iJRASET
International Journal For Research in
Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 10 **Issue:** VII **Month of publication:** July 2022

DOI: https://doi.org/10.22214/ijraset.2022.46001

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VII July 2022- Available at www.ijraset.com

The use of Different Filters and their effects on Convolution with Grayscale Images

Abhik Biswas
Indian Institute of Technology, Madras

***Abstract:** Computer vision is ubiquitous in the world today, starting from unlocking phones to robots performing complex surgeries in hospitals and managing the assembly line in factories. In this paper, the focus has been placed on the use of filters or kernels or masks and their effects when convolved with grayscale images.*
***Keywords:** Computer Vision, Filters, Convolution, Fuzzy-Filters, Noise, Gaussian*

I. INTRODUCTION

When we capture an image with our smartphone camera, or with our DSLR, we do not always get the raw image in the form that we desire. Some images may be too sharp, some may be too noisy. In some cases, we significantly need to alter the colors to get the photo we desire. In this paper, some of the steps and procedures used to smoothen images and remove unwanted noise from pictures has been considered for discussion.

II. CONVOLUTION (1D) - SIGNALS

A. Linear Shift Invariant Systems (LSIS)

As is clear from the name, each LSIS system must be Linear in nature and be shift invariant. Let g be obtained after LSIS on f. That is, to say that $f(x) \xrightarrow{LSIS} g(x)$. Then, if

$$f_1(x) \xrightarrow{LSIS} g_1(x) \text{ and } f_2(x) \xrightarrow{LSIS} g_2(x) \qquad (1)$$

and α, β ∈ R, then

$$\alpha f_1(x) + \beta f_2(x) \xrightarrow{LSIS} \alpha g_1(x) + \beta g_2(x) \qquad (2)$$

The property in (2) is called the Linearity Property. Another property that LSIS satisfies is that

$$f(x-a) \xrightarrow{LSIS} g(x-a) \qquad (3)$$

The property in (3) is called the Shift Invariant Property. It can be shown that Convolution ⟺ LSIS

B. Convolution

The convolution of two functions $f(x)$ and $h(x)$, denoted by $*$ is defined to be

$$g(x) = f(x) * h(x) = \int_{-\infty}^{\infty} f(\tau)h(x-\tau)d \qquad (4)$$

Let us consider two functions $f_1(x)$ and $f_{2(x)}$ and convolve them with a function $h(x)$.

Then, $g_1(x) = \int_{-\infty}^{\infty} f_1(\tau)h(x-\tau)d\tau$ and $g_2(x) = \int_{-\infty}^{\infty} f_2(\tau)h(x-\tau)d\tau$. Now, if we consider the function $\alpha f_1 + \beta f_2$, then,

$$G(x) = \int_{-\infty}^{\infty} (\alpha f_1(\tau) + \beta f_2(\tau))h(x-\tau)d\tau$$

$$\Rightarrow G(x) = \alpha \int_{-\infty}^{\infty} f_1(\tau)h(x-\tau)d\tau + \beta \int_{-\infty}^{\infty} f_2(\tau)h(x-\tau)d\tau$$

$$\Rightarrow G(x) = \alpha g_1(x) + \beta g_2(x) \qquad (5)$$

Equation (5) shows that the Convolution operation $*$ is Linear in nature. Now, to show shift invariance, let us consider

$$g(x) = \int_{-\infty}^{\infty} f(\tau)h(x-\tau)d\tau$$

Then,

$$\int_{-\infty}^{\infty} f(\tau-a)g(x-\tau)d\tau = \int_{-\infty}^{\infty} f(\mu)g(x-a-\mu)d\mu \quad (\text{Putting } \mu = \tau - a)$$

$$= \int_{-\infty}^{\infty} f(\mu)g(x-a-\mu)d\mu = g(x-a) \qquad (6)$$

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VII July 2022- Available at www.ijraset.com

Equation (6) proves that Convolution Operation $*$ is Shift Invariant. Thus, we can apply convolution operation on a signal without changing the spatial characteristics of the signal. Upon application of a convolution filter, the signal will not undergo any change in orientation, i.e., to say that the only changes would be the shifting and enhancement of a signal.

III. CONVOLUTION (2D) – IMAGES

Any image of size $M \times N$ pixels can be considered to be a matrix of the same size. Since we are considering grayscale images, each image has only one layer. A filter or a kernel or a mask is a k-dimensional square matrix, that will be convolved with the image to extract features from the image. Since convolutions are Linear Systems, any filter that can be applied with the help of convolution can be called Linear Filters. Equation (4) given a convolution of a continuous system. However, images are not continuous – rather they are discrete objects, with each pixel being represented by an integer or a floating-point number. The 2D Convolution process is described by the equation

$$g[i,j] = \sum_{m=1}^{M} \sum_{n=1}^{N} f[m,n]h[i-m,j-n] \qquad (7)$$

where $h[i-m,j-n]$ is the kernel or filter to be used in the convolution process. It is better to use a kernel of odd dimensions for easier calculations.

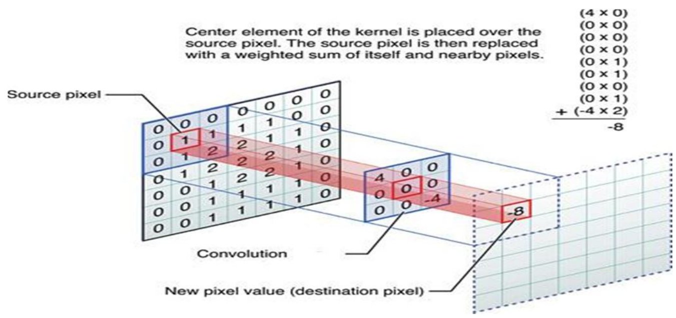


Fig. 1 Example of a 2D Discrete Convolution

Following the procedure in Fig. 1, we are always bound to leave a border on the perimeter of our image. For a kernel of size k, where k is odd, a border of $\frac{k-1}{2}$ pixels will be left on each side. We may consider padding or ignoring the border altogether, since these bordering pixels would not affect our calculations in any way.

IV. IDENTITY FILTER

An identity filter of dimension k (k odd) will be a $k \times k$ matrix with a 1 at the $\left[\frac{k+1}{2}, \frac{k+1}{2}\right]$ position. For example, an identity filter of size 3 would be $\begin{bmatrix} 0 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix}$. Since all elements are zero except a 1 at the middle, and this is the pixel aligned with every pixel during convolution (Fig. 1), we would get back the exact same image when convolved with this filter.

Note: In all future plottings of a kernel, each dark cell is a value of 0 and each light cell is a value greater than 0.

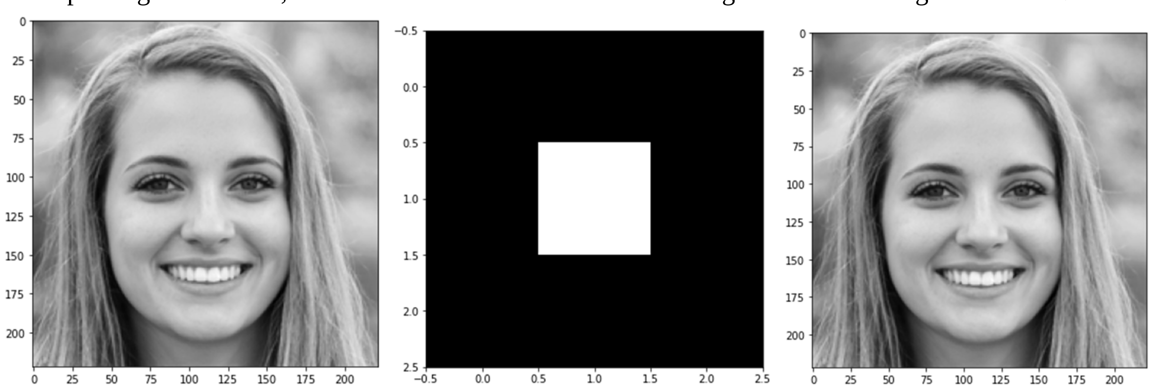


Fig. 2 (a) Original 224×224 Image (b) Kernel of size 3 with 1 at the centre and other elements 0 (c) Image after convolution with the kernel

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VII July 2022- Available at www.ijraset.com

V. BLOCK FILTER

A block filter is used to blur or smoothen images. In such a filter, each element of the kernel is either 1, or is normalized by the sum of all entries, i.e., $\frac{1}{k^2}$ for a k-dimensional filter. A 3-dimensional block filter is $\begin{bmatrix} \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \\ \frac{1}{9} & \frac{1}{9} & \frac{1}{9} \end{bmatrix}$.

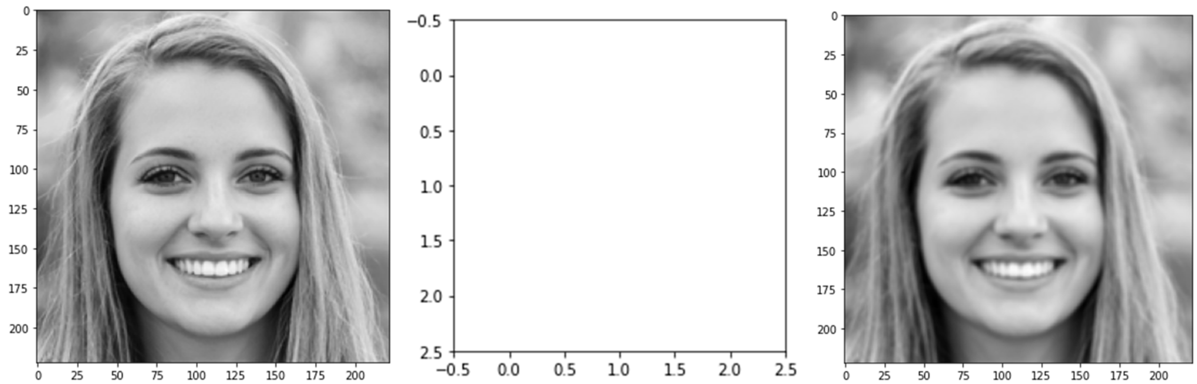


Fig. 3 (a) Original 224 × 224 Image (b) Kernel of size 3 with all elements $\frac{1}{9}$ (c) Smoothened Image after convolution with kernel

Notice that the blurring effect is not that visible, but it is certain that the image on the right has much smoother boundaries than the image on the left. Let us increase the filter size to 4 and 10 respectively in Fig. 4 and Fig. 5 to see the changes.

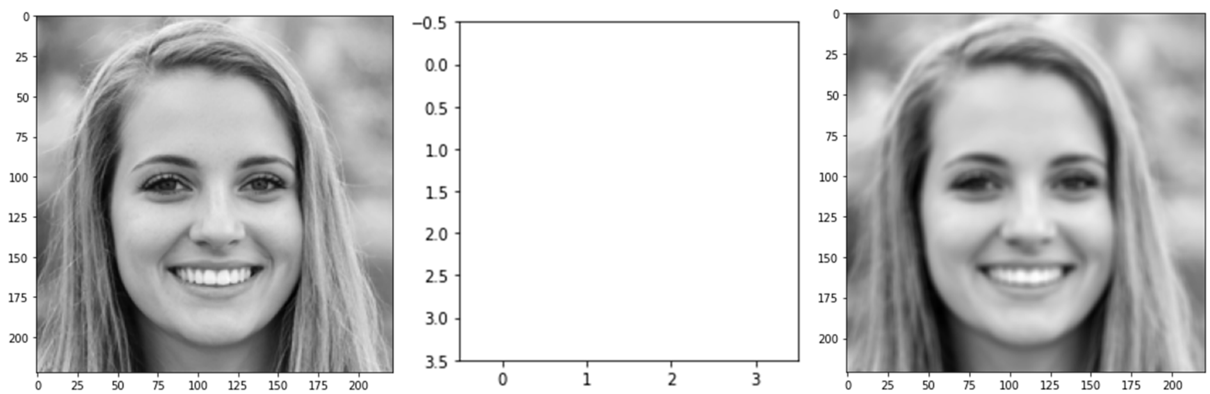


Fig. 4 (a) Original 224 × 224 image (b) Kernel of size 4 with all elements $\frac{1}{16}$ (c) Smoothened image after convolution with kernel

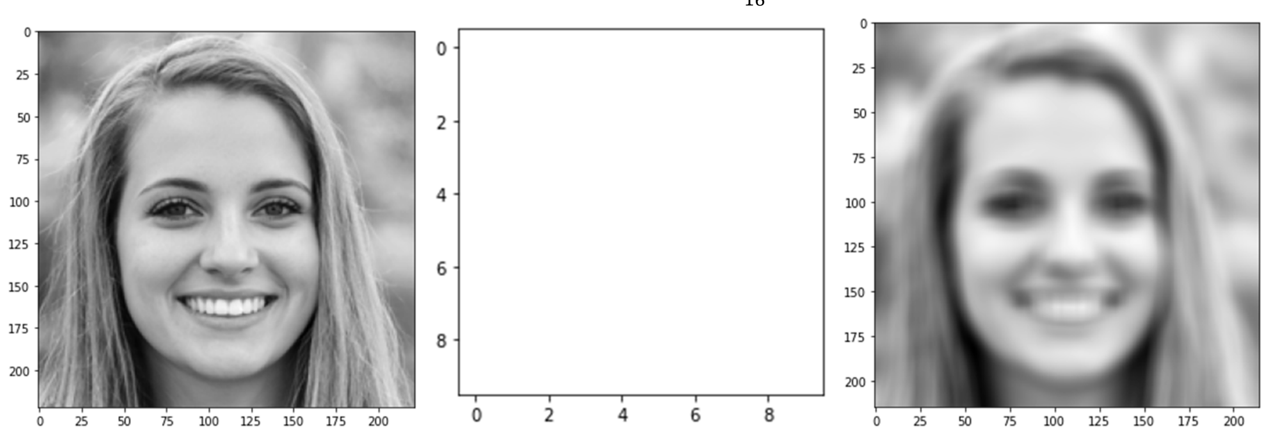


Fig. 5 (a) Original 224 × 224 image (b) Kernel of size 10 with all elements $\frac{1}{100}$ (c) Smoothened image after convolution with kernel

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 10 Issue VII July 2022- Available at www.ijraset.com

Dividing by the sum of all elements is just a normalization procedure to prevent over-saturation. If all elements had been 1, after convolution, each pixel would have contained a very high value and the image would have appeared washed out. Now, notice that as we smoothen the images, there are some square patterns that appear throughout the image, especially observable in Fig. 5 (c). That is because we have applied a block filter, and its block characteristic has been made prominent throughout the image. To avoid this, we use a Fuzzy Filter.

VI. FUZZY FILTER

A fuzzy filter is used to prevent the appearance of block like features that arise upon convolving with a block filter. In a fuzzy filter, the value at the centre is maximum, and they decrease as we move towards the edges and corners. A fuzzy filter is rotationally symmetric. A very good example of a Fuzzy Filter would be the Gaussian Kernel, n_σ. Each element of n_σ is given by

$$n_\sigma[i,j] = \frac{1}{2\pi\sigma^2} e^{-\frac{i^2+j^2}{2\sigma^2}} \qquad (8)$$

The larger the value of σ, the broader or more spread is the filter. For a given σ, it is a common and good practice to choose $k \approx 2\pi\sigma$, where k is the size of the kernel. The larger the value of sigma, the greater will be the blurring effect.

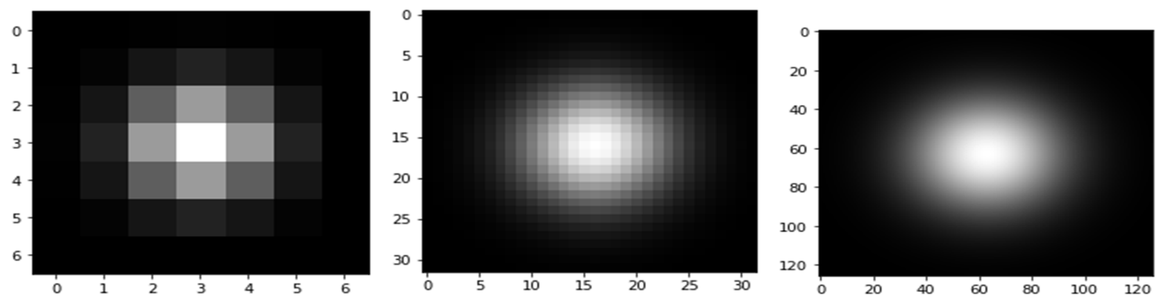


Fig. 6 Gaussian Kernels (a) $\sigma = 1$ (b) $\sigma = 5$ (c) $\sigma = 20$

Some results of convolution with Fuzzy (Gaussian) Kernels are given in the following figures.

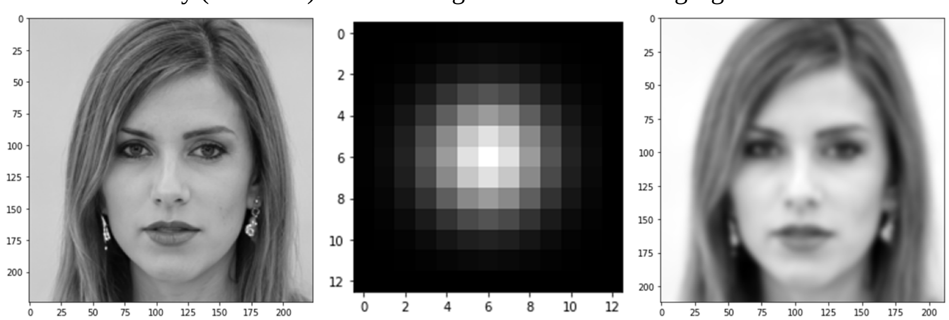


Fig. 7 (a) Original 224×224 Image (b) Gaussian Kernel of size 13 (c) Smoothened Image after convolution

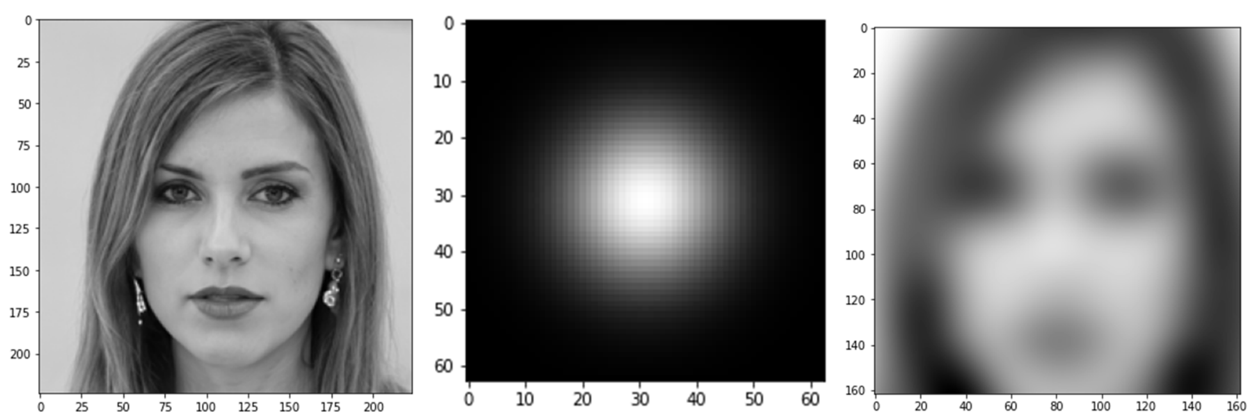


Fig. 8 (a) Original 224×224 image (b) Gaussian Kernel of size 63 (c) Smoothened Image after convolution

VII. UNI-CRISP FILTER AND SOBEL FILTER

Uni-crisp filter and Sobel filter are used to detect edges in an image, or in other words, sharpen an image. The Uni-crisp filter is $\begin{bmatrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{bmatrix}$ and the Sobel filter is $\begin{bmatrix} 1 & 2 & 1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix}$. The results of convolution are as follows:

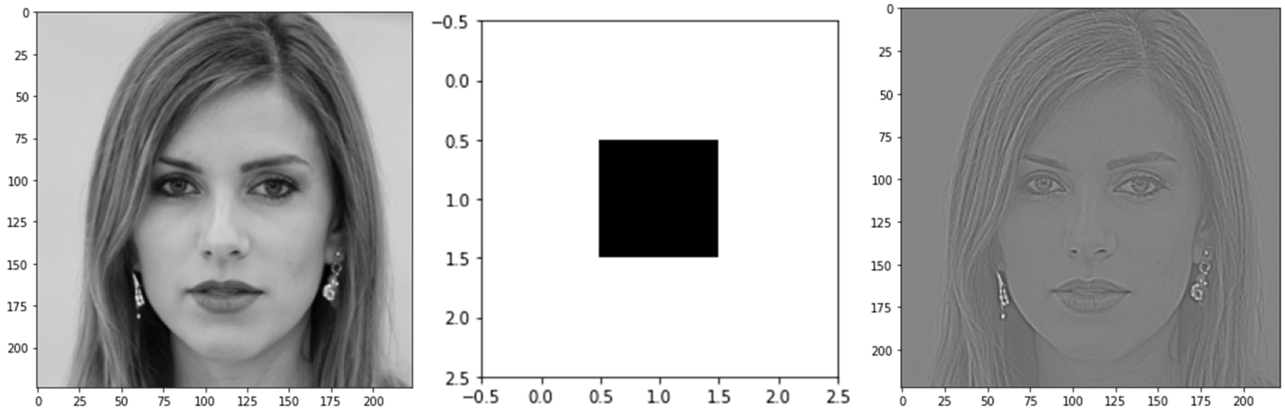


Fig. 9 (a) Original 224×224 Image (b) Uni-crisp Filter (c) Sharpened Image after convolution with Uni-crisp Filter

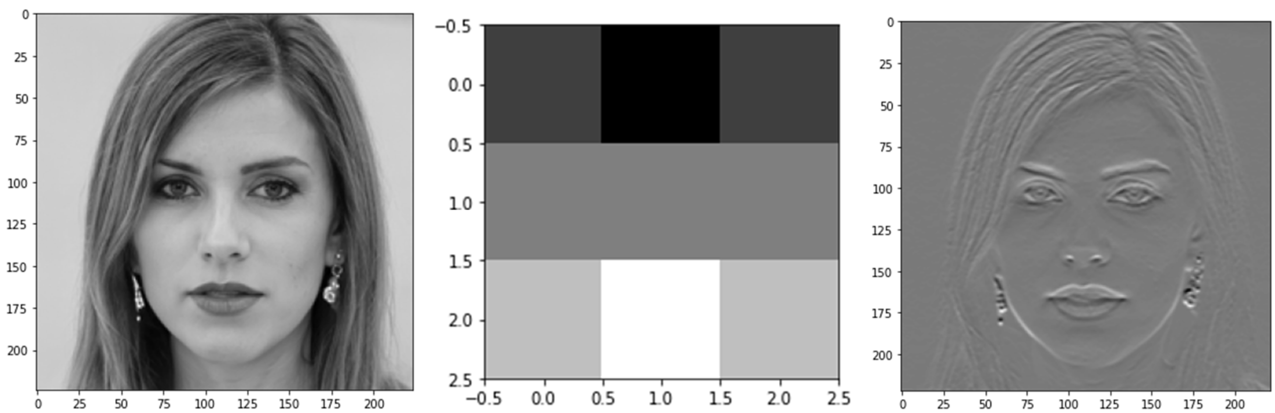


Fig. 10 (a) Original 224×224 Image (b) Sobel Filter (c) Sharpened Image after convolution with Sobel Filter

VIII. SALT-PEPPER NOISE AND MEDIAN FILTERING

Noise means random disturbance in a signal in a computer version. In our case, the signal is an image. Random disturbance in the brightness and colour of an image is called Image noise. As the name suggests, salt-pepper noise are random white dots in dark regions and black dots in light regions. This type of noise is found only in grayscale images.

