Gaussian Kernel

- Formula:

$$G(x,y) = \frac{1}{2\pi\sigma^2} e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Where:

- x and y are the distances from the center of the kernel,
- σ is the standard deviation, and
- $G(x,y)$ is the value of the Gaussian function at point (x,y).

- Sample kernel (5x5)

Kernel	Formatted Kernel
[0.00291502 0.0130642 0.02153923 0.0130642 0.00291502] [0.0130642 0.05854969 0.09653213 0.05854969 0.0130642] [0.02153923 0.09653213 0.15915457 0.09653213 0.02153923] [0.0130642 0.05854969 0.09653213 0.05854969 0.0130642] [0.00291502 0.0130642 0.02153923 0.0130642 0.00291502]]	[[1 4 7 4 1] [4 20 33 20 4] [7 33 54 33 7] [4 20 33 20 4] [1 4 7 4 1]]

- Input and Output:

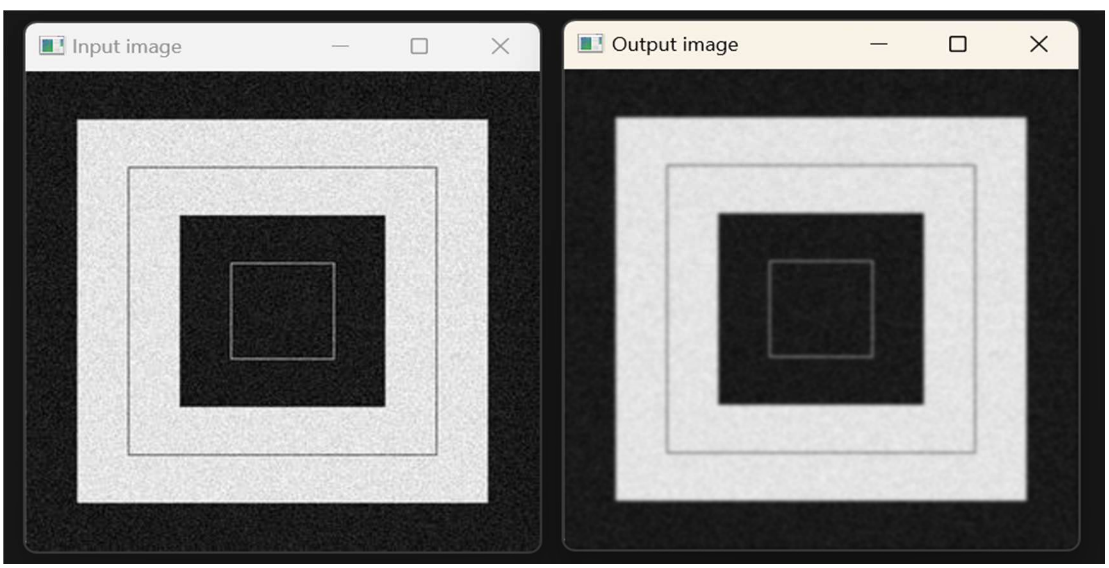

Fig-1: Gaussian (7x7) with sigmaX=1, sigmaY=1 and center=(3,3)

Mean Kernel

- It is a matrix having equal weight at each cell.
- Sample kernel (5x5)

Kernel	Formatted Kernel
[[0.04 0.04 0.04 0.04 0.04] [0.04 0.04 0.04 0.04 0.04] [0.04 0.04 0.04 0.04 0.04] [0.04 0.04 0.04 0.04 0.04] [0.04 0.04 0.04 0.04 0.04]]	[[1. 1. 1. 1. 1.] [1. 1. 1. 1. 1.] [1. 1. 1. 1. 1.] [1. 1. 1. 1. 1.] [1. 1. 1. 1. 1.]]

