Shri Vile Parle Kelavani Mandal's
DWARKADAS J. SANGHVI COLLEGE OF ENGINEERING
(Autonomous College Affiliated to the University of Mumbai)
NAAC Accredited with "A" Grade (CGPA : 3.18)

Department of Artificial Intelligence & Machine Learning
Academic Year 2023-2024

Name: Abhay Mathur **Sapid:** 60017210016
Batch: A1

Experiment No. 3

Aim: Image Denoising

Objective: Develop a program to add noise & remove it by different operators

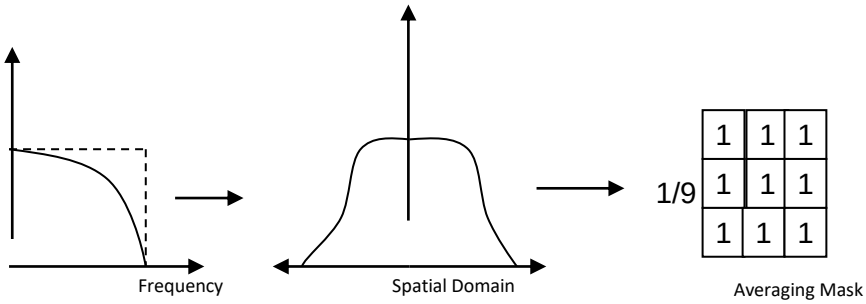
Theory:

Smoothing

Low pass filtering removes the high frequency content from the image. It is used to remove the noise present in the image which is generally a high frequency signal.

1. Averaging Filter

If an image has Gaussian noise, a low pass averaging filter is used to remove the noise. The frequency response and spatial response is show below.



From spatial response we generate the mask that would give us the low pass filtering operation. Here all the coefficients are positive and the multiplying factor for a M X N matrix is 1 / (M X N).

2. Median Filtering

The averaging filter removes the noise by filtering it till it is no longer seen. But in the process it also blurs the edges. If e use averaging filter to remove salt and pepper noise from an image it will blur the noise but will also ruin the edges. Hence when we need to remove salt and pepper noise we use non-linear filter known as Median Filter. They are also called order statistical filters because their response is based on the ordering and ranking of the pixels contained within the mask.

Image Denoising in OpenCV

OpenCV provides four variations of this technique.

cv.fastNlMeansDenoising() - works with a single grayscale images

cv.fastNlMeansDenoisingColored() - works with a color image.

cv.fastNlMeansDenoisingMulti() - works with image sequence captured in short period of time (grayscale images)

cv.fastNlMeansDenoisingColoredMulti() - same as above, but for color images.

Common arguments are:

h : parameter deciding filter strength. Higher h value removes noise better, but removes details of image also. (10 is ok)

hForColorComponents : same as h, but for color images only. (normally same as h)

