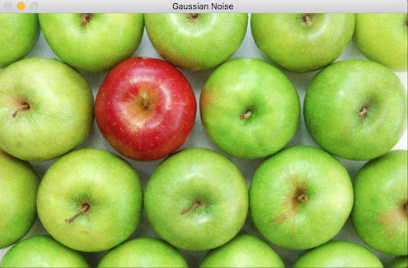
**REPORT**

The assignment consists of four parts. The first three sections are solved by manipulating the pixel values while the last section uses in-built functions provided by OpenCV. The four parts are explained as follows:

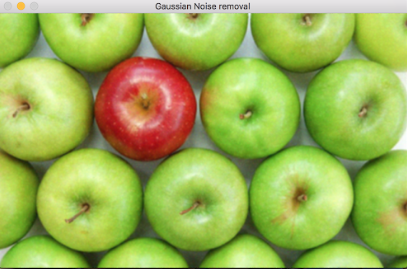
1. **Add Gaussian Noise with sigma 0.15 to an Image. Remove the noise from the Image.** We begin by creating a Gaussian Noise Matrix by using the size and type of the original image. Since the maximum variation that can be used is 15%, we obtain a random value between 0 and 15% and add it to the original image.  
      
    In order to remove the Gaussian noise, we use a 3x3 Box (Linear) Filter. This filter calculated the sum of the pixel values in each of the 9 neighboring pixels and divides it by 9. This result is used to replace the value in the center pixel. This filter is used on every pixel of the noisy image to obtain a smoothened image.  
    **Original Image:**



**Image with Gaussian Noise, N1:**

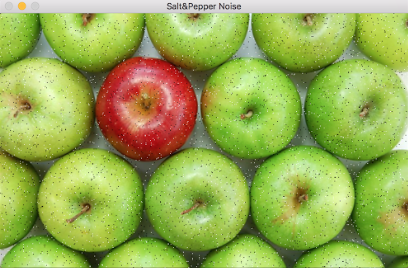
****

**Image with Gaussian noise Removed, H1:**

****

1. **Add Salt and Pepper Noise of density 0.02 to an Image. Remove the noise from the Image.** We create a Salt and Pepper Noise matrix, filled with zeros, using the size of the original image. We then assign random values between 0 and 255 for each pixel in the matrix. Since we consider the noise density to be 0.02 and it lies between 0 and 1, we multiply it with 255 to make it range between 0 and 255. Next, we check if the value of the matrix is less than half of the density. If so we add black noise to the image else white noise.  
     
    To remove Salt and Pepper Noise, we use a 3x3 Median Filter. This filter sorts the pixel values of all the 9 neighboring pixel values, finds the median and replaced the center pixel with this median. This filter is applied to every pixel in the noisy image to obtain a smoothened image.

**Image with Salt and Pepper noise, N2:**

****

**Image with Salt and Pepper noise removed, H2:**

****

1. **Create a Brightness adjusted image by adding a constant factor of 50.**An 8-bit RGB image has 3 channels corresponding to the three colors-Blue, Green and Red. In order to increase the brightness of the image, a constant factor 50 is added to each of the channels, for each pixel.  
   That is, at a particular pixel,  
    Value of R/G/B in new image = Value of R/G/B in original image + 50

**Brightened Image, B1:**

****

1. **Count the number of apples in different images.**Each image is first converted to Binary. This is followed by performing erosion and dilation. Here we use in-built functions provided by OpenCV. Further, we count the number of apples in these imaged using an in-built, connected components function. In each image, a different structuring element is used and the connectivity values also vary (4 or 8).

**Output:**  
 