# Shri Ramdeobaba College of Engineering and Management
# Nagpur, 440013

## Department of Computer Science Engineering

### FDVIP Lab

**Name** : *Shantanu Mane*
**Roll No.** : *E63*
**Batch** : *CSE-AIML*
**Date** : *29/3/2023*

### AIM - To study and perform concepts of filtering and applying the following filters on an image.

1. Low Pass Filter
2. High Pass Filter
3. Median Filter

## Importing Dependencies

```
import cv2
import matplotlib.pyplot as plt
import numpy as np
```

## Reading the images

```
image = cv2.imread("../data/mri_2.png", 0)
```

## 1. Low Pass Filter

### Creating Kernel

```
kernel_3x3 = np.ones((3, 3), np.float32) / 9

kernel_5x5 = np.ones((5, 5), np.float32) / 25
```

### 3x3 Kernel

```
import cv2

low_pass_3x3 = cv2.filter2D(image, -1, kernel_3x3)

cv2.imshow("3x3 Kernel", low_pass_3x3)
```

### 5x5 Kernel

```
low_pass_5x5 = cv2.filter2D(image, -1, kernel_5x5)

cv2.imshow("5x5 Kernel", low_pass_5x5)
```

### Output

| 3x3 Kernel | 5x5 Kernel |
| --- | --- |
|  |  |

## 2. High Pass Filter

### Creating Kernel

```
kernel_3x3 = np.array([
    [0.0, -1.0, 0.0],
    [-1.0, 4.0, -1.0],
```

```
    [0.0, -1.0, 0.0],
])
kernel_3x3 = kernel_3x3 / (np.sum(kernel_3x3) if np.sum(kernel_3x3) != 0 else 1)

kernel_5x5 = np.array([
    [0.0, 0.0, -1.0, 0.0, 0.0],
    [0.0, -1.0, -2.0, -1.0, 0.0],
    [-1.0, -2.0, 16.0, -2.0, -1.0],
    [0.0, -1.0, -2.0, -1.0, 0.0],
    [0.0, 0.0, -1.0, 0.0, 0.0],
])

kernel_5x5 = kernel_5x5 / (np.sum(kernel_5x5) if np.sum(kernel_5x5) != 0 else 1)

kernel_5x5_2 = np.array([
    [-1.0, -1.0, -1.0, -1.0, -1.0],
    [-1.0, 1.0, 2.0, 1.0, -1.0],
    [-1.0, 2.0, 4.0, 2.0, -1.0],
    [-1.0, 1.0, 2.0, 1.0, -1.0],
    [-1.0, -1.0, -1.0, -1.0, -1.0],
])

kernel_5x5_2 = kernel_5x5_2 / (np.sum(kernel_5x5_2) if np.sum(kernel_5x5_2) != 0 else 1)
```

## 3x3 Kernel

```
high_pass_3x3 = cv2.filter2D(image, -1, kernel_3x3)

cv2.imshow("3x3 Kernel", high_pass_3x3)
```

## 5x5 Kernel

```
high_pass_5x5 = cv2.filter2D(image, -1, kernel_5x5)

cv2.imshow("5x5 Kernel", high_pass_5x5)
```

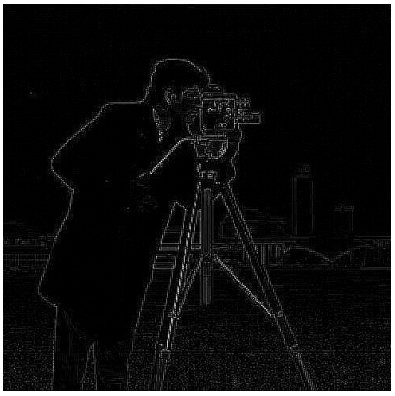
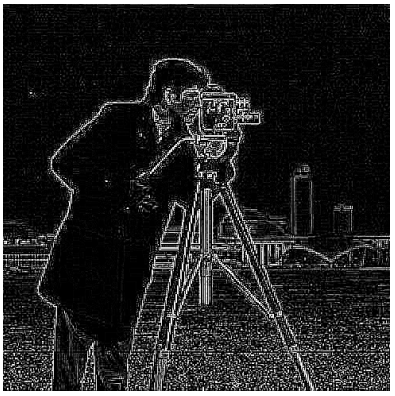
## 5x5 Kernel 2

```
high_pass_5x5_2 = cv2.filter2D(image, -1, kernel_5x5_2)

cv2.imshow("5x5 Kernel 2", high_pass_5x5_2)
```

## Output

| 3x3 Kernel | 5x5 Kernel | 5x5 Kernel 2 |
| --- | --- | --- |

| 3x3 Kernel | 5x5 Kernel | 5x5 Kernel 2 |
| --- | --- | --- |
|  |  |  |

# 3. Median Filter

## 3x3 Kernel

```
for i in range(image.shape[0] - 1):
    for j in range(image.shape[1] - 1):
        temp = [image[i - 1, j - 1], image[i - 1, j], image[i - 1, j + 1], image[i, j - 1],
                image[i, j + 1], image[i + 1, j - 1], image[i + 1, j], image[i + 1, j + 1]]

        temp.sort()
        filtered_image_3x3[i, j] = temp[4]

cv2.imshow("3x3 Kernel", filtered_image_3x3)
```

## Output