CMPE 465 Computer Vision Homework 1

Tolga Özdemir

35320066566

Department of Computer Engineering, TED University

March 14, 2021

Introduction

In the CMPE465 first homework, I am expected to add noise to an image, add a blurring effect on the noisy image, and observe it. To do that, I installed and setup MATLAB. So, the whole process was done by using MATLAB.

Implementation

I used figure 1 as an image for this homework. It is an RGB image and its resolution is 1050x1680.

Figure 1

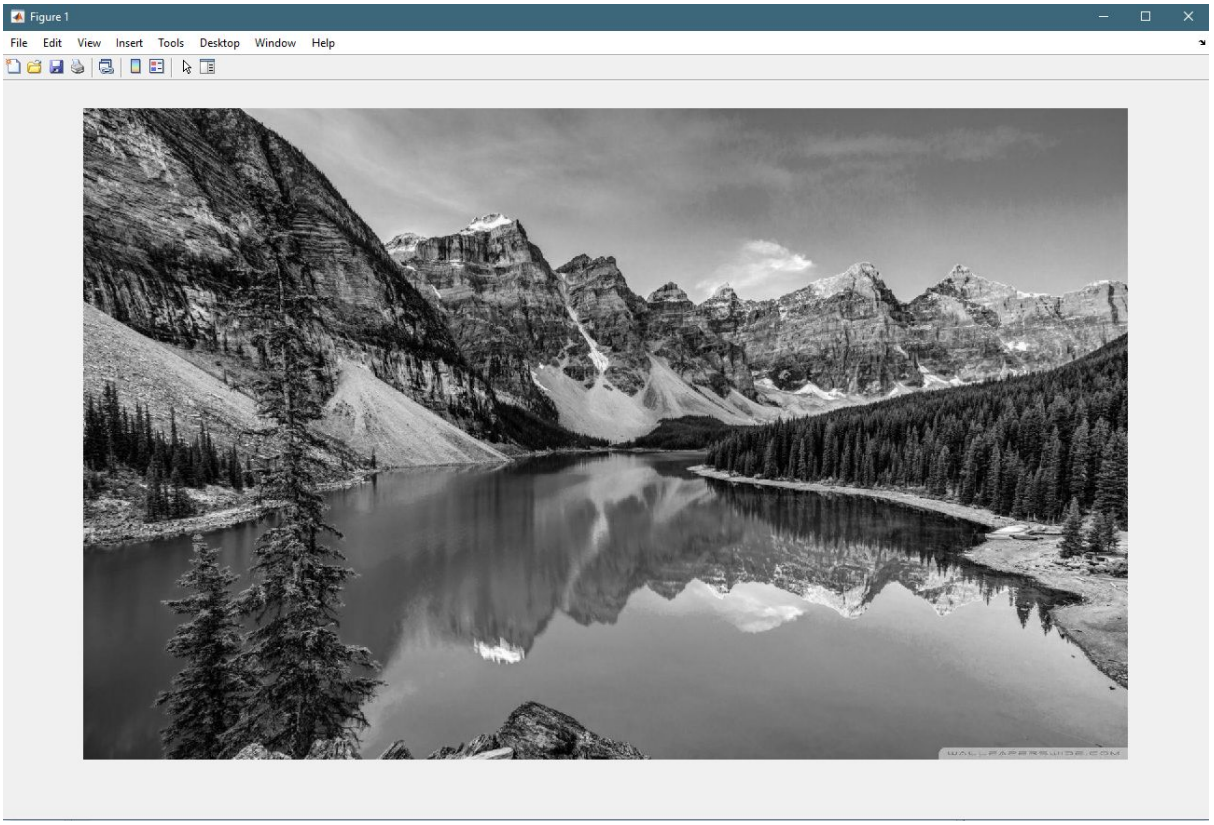
Selected image

Adding noise to the selected image

First, I read and loaded this image into the variable. Then, I obtained grayscaled image in figure 2 by using the *rgb2gray* method.

Figure 2

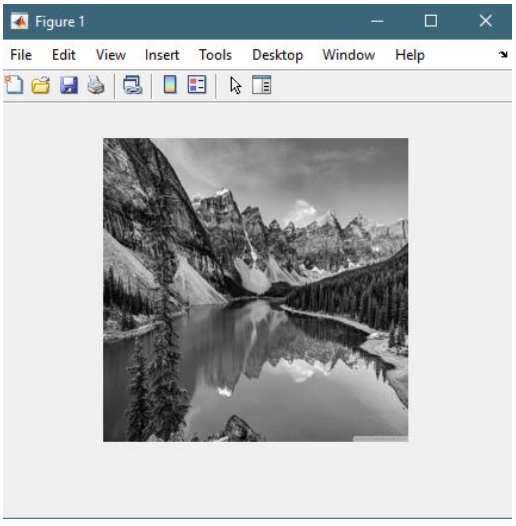
The grayscaled image



After that, I resized the image to 256x256 by using the *imresize* method in order to obtain the figure 3.

