**EEL 4930/5934**

**Introduction to Biomedical Image Analysis**

**Assignment – 6**

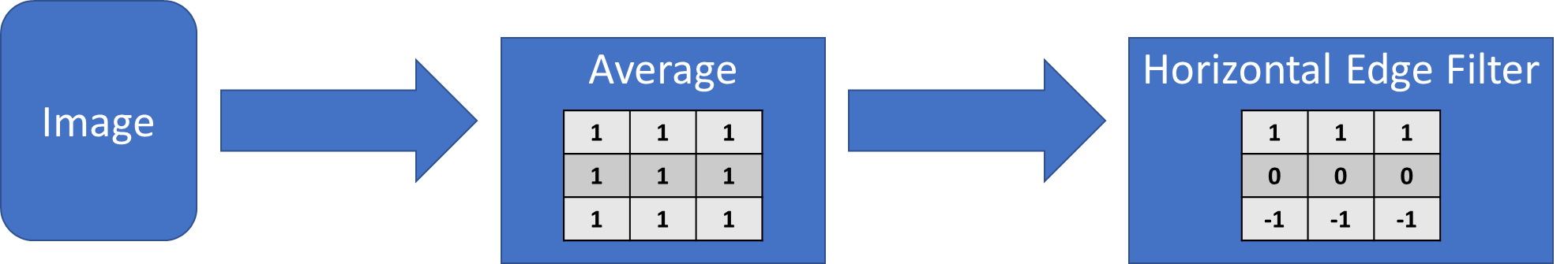
**Due: 03/12/2024, Noon**

**Question 1: Linear vs. Non-Linear filters (5 points)**

Last week we discussed how filters can be linear or non-linear in regards to how the order in which they are applied influences the output. For this problem, we will be using *Assignment\_6\_codes.m* to demonstrate sequence dependence/independence of different filters and image operations. For the following scenarios, (i) apply the filters to the image in the order presented and include your output, (ii) apply the same filters to the image in reverse order and determine if the result is the same.

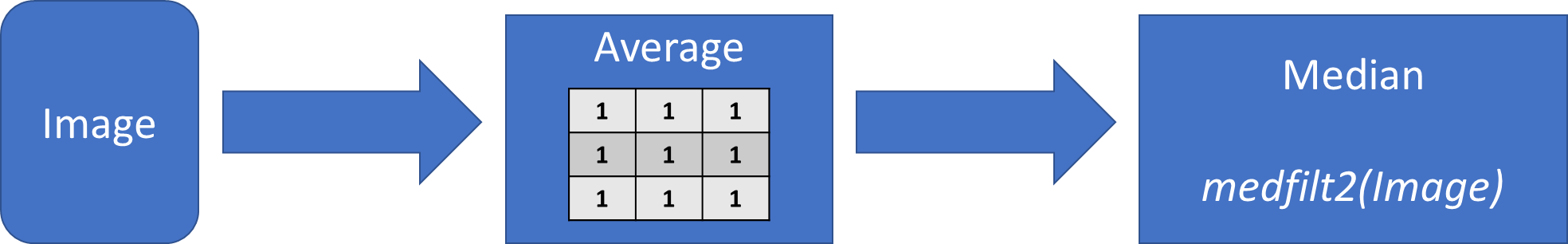
A black and white image of a white circle

Description automatically generated



A close-up of a grey surface

Description automatically generated



A white circle on a black background

Description automatically generated

Which of the three filters are linear? Which are non-linear?

The average and horizontal edge filter are both linear. The median filter is non-linear. The results from applying the filters can be found above.

**Question 2: Identifying filter kernels (5 points)**

For the following 3x3 kernels, identify the type of filter and briefly explain the effect on an image.



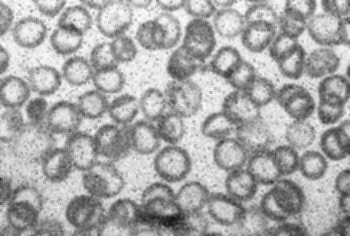
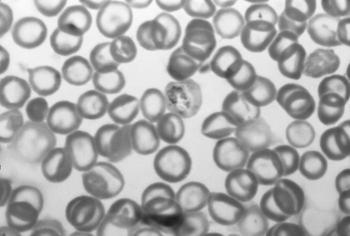
This is a Laplacian Kernel, which will highlight the areas of rapid change vertical and horizontal change in an image. The 8 in the middle of the filter will be contrasted with the pixels immediately surrounding it (not on its diagonals). If the center pixel is significantly higher in intensity than those pixels, the filter will be highly positive. Similarly, if the center pixel is significantly lower in intensity, the filter will be highly negative. If there is little change in intensity, the filter will be close to 0.