Table-2: Mean Kernel Table (5x5)

- Input and Output

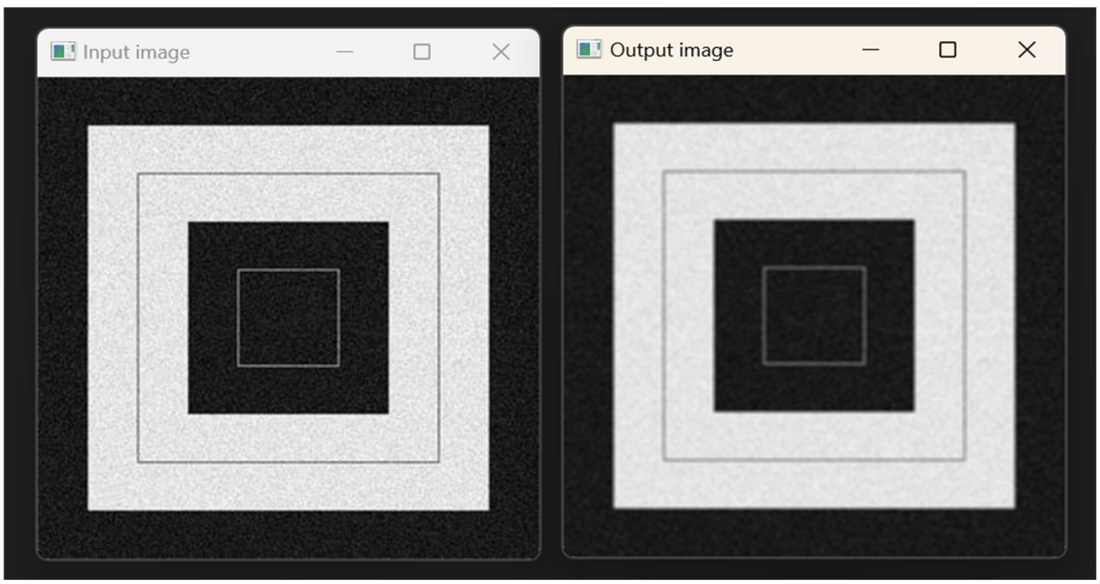

Fig-2: Mean kernel of size (3x3) and center=(1,1)

Laplacian kernel

- Formula

$$\Delta^2 f = \frac{\partial^2 f}{\partial x^2} + \frac{\partial^2 f}{\partial y^2}$$

- Sample kernel (3 x 3)

```
[[ 0 -1  0]
 [-1  4 -1]
 [ 0 -1  0]]
```

- Input and Output:

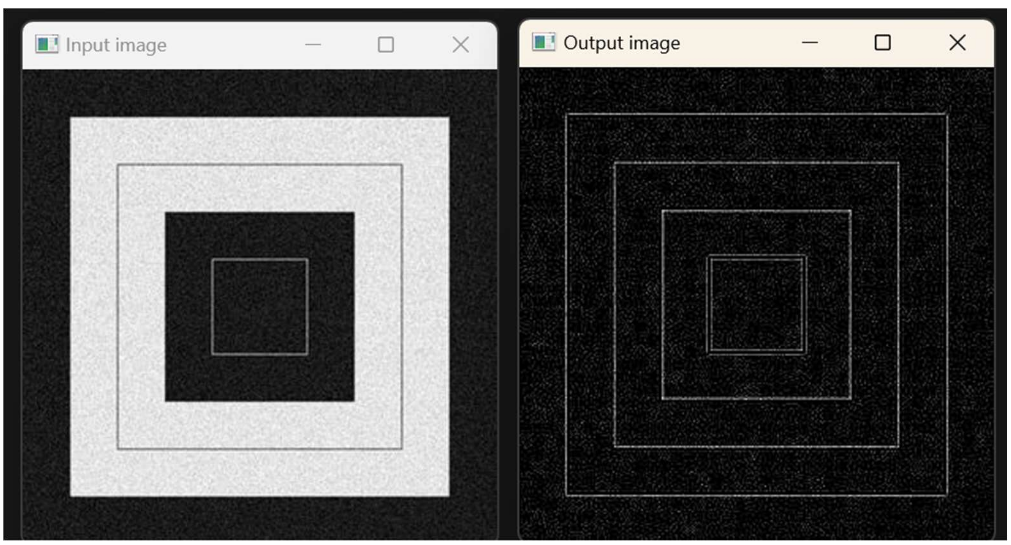

Fig-3: Laplacian kernel of size (3x3) and center=(1,1)

LOG Kernel:

- Formula

$$LoG(x,y) = \frac{1}{\pi\sigma^4}\left(1 - \frac{x^2+y^2}{2\sigma^2}\right) e^{-\frac{x^2+y^2}{2\sigma^2}}$$

Where:

- (x, y) are the spatial coordinates.
- σ is the standard deviation of the Gaussian function, controlling the amount of smoothing.
- $LoG(x, y)$ represents the value of the LoG kernel at coordinates (x, y).

- Sample kernel (9x9)

```
[[   1    4   12   21   25   21   12    4    1]
 [   4   17   41   59   63   59   41   17    4]
 [  12   41   66   37    3   37   66   41   12]
 [  21   59   37 -144 -282 -144   37   59   21]
 [  25   63    3 -282 -489 -282    3   63   25]
 [  21   59   37 -144 -282 -144   37   59   21]
 [  12   41   66   37    3   37   66   41   12]
 [   4   17   41   59   63   59   41   17    4]
 [   1    4   12   21   25   21   12    4    1]]
```

- Input and Output

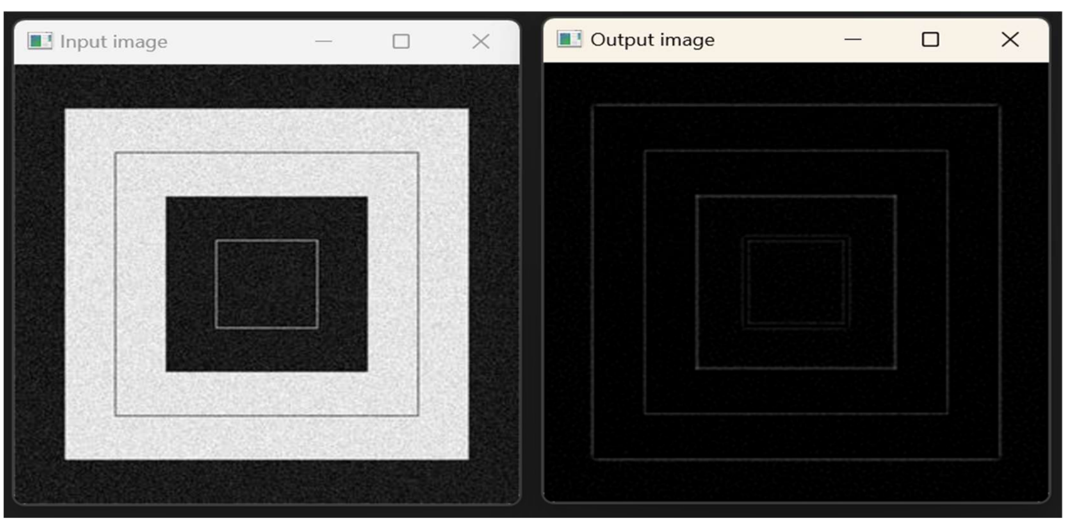

Fig-4: LoG kernel of size (9x9) and center=(4,4)

Sobel kernel

- It is used for detecting edges.
- Sample kernel (3x3)

<table>
<tr><th>Vertical Sobel Kernel</th><th>Horizontal Sobel kernel</th></tr>
<tr><td>$\begin{vmatrix} -1 & -2 & -1 \\ 0 & 0 & 0 \\ 1 & 2 & 1 \end{vmatrix}$</td><td>$\begin{vmatrix} -1 & 0 & 1 \\ -2 & 0 & 2 \\ -1 & 0 & 1 \end{vmatrix}$</td></tr>
</table>

- Two convolution is performed, then result is generated from them using

$$G = \sqrt{(G_x)^2 + (G_y)^2}$$

Where:

- G is the gradient magnitude.
- G_x is the horizontal gradient.
- G_y is the vertical gradient.

