Homework 3

 $\rm EE604$ - Image Processing Chirag Garg (210288)

August 28, 2024

Input

The image used for this assignment is shown below.



Figure 1: Input Image

Code

The following code demonstrates the application of Gaussian Highpass Filtering to the input image.

```
import numpy as np
  import cv2
  import matplotlib.pyplot as plt
3
  def gauss_dist_mat(shape, freq):
5
      m, n = shape
6
      cy, cx = m // 2, n // 2
                                   # center coordinates
      x = np.arange(n)
8
      y = np.arange(m)
      X, Y = np.meshgrid(x, y)
       dist_mat = np.sqrt((X - cx)**2 + (Y - cy)**2) # distance matrix
12
       gauss_mat = 1 - np.exp(- (dist_mat**2) / (2 * (freq**2))) # gaussian matrix
14
      return gauss_mat
  def highpass_func(img, freq):
17
       # Fourier Transform
18
       f_transform = np.fft.fft2(img)
19
       f_transform_shift = np.fft.fftshift(f_transform)
20
21
       # Gaussian Highpass Filter matrix
       hpass_filt = gauss_dist_mat(img.shape, freq)
       # Apply highpass filter
       filt_f_transform= f_transform_shift * hpass_filt
26
       # Inverse Fourier Transform
28
       f_invshift = np.fft.ifftshift(filt_f_transform)
29
       final_img = np.fft.ifft2(f_invshift)
30
       final_img = np.abs(final_img)
32
      return img, final_img, hpass_filt
33
34
  # Load image in grayscale
35
  image = cv2.imread('img.jpg', cv2.IMREAD_GRAYSCALE)
36
37
  # Cutoff frequencies
38
  cutoff_frequency = [20, 40, 80]
39
40
  for freq in cutoff_frequency:
41
       original_img, filtered_img, highpass_filter = highpass_func(image, freq)
42
43
       # Plots
44
       plt.figure(figsize=(18, 6))
45
       plt.suptitle(f'Highpass Filtering with Cutoff Frequency = {freq}', fontsize
46
          47
       plt.subplot(1, 3, 1)
48
       plt.imshow(original_img, cmap='gray')
49
       plt.title('Original Grayscale Image', fontsize=12)
      plt.axis('off')
       plt.subplot(1, 3, 2)
      plt.imshow(highpass_filter, cmap='gray')
54
      plt.title('Gaussian Highpass Filter')
      plt.axis('off')
56
57
      plt.subplot(1, 3, 3)
58
       plt.imshow(filtered_img, cmap='gray')
59
```

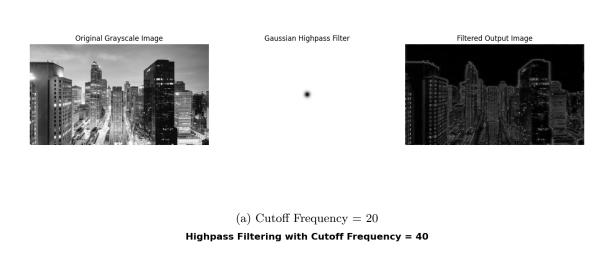
```
plt.title('Filtered Output Image', fontsize=12)
plt.axis('off')

plt.tight_layout(pad=2, rect=[0, 0, 1, 0.95])
plt.show()
```

Output

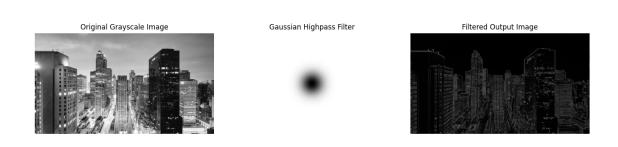
The results of applying Gaussian Highpass Filtering at different cutoff frequencies are shown below.

Highpass Filtering with Cutoff Frequency = 20





 $\label{eq:cutoff} \mbox{(b) Cutoff Frequency} = 40 \\ \mbox{Highpass Filtering with Cutoff Frequency} = \mbox{80}$



(c) Cutoff Frequency = 80

Figure 2: Filtered images with varying cutoff frequencies.