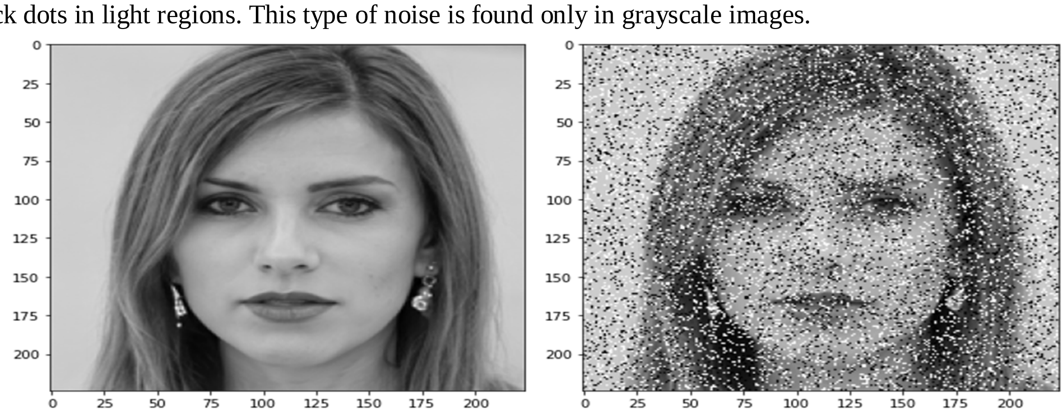


Fig. 11 (a) Original Image (b) Image with Salt-Pepper Noise

A common way to filter out the salt-pepper noise is to use the median filter. To create a median filter, place the $k \times k$ kernel as required during the above-mentioned convolution process. Sort the k^2 values in ascending order and assign the middle-most value to the pixel at the centre. It is important to note that the median filter is not a linear, or for that matter LSIS filter and hence, cannot be applied through the usual convolution process. It needs to be applied through the ordinary raster-scan process.