- Input and Output

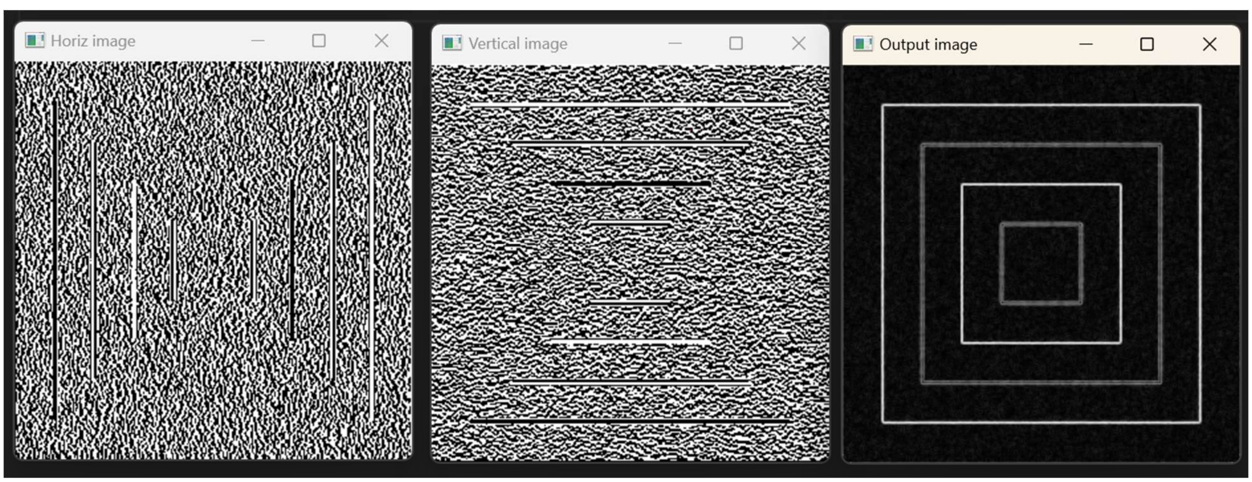

Fig-5: Two Sobel kernel of size (3x3) and center=(1,1)

Operation Type: HSV & RGB Difference

- Using (7x7) Gaussian kernel, sigmaX=1, sigmaY=1, center = (3,3)

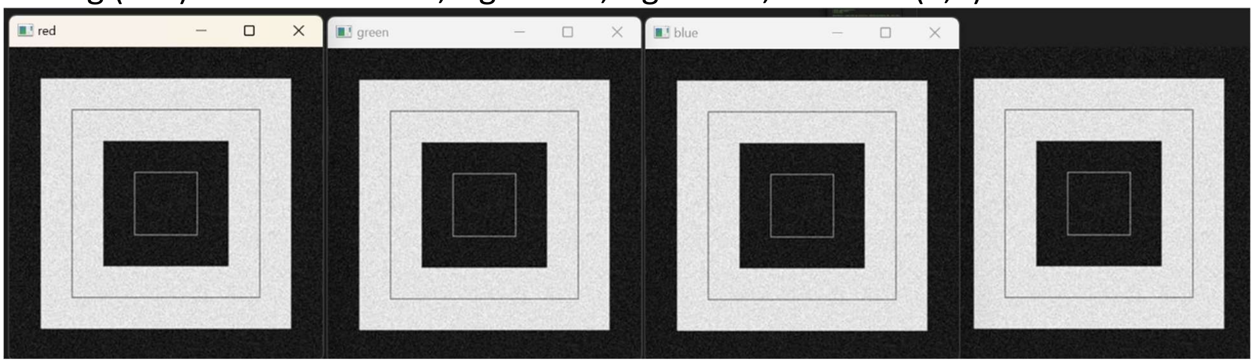

Fig-6: red, green, blue channel and input

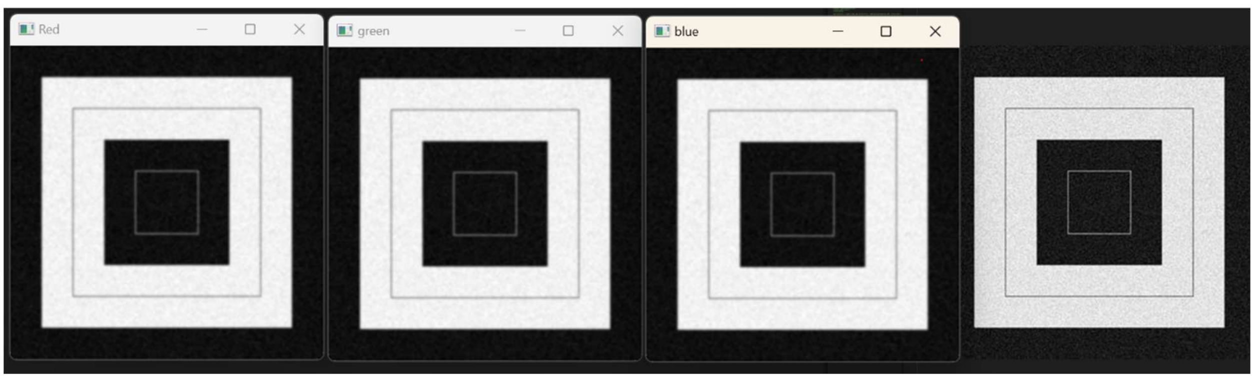

Fig-7: Convoluted red, green, blue channel and input

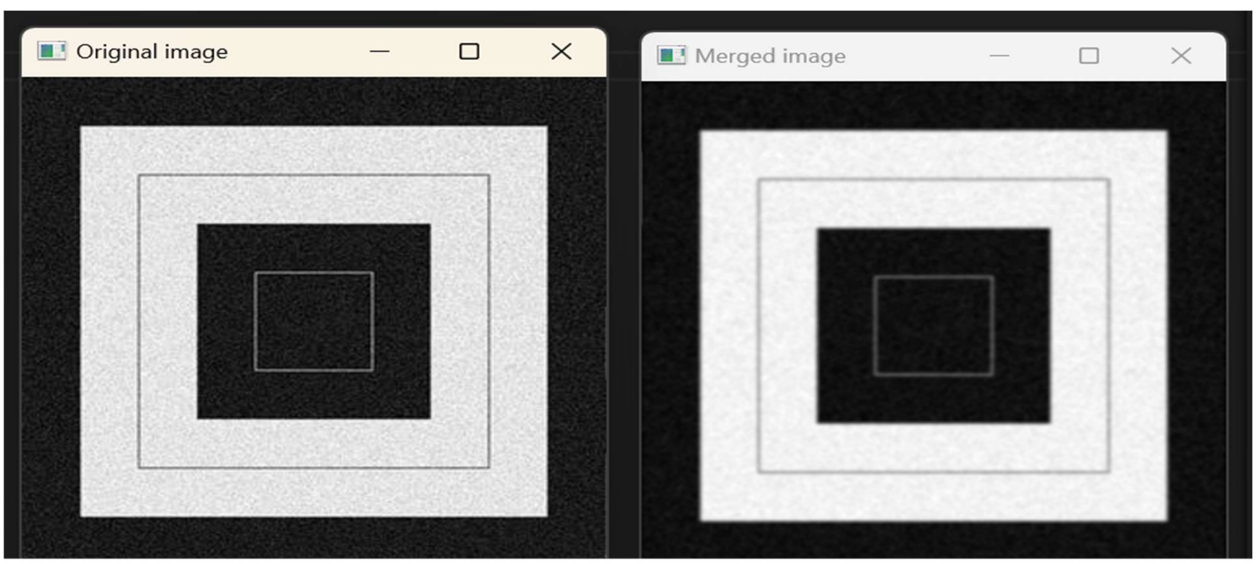

Fig-8: Original and merged convoluted image

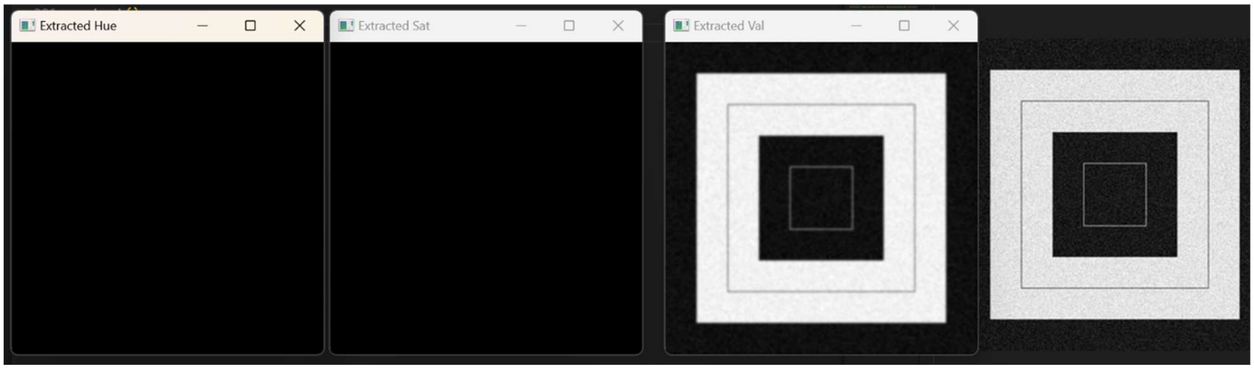

Fig-9: h, s, v channel with input

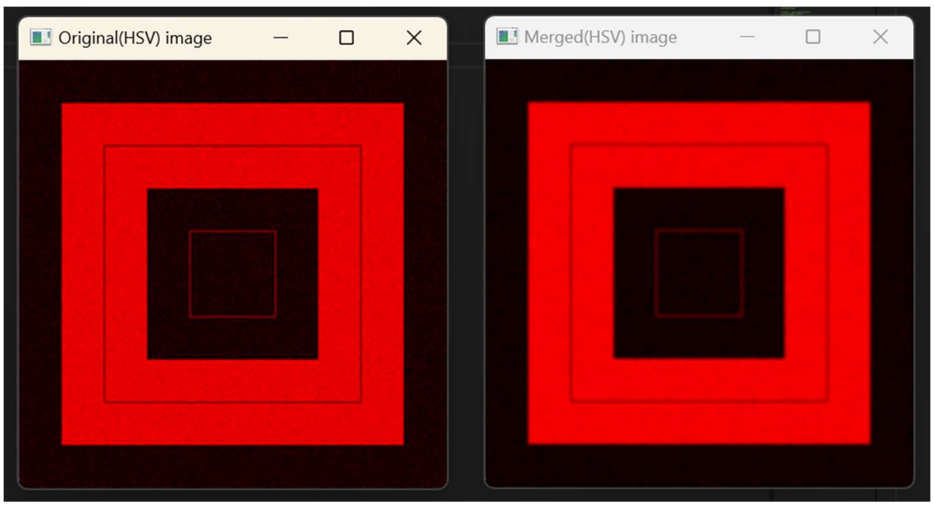

Fig-10: Original and merged convoluted HSV

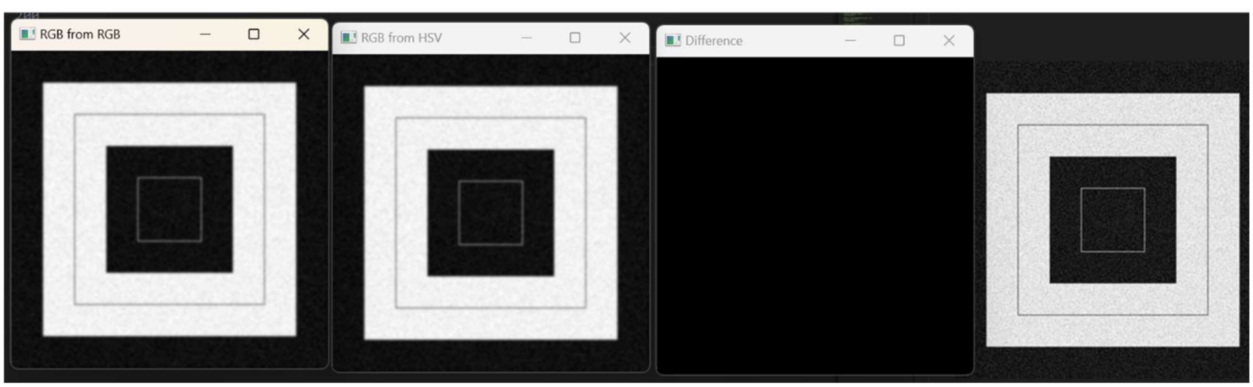

Fig-11: Difference between two convoluted images

- **RGB and HSV difference using Mean kernel of (3x3) center=(1,1)**

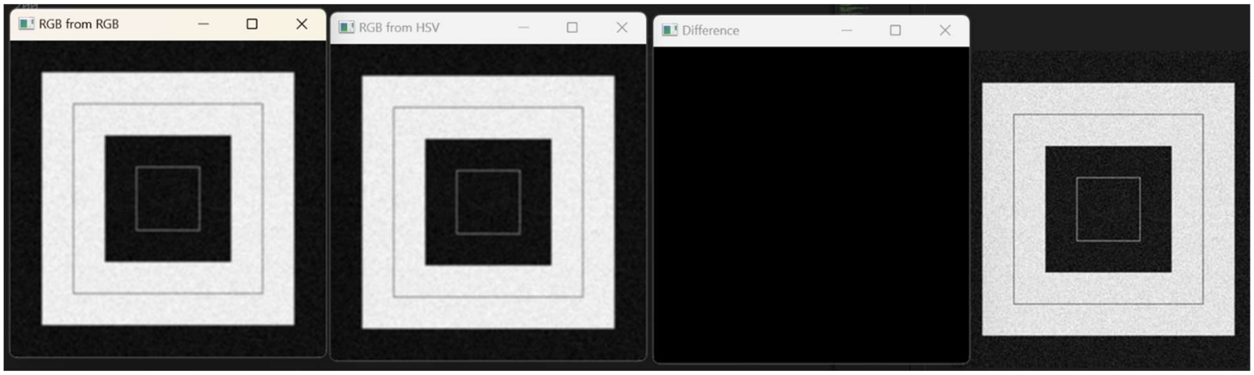

Fig-12: Difference between two convoluted images

- RGB and HSV difference using Laplacian kernel of (3x3) center=negative

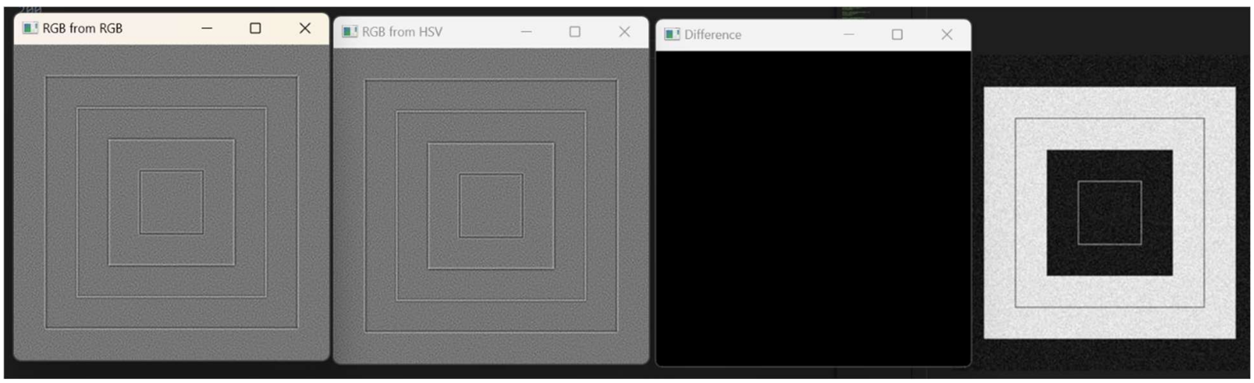

Fig-13: Difference between two convoluted images

- RGB and HSV difference using Log kernel of (3x3) sigma=1, center = (1,1)

