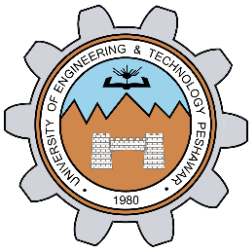
Assignment 1



Spring 2025

CSE-408 Digital Image Processing

Submitted by: **Suleman Shah**

Registration No.: **21PWCSE1983**

Class Section: **C**

Submitted to:

Engr. Mehran Ahmad

Date:

21st May 2025

Department of Computer Systems Engineering

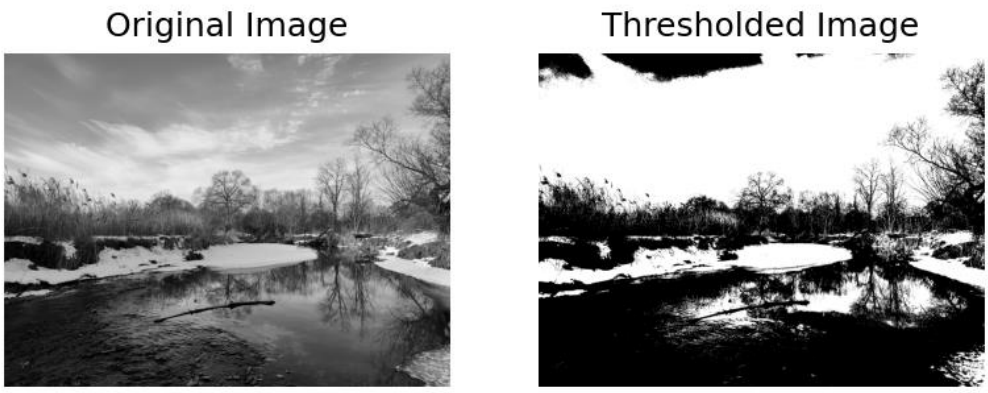
University of Engineering and Technology, Peshawar

Activity 1: Implement thresholding using MATLAB or Python. **Show the code and the output result** (original image and thresholded image).

Code:

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

img = cv2.imread('input.jpg', cv2.IMREAD_GRAYSCALE)
threshold = 128
_, thresh_img = cv2.threshold(img, threshold, 255, cv2.THRESH_BINARY)

plt.subplot(1, 2, 1)
plt.title('Original Image')
plt.imshow(img, cmap='gray')
plt.axis('off')

plt.subplot(1, 2, 2)
plt.title('Thresholded Image')
plt.imshow(thresh_img, cmap='gray')
plt.axis('off')

plt.show()
```

Output:

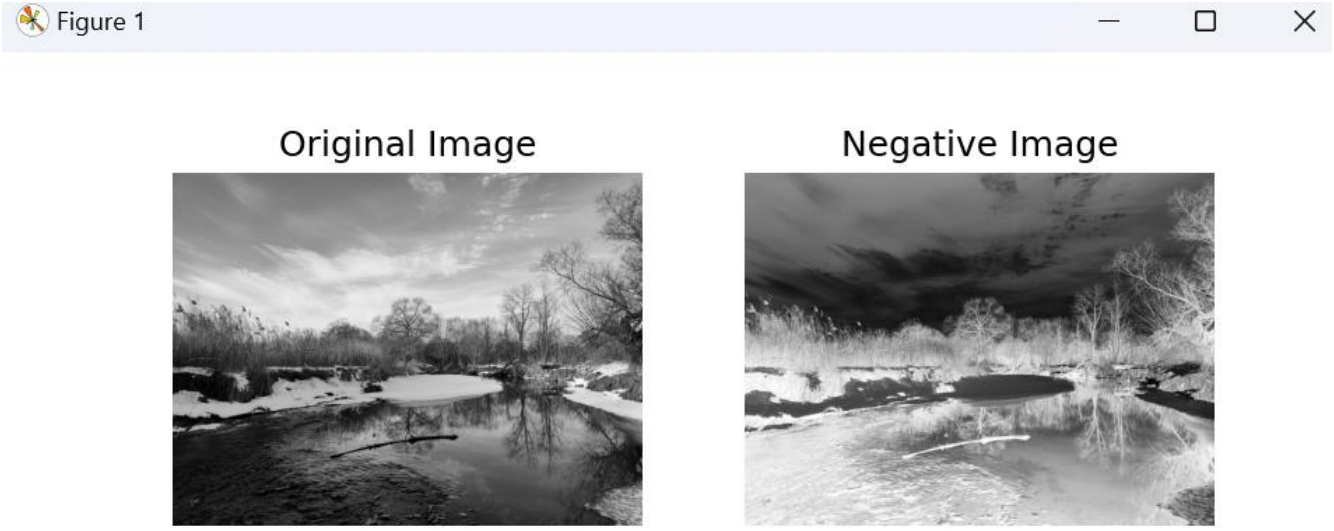


Activity 2: Write a MATLAB/Python script to perform negative transformation of an input image. **Show the code and the output result** (original and negative image, along with histograms).

Code:

```
import matplotlib.pyplot as plt

img = cv2.imread('input.jpg', cv2.IMREAD_GRAYSCALE)
neg_img = 255 - img
plt.subplot(2, 2, 1)
plt.title('Original Image')
plt.imshow(img, cmap='gray')
plt.axis('off')
plt.subplot(2, 2, 2)
plt.title('Negative Image')
plt.imshow(neg_img, cmap='gray')
plt.axis('off')
plt.subplot(2, 2, 3)
plt.title('Histogram (Original)')
plt.hist(img.ravel(), 256, [0, 256])
plt.subplot(2, 2, 4)
plt.title('Histogram (Negative)')
plt.hist(neg_img.ravel(), 256, [0, 256])
plt.show()
```

Output:



Activity 3: Implement a logarithmic transformation in MATLAB/Python. **Show the code and the output result** (original image and log-transformed image).

Code:

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

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

c = 255 / np.log(1 + np.max(img))
log_img = c * (np.log(1 + img))
log_img = np.array(log_img, dtype=np.uint8)

plt.subplot(1, 2, 1)
plt.title('Original Image')
plt.imshow(img, cmap='gray')
plt.axis('off')

plt.subplot(1, 2, 2)
plt.title('Log Transformed Image')
plt.imshow(log_img, cmap='gray')
plt.axis('off')

plt.show()
```

Output:

Activity 4: Write MATLAB code to apply a power-law transformation on an image using different values of *γ*. **Show the code and the output result** (original image and transformed images for various γ values).
Code:

```python
import cv2
import numpy as np
import matplotlib.pyplot as plt
img = cv2.imread('input.jpg', cv2.IMREAD_GRAYSCALE)
gamma_values = [0.5, 1.0, 2.0]
plt.figure(figsize=(10, 4))

for i, gamma in enumerate(gamma_values):
    gamma_img = np.array(255 * (img / 255) ** gamma, dtype='uint8')
    plt.subplot(1, len(gamma_values), i+1)
    plt.title(f'Gamma = {gamma}')
    plt.imshow(gamma_img, cmap='gray')
    plt.axis('off')
plt.show()
```

Output: