***CT-part2:***

***Code:***

import cv2

import copy

import imutils

img = cv2.imread('C:/Users/dibbo/Desktop/inverse.Ai/214.png',0)

#smoothing Filter

#Manually implementing the mean filter on the img

avg\_img = copy.deepcopy(img)

for i in range(1, img.shape[0]-1):

for j in range(1, img.shape[1]-1):

sum\_pix = sum([img[l, k] for l in range(i-1, i+2) for k in range(j-1, j+2)])

avg\_img [i, j] = int(sum\_pix/9)

cv2.imshow('original image', imutils.resize(img, 500))

cv2.imshow('average image(Smoothing Img)', imutils.resize(avg\_img , 500))

#Sharpening Filter

#Manually Implementing the Laplacian Filter on the img

lap\_img = copy.deepcopy(img)

for i in range(1, img.shape[0]-1):

for j in range(1, img.shape[1]-1):

temp = img[i-1, j-1] + img[i-1, j] + img[i-1, j+1] + img[i, j-1] + \

img[i, j+1] + img[i+1, j-1] + img[i+1, j] + img[i+1, j+1] - 8\*img[i, j]

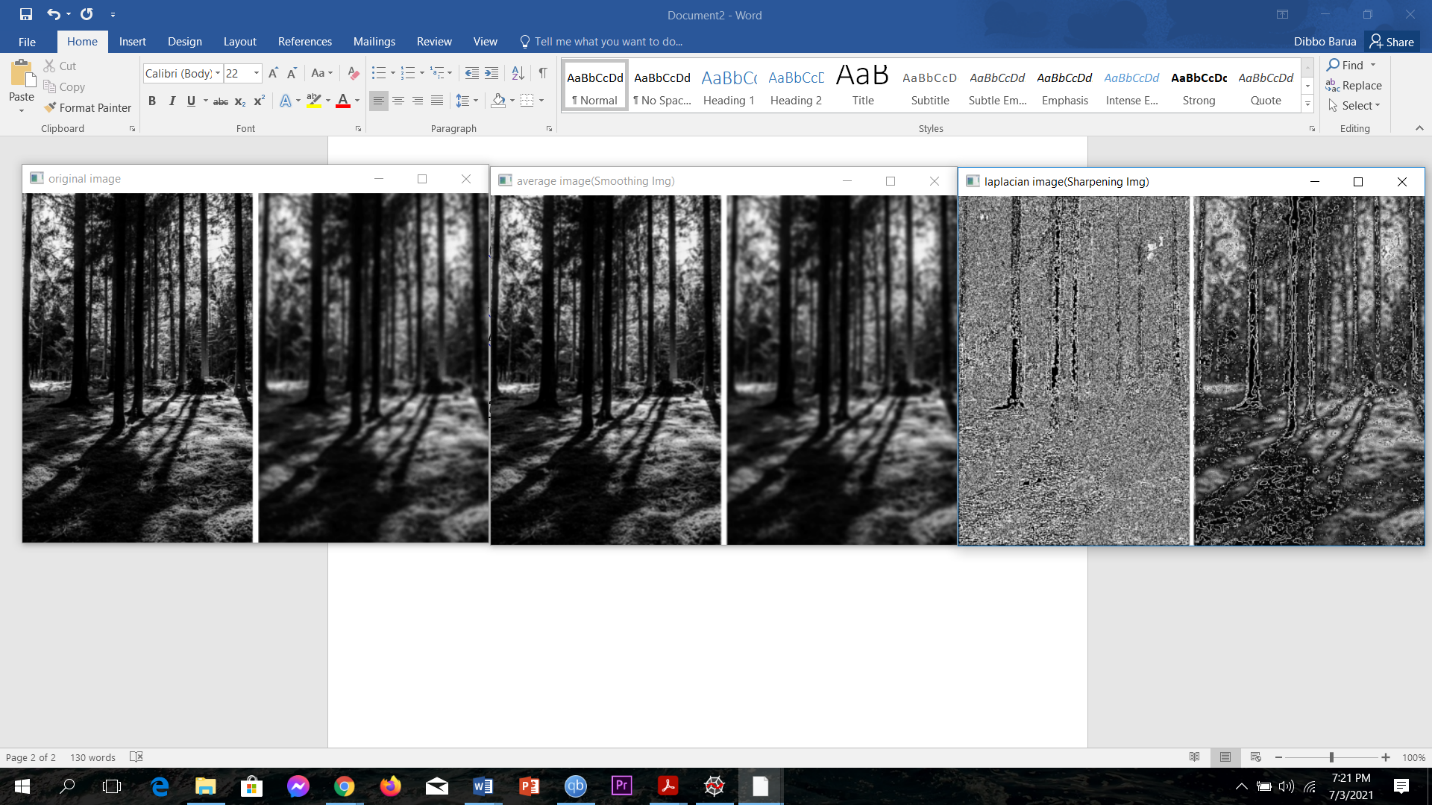
lap\_img[i, j] = img[i, j] - temp

cv2.imshow('laplacian image(Sharpening Img)', imutils.resize(lap\_img, 500))