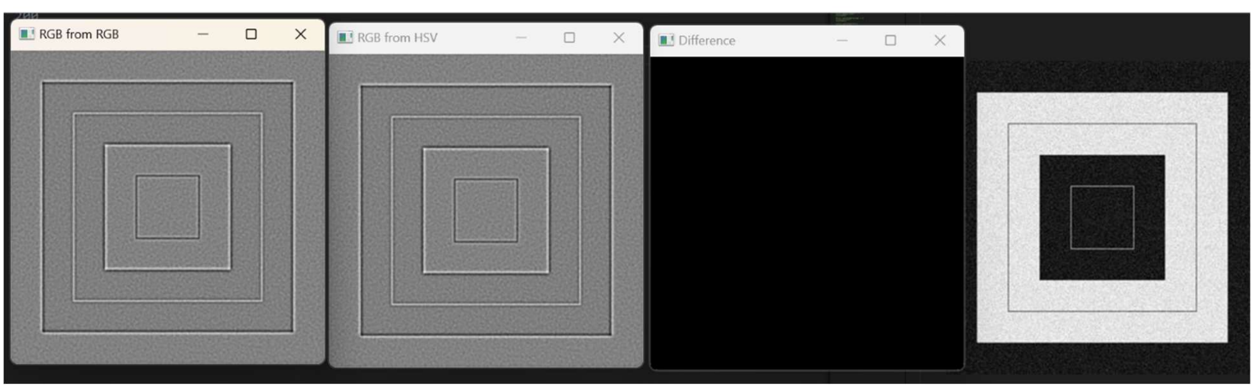

Fig-14: Difference between two convoluted images

- RGB and HSV difference using Soble kernel of (3x3)

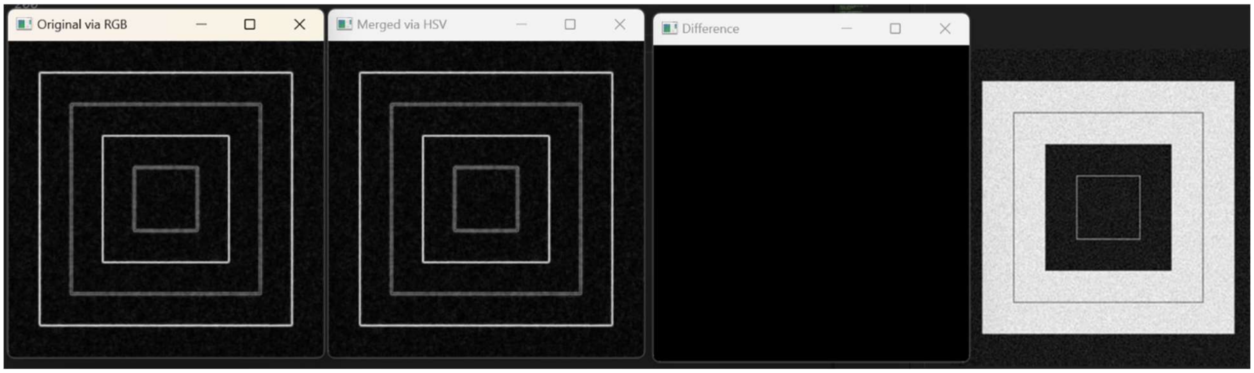

Fig-15: Difference between two convoluted images

- RGB and HSV difference sing (7x7) Gaussian kernel, sigmaX=1, sigmaY=1, center = (3,3)

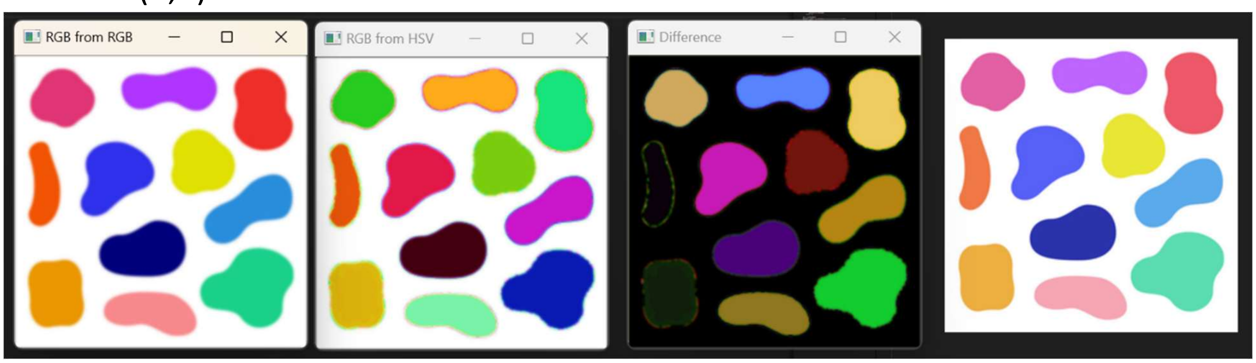

Fig-16: Difference between two convoluted image

- Edge detection using Sobel