Homework 2

EE604 - Image Processing
Chirag Garg (210288)

August 15, 2024

Input

Below is the image used for this assignment.

Figure 1: Input image

The Python code uses libraries like numpy, matplotlib and opencv, and performs bit-plane slicing on a grayscale image and combines the bit-planes to create images with gradually increasing quality. The combined images are saved and displayed.

Code

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

img = cv2.imread('img.jpg', cv2.IMREAD_GRAYSCALE)

## bit-plane slicing
bit_planes = []
for i in range(8):
    plane = img%2           ## getting ith bit-value
    img //=2                ## removing last bit
    bit_planes.append(plane)

## reversing the order of list to get better experience as image improves by each adding bit-plane, from 7th bit to 0th bit
bit_planes.reverse()

## combining the images
combined_img = []
image = bit_planes[0]*(2**7)
combined_img.append(image)        ## scaling the image for more visibility
for i in range(1,8):
    image = image + bit_planes[i]*(2**(7-i))
    scaled_img = cv2.normalize(image, None,0,255,cv2.NORM_MINMAX)  ## scaling the image for more visibility
    combined_img.append(scaled_img)


## code for plotting the combined images

fig, axes = plt.subplots(4, 2, figsize=(12, 12))
axes = axes.ravel()
for idx, combined_image in enumerate(combined_img):
    axes[idx].imshow(combined_image, cmap='gray')
    axes[idx].set_title(f'Combined from Bit 7 to Bit {7-idx} ')
    axes[idx].axis('off')
plt.tight_layout()
plt.savefig('combined_images.png')
# plt.show()

h, w = img.shape
video = cv2.VideoWriter('210288.avi', cv2.VideoWriter_fourcc(*'XVID'), 1, (w, h), isColor=False)
```

```
41
42 for image in combined_img:
43     video.write(image)
44
45 video.release()
```

Output

The combined images are shown in the figure below:

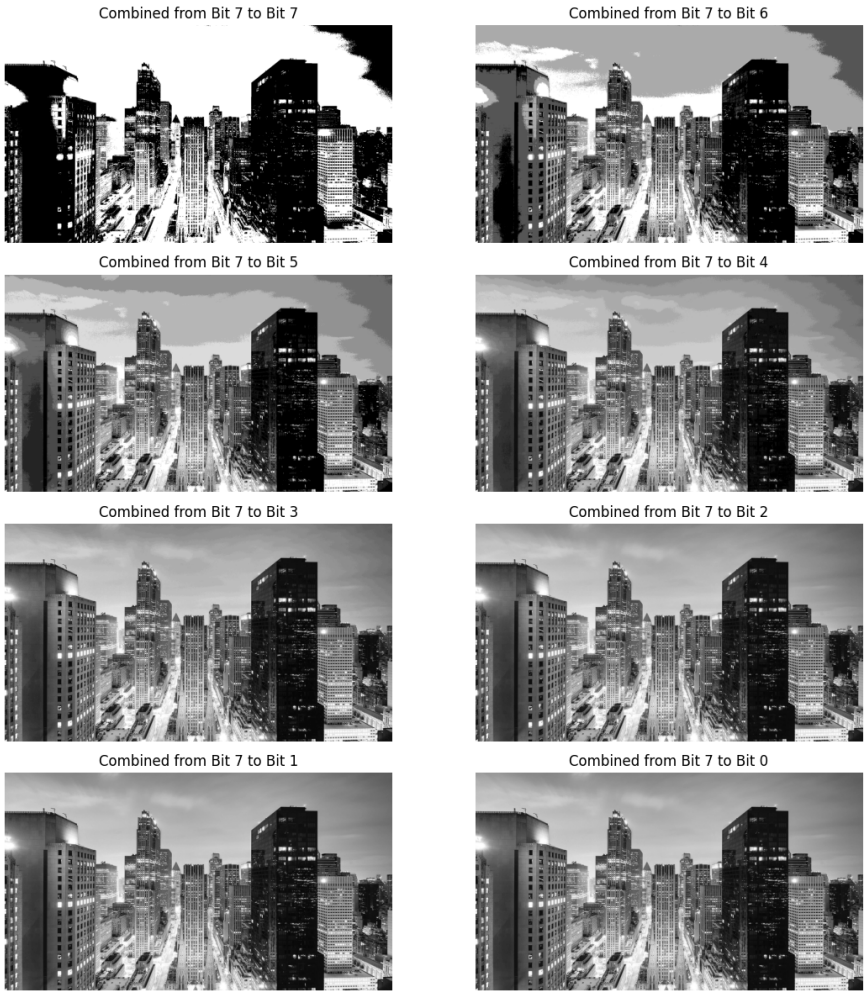

Figure 2: Combined images up to different bit-planes