Figure 3

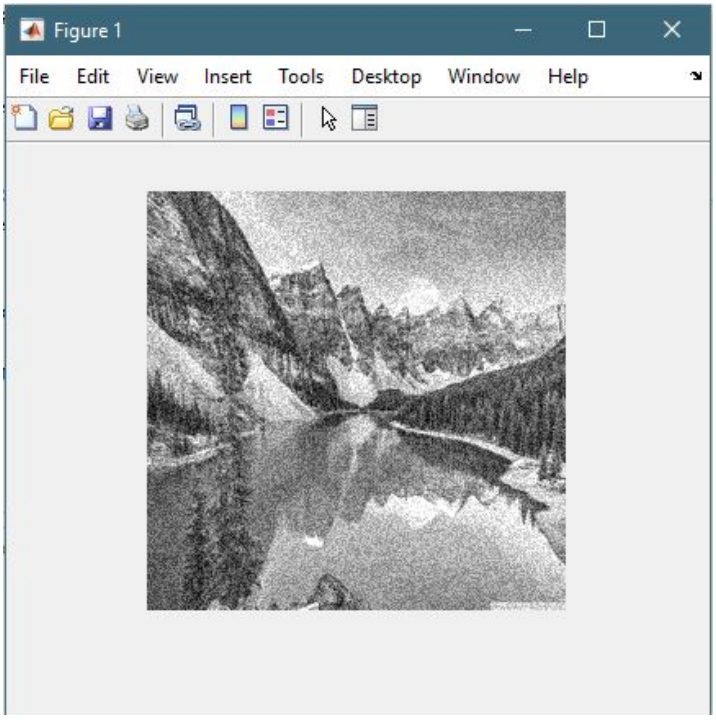
Resized image



I created random noise by using the *randi* method. Then, I smoothed the noise by element-wise multiplication with 0.3. After that, I cast the noise double to uint8 by using the *cast* method since the type of image is uint8. I added this noise to the image in figure 4.

Figure 4

The noisy image

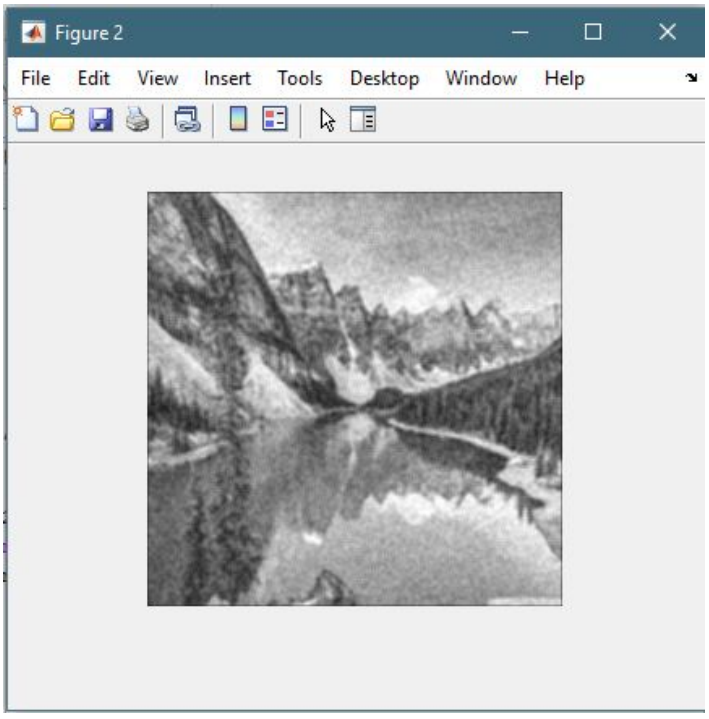


Blurred Image

To apply the blurring effect on the noisy image, I used the *conv2* method. This method takes 3 parameters: the first one is the noisy image; the second one is the 3x3 average filter which is created by this formula, $\frac{ones(n)}{n^2}$ (the *ones* method creates nxn array with all element is 1); the third parameter can take *full*, *same*, and *valid* (I used the *same* parameter since it ensures that the result is the same size with the first parameter). Then, I obtained figure 5 after casting it uint8.

Figure 5

The blurred image



Result

After my observation of the noisy image and the image with blurring effect, the blurring effects help us to get rid of noises on the image. The reason for this is that it smoothes the image. So, we can see the image more clear than the noisy version. In other words, if the computer tries to understand what is on the image, it can do a better job on blurred images rather than noisy images. However, we should tune the amount of blurriness on the image. If it has too much blurriness or if we increased the value of *n* on the average filter, the computer cannot recognize anything since the image turns into an image with only gradient colors.