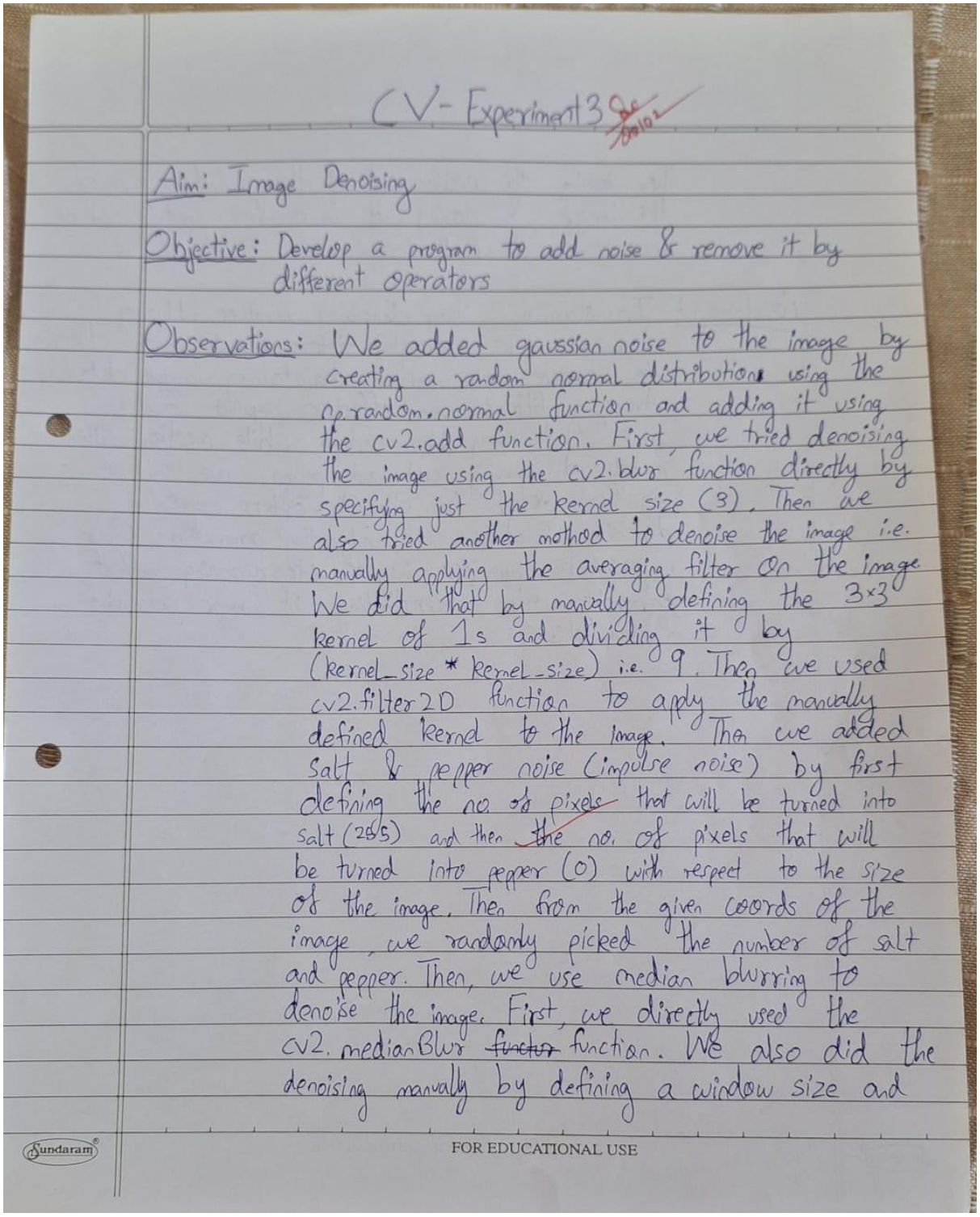
templateWindowSize : should be odd. (recommended 7)

searchWindowSize : should be odd. (recommended 21)

Problem Definition

- Add Noise & Remove it using different Operators
- Use OpenCV to denoise Images

Observations



CV- Experiment 3

Aim: Image Denoising

Objective: Develop a program to add noise & remove it by different operators

Observations: We added gaussian noise to the image by creating a random normal distribution using the np.random.normal function and adding it using the cv2.add function. First, we tried denoising the image using the cv2.blur function directly by specifying just the kernel size (3). Then we also tried another method to denoise the image i.e. manually applying the averaging filter on the image. We did that by manually defining the 3×3 kernel of 1s and dividing it by (kernel_size * kernel_size) i.e. 9. Then we used cv2.filter2D function to apply the manually defined kernel to the image. Then we added salt & pepper noise (impulse noise) by first defining the no. of pixels that will be turned into salt (255) and then the no. of pixels that will be turned into pepper (0) with respect to the size of the image. Then from the given coords of the image, we randomly picked the number of salt and pepper. Then, we use median blurring to denoise the image. First, we directly used the cv2.medianBlur ~~functor~~ function. We also did the denoising manually by defining a window size and

Sundaram

then taking the median out of the window size of the image & storing it in another matrix which becomes the denoised image once filled up.

Conclusion: In summary, image denoising involves utilizing techniques like averaging ~~filters~~ and median filtering to remove noise while maintaining image details. Averaging filters are effective against Gaussian noise but may blur edges, while median filters excel at removing salt and pepper noise without significant detail loss. OpenCV offers convenient denoising functions with adjusting parameters. Our experiments provided insights into denoising methods, enhancing our understading of image processing fundamentals.

Conclusion

In summary, image denoising involves utilizing techniques like averaging and median filtering to remove noise while maintaining image details. Averaging filters are effective against Gaussian noise but may blur edges, while median filters excel at removing salt and pepper noise without significant detail loss. OpenCV offers convenient denoising functions with adjustable parameters. Our experiments provided insights into denoising methods, enhancing our understanding of image processing fundamentals.