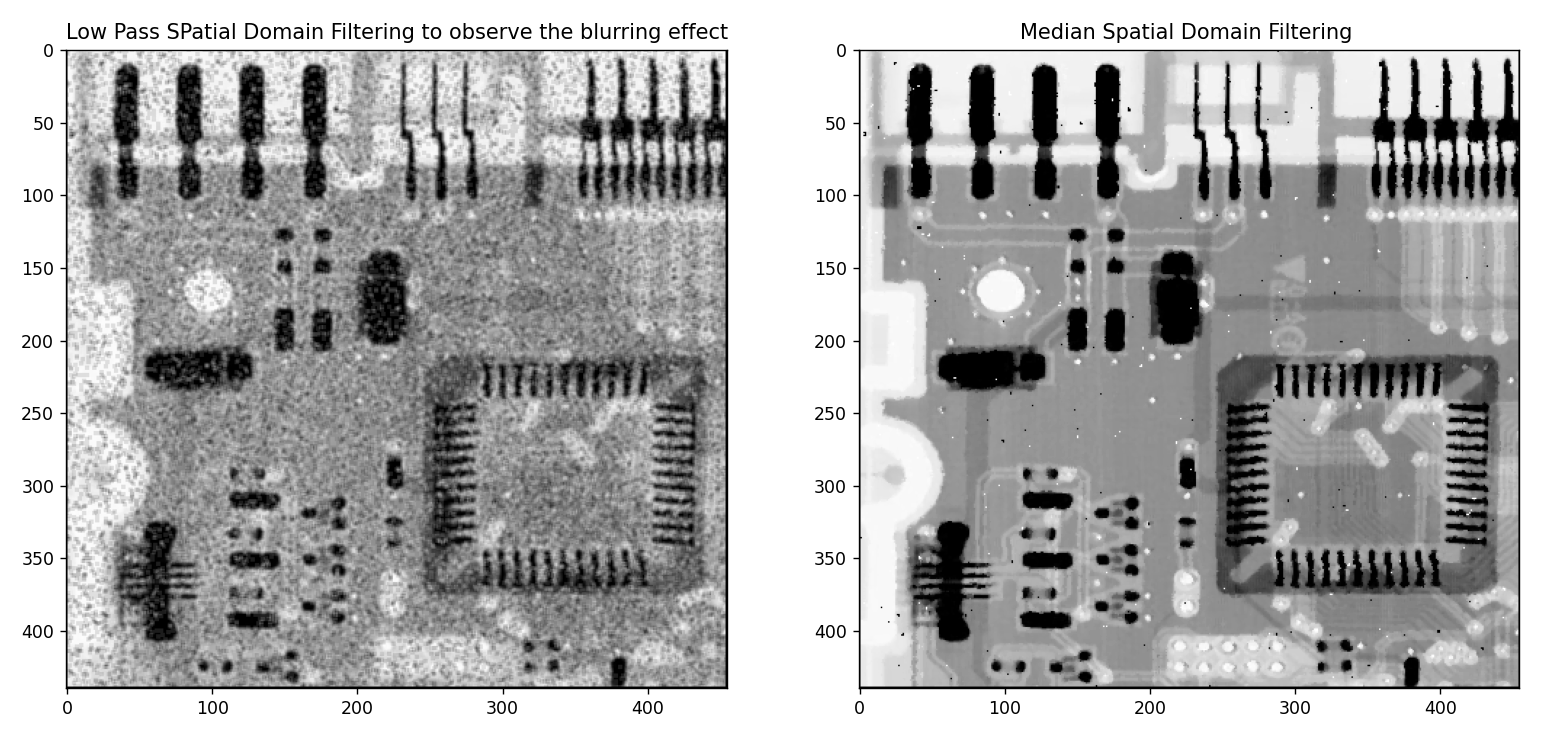
**Source Code :**   
  
import cv2   
import numpy as np  
import matplotlib.pyplot as plt  
  
# Read the image   
img = cv2.imread('noisysalterpepper.png', 0)   
  
m, n = img.shape   
  
# Develop Averaging filter(3, 3) mask   
mask = np.ones([3, 3], dtype = int)   
mask = mask / 9  
  
# Convolve the 3X3 mask over the image   
img\_new = np.zeros([m, n])   
  
for i in range(1, m-1):   
 for j in range(1, n-1):   
 temp = img[i-1, j-1]\*mask[0, 0]+img[i-1, j]\*mask[0, 1]+img[i-1, j + 1]\*mask[0, 2]+img[i, j-1]\*mask[1, 0]+ img[i, j]\*mask[1, 1]+img[i, j + 1]\*mask[1, 2]+img[i + 1, j-1]\*mask[2, 0]+img[i + 1, j]\*mask[2, 1]+img[i + 1, j + 1]\*mask[2, 2]   
   
 img\_new[i, j]= temp   
   
img\_new = img\_new.astype(np.uint8)   
  
# Read the image  
img\_noisy1 = cv2.imread('noisysalterpepper.png', 0)  
  
# Obtain the number of rows and columns  
# of the image  
m, n = img\_noisy1.shape  
  
# Traverse the image. For every 3X3 area,  
# find the median of the pixels and  
# replace the center pixel by the median  
img\_new1 = np.zeros([m, n])  
  
for i in range(1, m-1):  
 for j in range(1, n-1):  
 temp = [img\_noisy1[i-1, j-1],  
 img\_noisy1[i-1, j],  
 img\_noisy1[i-1, j + 1],  
 img\_noisy1[i, j-1],  
 img\_noisy1[i, j],  
 img\_noisy1[i, j + 1],  
 img\_noisy1[i + 1, j-1],  
 img\_noisy1[i + 1, j],  
 img\_noisy1[i + 1, j + 1]]  
   
 temp = sorted(temp)  
 img\_new1[i, j]= temp[4]  
  
img\_new1 = img\_new1.astype(np.uint8)  
plt.figure(figsize=(15, 8))  
plt.subplot(121), plt.imshow(img\_new, cmap='gray'), plt.title("Low Pass SPatial Domain Filtering to observe the blurring effect")  
plt.subplot(122), plt.imshow(img\_new1, cmap='gray'), plt.title("Median Spatial Domain Filtering ")  
plt.show()

**Output :   
**