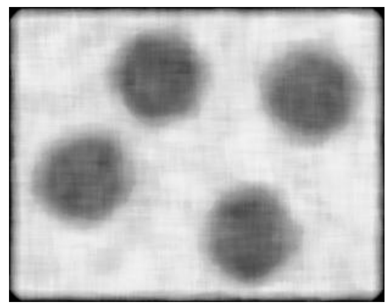
(c) Mean filter 21x21



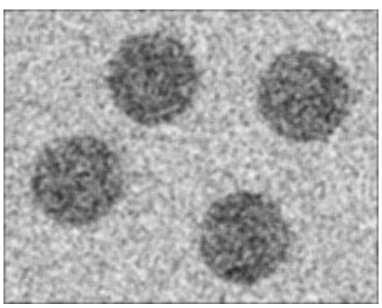
(a) Median filter 5x5



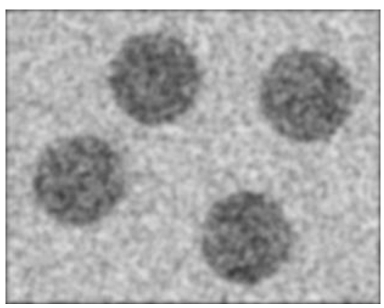
(b) Median filter 11x11



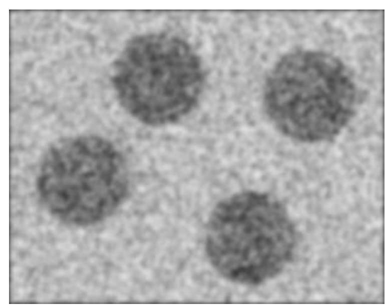
(c) Median filter 21x21



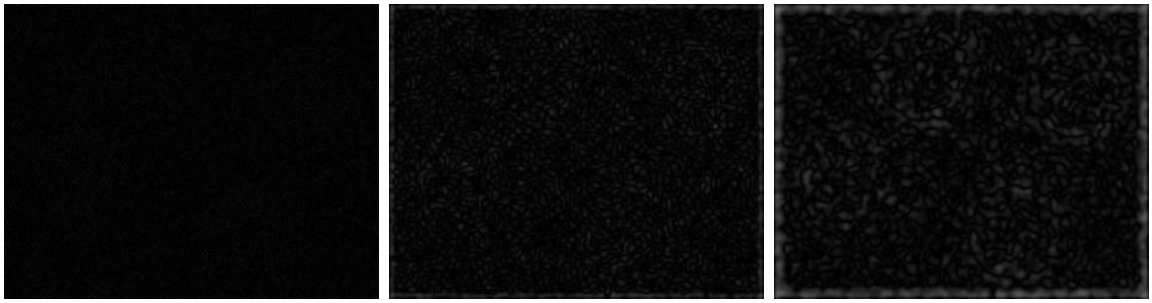
(a) Gaussian filter 5x5



(b) Gaussian filter 11x11



(c) Gaussian filter 21x21



(a) M&G Abs diff 5x5 (b) M&G Abs diff 11x11 (c) M&G Abs diff 21x21

For Gaussian noise, it would seem that the mean filter, as an attempt to smooth out the noise, darkens the image and blurs the image as the filter size increases as there is more variation of values. The median filter struggles to remove Gaussian noise as median values are not as extreme as they were with the salt and pepper noise. The Gaussian filter just blurs the entire image to attempt to smooth out the s&p noise. As we increase the density of noise, however, each respective filter has a harder and harder time getting rid of the noise. The absolute difference between the mean and the Gaussian filtered images shows that they do work differently especially at the edges of the coins.

Question 2: Edge detection

(a) Convolve an image with the below filters. Display the input and output of the convolution operator. What do these filters do to the images? What type of features do you observe in the output? Can you observe any variations in the output images, is one better than the other in some features?

(a) First filter

(b) Second filter

(c) Third Filter

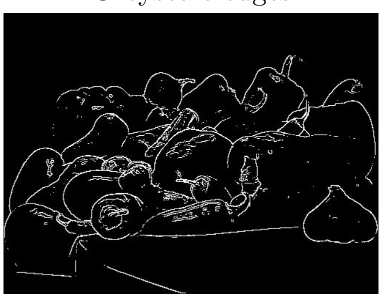
The first filter details all the vertical edges very well, but not the horizontal ones. It looks like a stack of paper on it's side. The second filter details all the horizontal edges very well, but not the vertical ones. It looks like a stack of paper. The third filter kind of does both horizontal and vertical edges and it looks like a bunch of dots that make up an image. It also picks up the details of the background very well, as it is hard to tell with the first and second filter.

(b) Write a function that applies the Roberts, Perwitt, and Sobel edge detectors on a given colored image. Display and comment on the results of each detector. Experiment with applying the filters on a grayscale version of the color image versus applying the filters on each color channel. Display and report all results. Explain which one is better in detecting edges and why. Comment on how some of the edge responses related to distinct color channels. Display results as a three-color channel and as individual color channels. When an edge is visible as white in the display three-color channel, what does that mean? Repeat all previous steps using the HSV color space. How do the results differ in this case?

