

Homework 3

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.

```
1 import numpy as np
2 import cv2
3 import matplotlib.pyplot as plt
4
5 def gauss_dist_mat(shape, freq):
6     m, n = shape
7     cy, cx = m // 2, n // 2      # center coordinates
8     x = np.arange(n)
9     y = np.arange(m)
10    X, Y = np.meshgrid(x, y)
11
12    dist_mat = np.sqrt((X - cx)**2 + (Y - cy)**2) # distance matrix
13    gauss_mat = 1 - np.exp(-(dist_mat**2) / (2 * (freq**2))) # gaussian matrix
14
15    return gauss_mat
16
17 def highpass_func(img, freq):
18     # Fourier Transform
19     f_transform = np.fft.fft2(img)
20     f_transform_shift = np.fft.fftshift(f_transform)
21
22     # Gaussian Highpass Filter matrix
23     hpass_filt = gauss_dist_mat(img.shape, freq)
24
25     # Apply highpass filter
26     filt_f_transform = f_transform_shift * hpass_filt
27
28     # Inverse Fourier Transform
29     f_invshift = np.fft.ifftshift(filt_f_transform)
30     final_img = np.fft.ifft2(f_invshift)
31     final_img = np.abs(final_img)
32
33     return img, final_img, hpass_filt
34
35 # Load image in grayscale
36 image = cv2.imread('img.jpg', cv2.IMREAD_GRAYSCALE)
37
38 # Cutoff frequencies
39 cutoff_frequency = [20, 40, 80]
40
41 for freq in cutoff_frequency:
42     original_img, filtered_img, highpass_filter = highpass_func(image, freq)
43
44     # Plots
45     plt.figure(figsize=(18, 6))
46     plt.suptitle(f'Highpass Filtering with Cutoff Frequency = {freq}', fontsize
47                  ↪ =16, weight='bold')
48
49     plt.subplot(1, 3, 1)
50     plt.imshow(original_img, cmap='gray')
51     plt.title('Original Grayscale Image', fontsize=12)
52     plt.axis('off')
53
54     plt.subplot(1, 3, 2)
55     plt.imshow(highpass_filter, cmap='gray')
56     plt.title('Gaussian Highpass Filter')
57     plt.axis('off')
58
59     plt.subplot(1, 3, 3)
60     plt.imshow(filtered_img, cmap='gray')
```

```
60 plt.title('Filtered Output Image', fontsize=12)
61 plt.axis('off')
62
63 plt.tight_layout(pad=2, rect=[0, 0, 1, 0.95])
64 plt.show()
```

Output

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

Highpass Filtering with Cutoff Frequency = 20



(a) Cutoff Frequency = 20

Highpass Filtering with Cutoff Frequency = 40



(b) Cutoff Frequency = 40

Highpass Filtering with Cutoff Frequency = 80



(c) Cutoff Frequency = 80

Figure 2: Filtered images with varying cutoff frequencies.