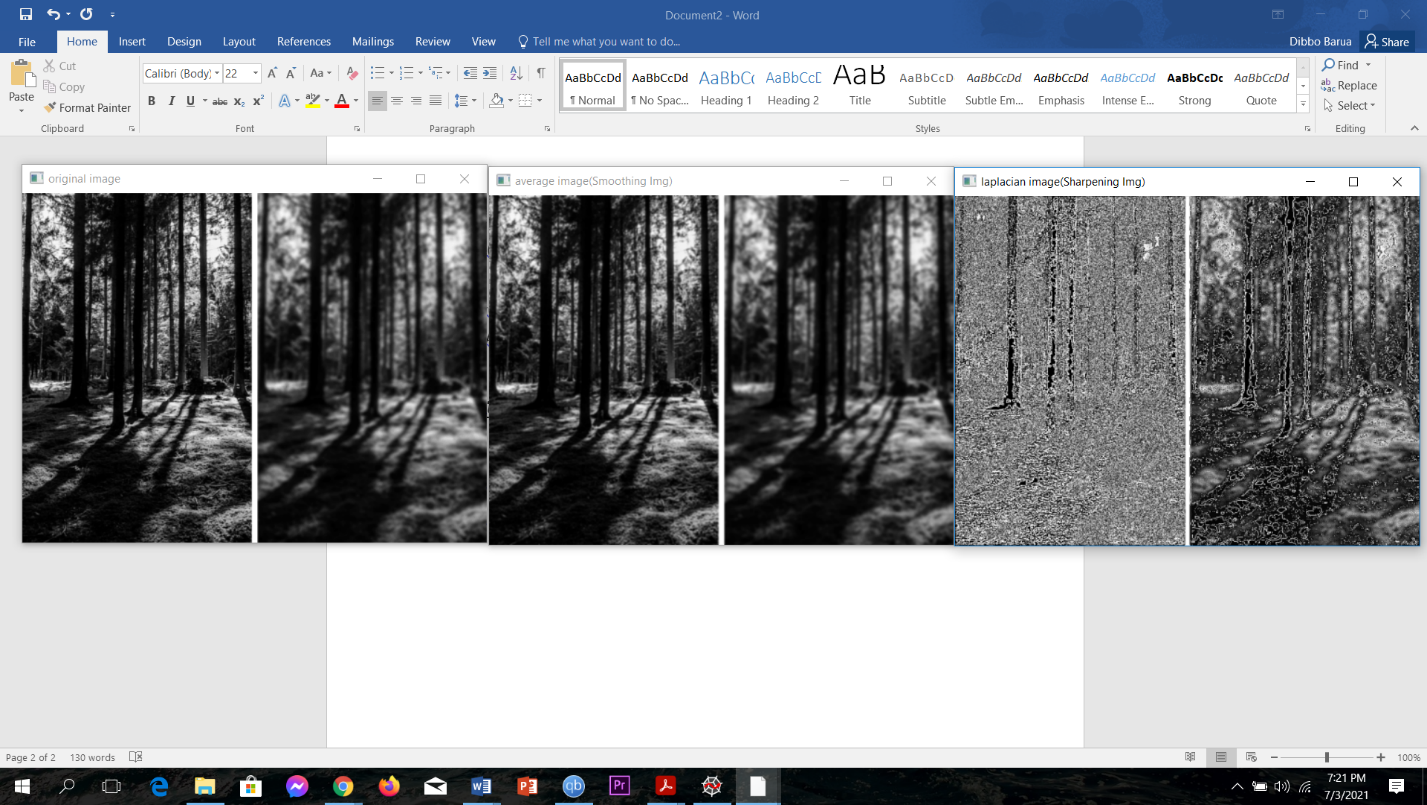
cv2.waitKey(0)

cv2.destroyAllWindows()

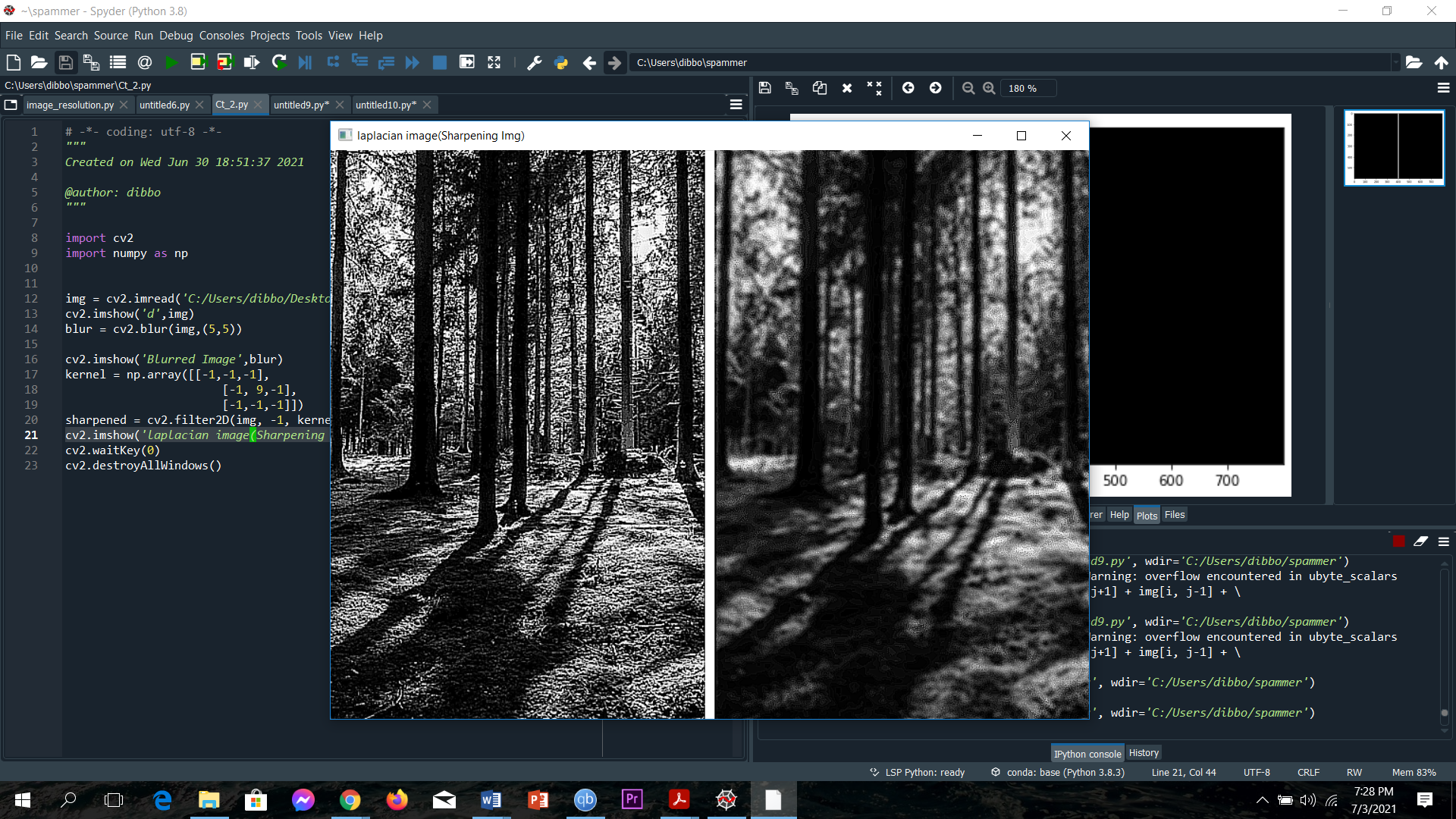
***Output:***



**Figure(1): Original Image**



**Figure(2): Smoothing Image(Mean Filter)**



**Figure(3): Sharpening Image(Laplacian Filter)**

***Explanation:***

**Smoothing Filter:**

In this Image, first I applied the smoothing filter. Here, I used the Mean Filter. Mean filtering works by simply replacing each pixel value in an image with the mean (average) value of its neighbors, including itself. This has the effect of removing pixel values that are out of place in their surroundings .Here, I Used (3\*3) mask.

|  |  |  |
| --- | --- | --- |
| 1/9 | 1/9 | 1/9 |
| 1/9 | 1/9 | 1/9 |

Each pixel in the neighborhood has the same weight, and the neighborhood average is used to calculate the filtered output value of the central pixel. This filter suppresses addictive noise. Because an average is calculated for each pixel, the image becomes blurred, especially at the edges. The blurry part gets even blurrier, and the sharpening gets even blurrier.

**Shapening Filter:**

Next, I used the sharpening Filter. Image sharpening is the process of using image differentiation to enhance edges and other mutations, weakening the area where the grayscale transformation is slow. Here, I used Laplacian Filter. I used a (3\*3) mask.

|  |  |  |
| --- | --- | --- |
| 1 | 1 | 1 |
| 1 | -8 | 1 |
| 1 | 1 | 1 |

This Laplacian filter usually find the detail of the image. A Laplacian operator will improve any feature with a sharp discontinuity (like noise, unfortunately). A Laplacian operator can be used to restore fine detail to an image that has been smoothed to remove noise. Here in this image, in the blur part , it sharpens the image but not much but in the sharpen part it finds more details. So, it becomes more sharper.