This is an Identity Kernel. It has no effect on the image (provided the image is padded when performing convolution). It preserves the pixel value in the center of the filter as it is convolved across the image.

This is a Sobel Vertical Edge Detector. This kernel will be high when there are rapid frequency changes in the vertical direction, and low when there are no such changes. This is because mathematically this filter is comparing the left and right side of the center point. When the left side and right side are similar, the kernel “cancels out” and takes a low value. When they are different, it is either strongly positive or strongly negative.

**Question 3: Application to real images (5 points)**

For the following noisy image, experiment with a few of the filters we discussed in class or others in order to create the cleanest image possible. (Hint: There is no single right answer and you will not be penalized for not producing a pristine image.)

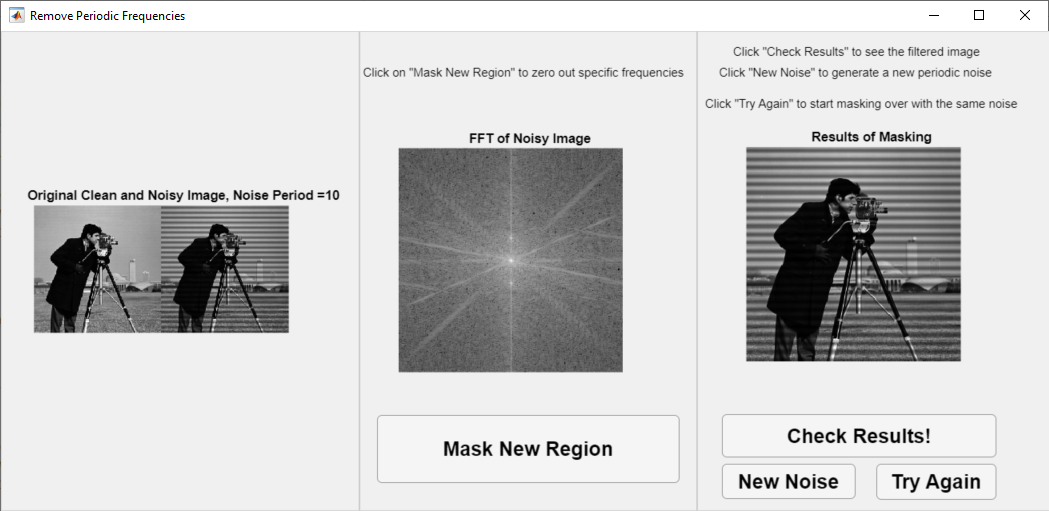


Close-up of several images of blood cells

Description automatically generated

**Question 4: Playing with Frequencies: (10 pts)**

Included in this assignment is a Matlab application called “*Remove\_Periodic\_Frequencies.mlapp*”. You can start up this application by typing “Remove\_Periodic\_Frequencies” in the command window and pressing the Enter key or by opening the “*Remove\_Periodic\_Frequencies.mlapp”* script in Matlab AppDesigner and clicking the green run arrow at the top. This application is designed to illustrate how an image can be translated into Fourier space and how, by masking regions in the Fourier space, we can remove periodic noise.



1. Click on the “**New Noise**” button a few times. How does the center image, displaying the 2D Fourier Transform of the image, change with different periods of noise? How do the higher noise period FT images differ from the lower noise period FT images? **(2 pts)**

**I do not have MATLAB. This course is supposed to be able to be completed without MATLAB, so I in the following answers I have taken my best guess/hypothesis as to what would happen if I were able to run the mlapp file.**

The center image (the Fourier Transform) will contain a vertical line representing the high-frequency changing at the edges of our noise. In addition, it will contain areas of increased low-frequencies due to the “dampening” of local pixels due to the uniform darkening in areas of the noise. As the period increases, the frequency of the bands decreases and we will see a dimmer vertical line and more noise in the center of the FFT.

1. Click on the “**Mask New Region**” button. Then hover your mouse over the center axis. You should see a set of crosshairs where your mouse cursor used to be. By clicking somewhere in this axis, a circular mask is created and zeros out a given area of the FT image. You can then click the “**Check Results!**” button to view the results of the inverse Fourier Transform. To start over with the same noise period, click the “**Try Again**” button to restart.
   1. Try your best to remove the periodic noise using this masking method. (Hint: Be sure to mask out both the positive (upper) and negative (lower) components of the FT.) **(2 pts)**

For each noise, I would mask out the vertical line and the bright “spots” in the lower frequency areas (towards the center).

* 1. Repeat this for 2 different noise periods by clicking New Noise. **(4 pts)**

For each noise, I would mask out the vertical line and the bright “spots” in the lower frequency areas (towards the center).

* 1. How would you mask out all the low-frequency components of this image? Include the results below. **(2 pts)**

To mask out ALL low frequency components of an image, mask out the entire center of the image. The result would look a lot like an edge-detected image.