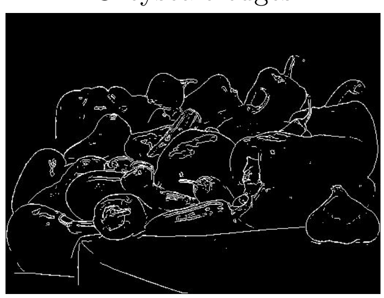
(a) Roberts
Greyscale edges

(b) Perwitt
Greyscale edges

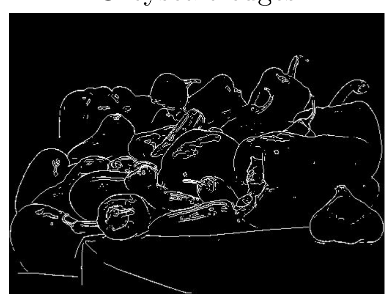
(c) Sobel
Greyscale edges



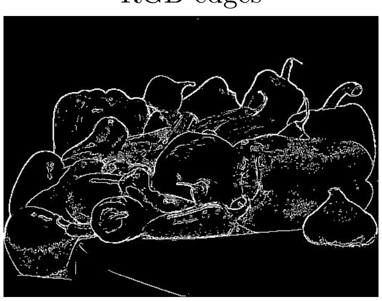
(d) Roberts
RGB edges



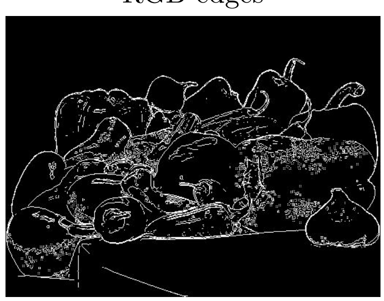
(e) Perwitt
RGB edges



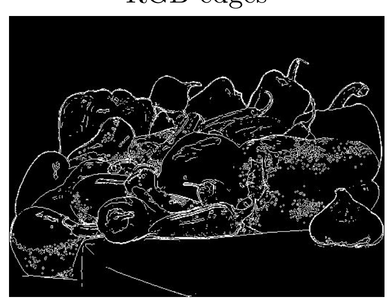
(f) Sobel
RGB edges



(g) Roberts
HSV edges



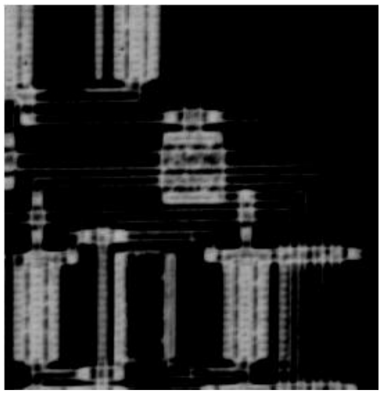
(h) Perwitt
HSV edges



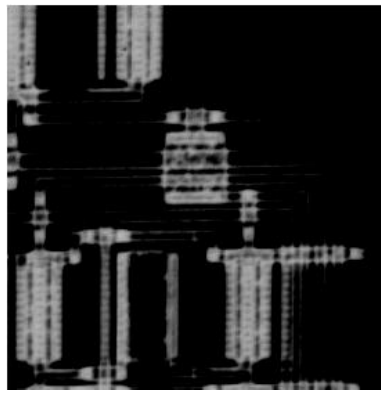
(i) Sobel
HSV edges

For the different ways to apply the filter, it seems that applying the filters to the separate RGB channels is the most effective at giving the clearest edges. There seems to be more "edges" detected for the color channels as it has 3 whole images, essentially, with varying values depending on the color to work with rather than a single grey-scale one. When an edge is visible as white in the display 3-color channel, it means that there is a extreme change in value for each channel. The HSV differs from the rest as it defines even more edges within the peppers.

(c) Write a function that computes the PSF convolution with the object/scene function (a given image as in (b)) using a random percentage of the points in the object function. For PSF, you can use a Gaussian function constructed by the function you wrote in (a). Experiment with different random percentages by randomly selecting a percentage of the input image pixels and set them to zero. What happens as the percentage of "known" points in the input domain increases and decreases? Comment on the results. Display the input and output of each experiment.



(a) Perwitt filter
3x3 matrix



(b) Perwitt filter
lin sep

(c) Sobel filter
3x3 matrix

(d) Sobel filter
lin sep

In the first image of each row, it is the normal filters for both x and y. For the second of each row, the x and y derivative filters for both Perwitt and Sobel have been linearly separated. For example, for the X derivative of Perwitt: [1; 1; 1] and [-1 0 1]. The two linearly separated matricies have been multiplied together and then used to detect the edges of the image. They result in the same image.

Question 3: Edge enhancement

(a) Write a function that applies a Laplacian edge sharpening to a given image. Report and comment on the results. What features of the image were enhanced?

(a) Intensity levels: 4 (b) Intensity levels: 8

The edges between the objects (the coins) and the background are sharper along with details within the coins and the noise of the background (the occasional dark spots are more pronounced).

(b) Write a function that extends the sharp operator to color images. How do the areas of image sharpening compare with the areas of edge-detection in Question 2(a)?

(a) Intensity levels: 4 (b) Intensity levels: 8

The details of the change of colors is more pronounced along with all the edges of the different peppers. The areas of sharpness are similar to the areas of edge detection in question 2a, and I think this is because those areas are the ones that are getting pronounced.

Question 4: Extended Projects