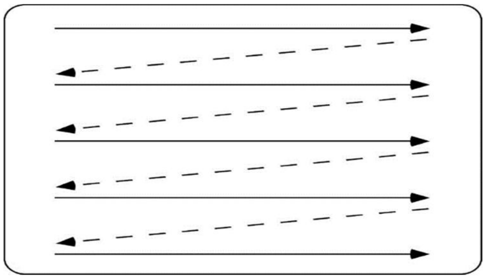


Fig. 12 Raster Scan

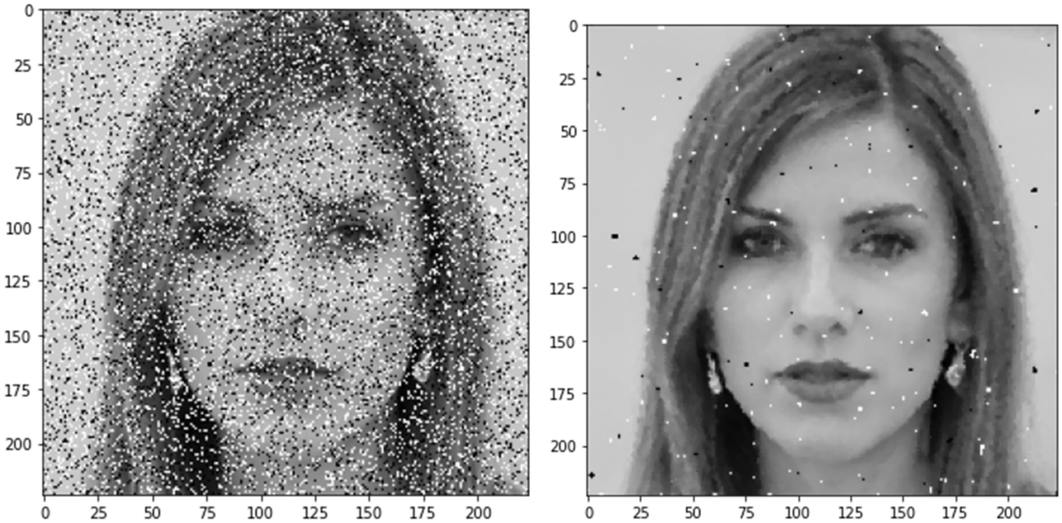


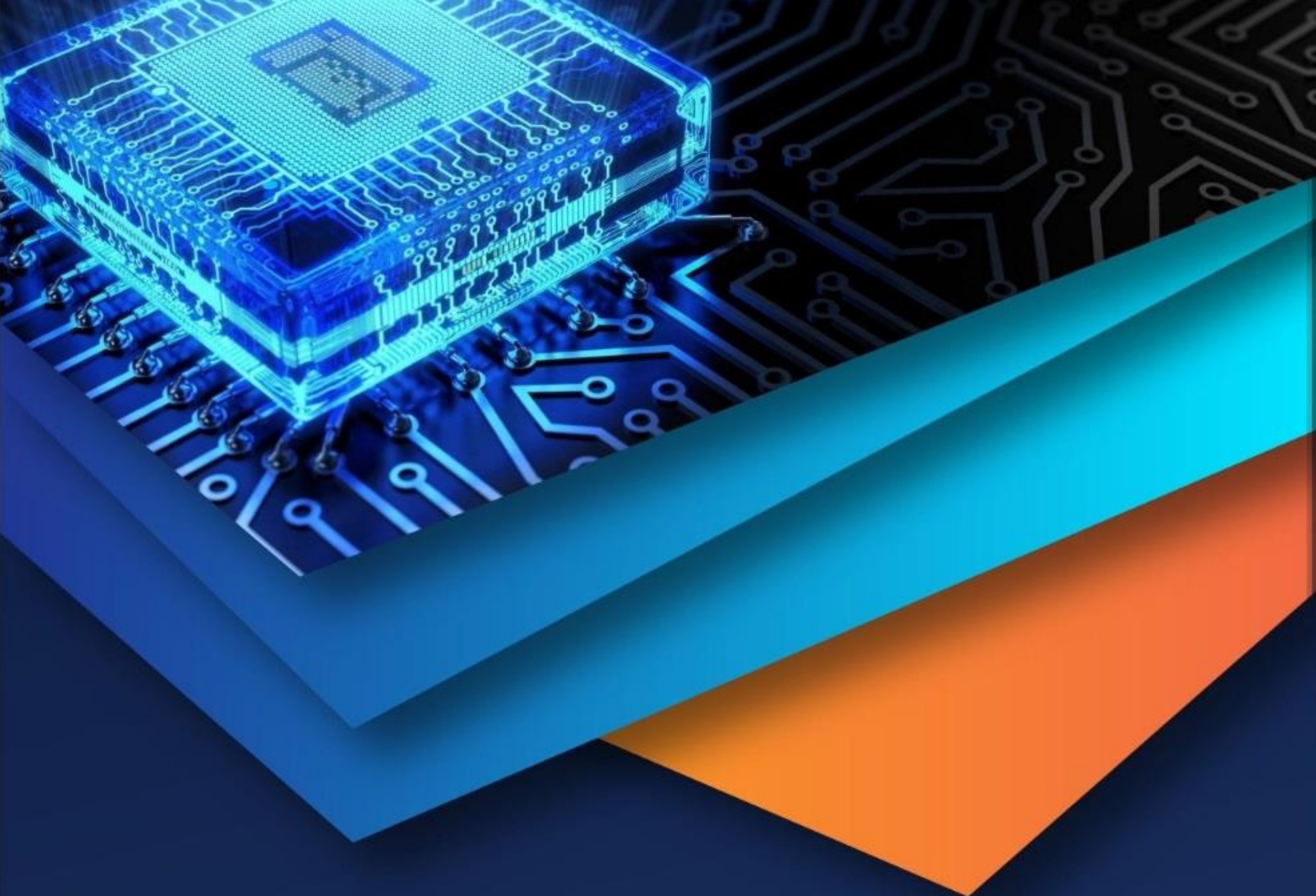
Fig. 13 (a) Image with Salt and Pepper Noise (b) Image filtered, using the Median Filter

IX. CONCLUSION

These are some of the crude techniques employed in computer vision. Of course, when you turn on the filtering option on your smartphone cameras, or the night-mode on your smartphone captures superb images even in extremely low-light conditions, there are much more sophisticated algorithms at work. But, all sophisticated algorithms at play in modern day devices stem from these algorithms, which make their study even more worthwhile.

REFERENCES

[1] Digital Image Processing, Gonzalez, R and Woods, R., Prentice Hall
[2] Computer Vision Algorithms and Applications, Szeliski, R., Springer
[3] R.A. Haddad and A.N. Akansu, "A Class of Fast Gaussian Binomial Filters for Speech and Image Processing," IEEE Transactions on Acoustics, Speech, and Signal Processing, vol. 39, pp 723–727, March 1991
[4] Diggle, P. J. (1985), "A kernel method for smoothing point process data", Journal of the Royal Statistical Society, Series C, 34 (2): 138–147
[5] Uludag, A. M. (1998), "On possible deterioration of smoothness under the operation of convolution", J. Math. Anal. Appl., 227 (2): 335–358
[6] Huang, Thomas S.; Yang, George J.; Tang, Gregory Y. (February 1979). "A fast two-dimensional median filtering algorithm" (PDF). IEEE Transactions on Acoustics, Speech, and Signal Processing. 27 (1): 13–18
[7] Arias-Castro, Ery; Donoho, David L. (June 2009). "Does median filtering truly preserve edges better than linear filtering?". Annals of Statistics. 37 (3): 1172–2009.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY