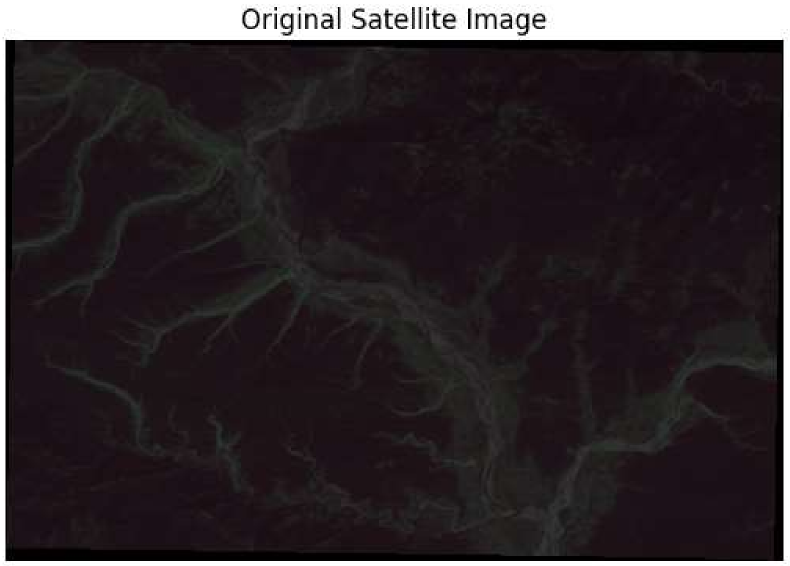
```
!pip install rasterio
```

```
Requirement already satisfied: rasterio in /usr/local/lib/python3.10/dist-packages (1.3.9)
Requirement already satisfied: affine in /usr/local/lib/python3.10/dist-packages (from rasterio) (2.4.0)
Requirement already satisfied: attrs in /usr/local/lib/python3.10/dist-packages (from rasterio) (23.2.0)
Requirement already satisfied: certifi in /usr/local/lib/python3.10/dist-packages (from rasterio) (2024.2.2)
Requirement already satisfied: click>=4.0 in /usr/local/lib/python3.10/dist-packages (from rasterio) (8.1.7)
Requirement already satisfied: cligj>=0.5 in /usr/local/lib/python3.10/dist-packages (from rasterio) (0.7.2)
Requirement already satisfied: numpy in /usr/local/lib/python3.10/dist-packages (from rasterio) (1.25.2)
Requirement already satisfied: snuggs>=1.4.1 in /usr/local/lib/python3.10/dist-packages (from rasterio) (1.4.7)
Requirement already satisfied: click-plugins in /usr/local/lib/python3.10/dist-packages (from rasterio) (1.1.1)
Requirement already satisfied: setuptools in /usr/local/lib/python3.10/dist-packages (from rasterio) (67.7.2)
Requirement already satisfied: pyparsing>=2.1.6 in /usr/local/lib/python3.10/dist-packages (from snuggs>=1.4.1->ra
```

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

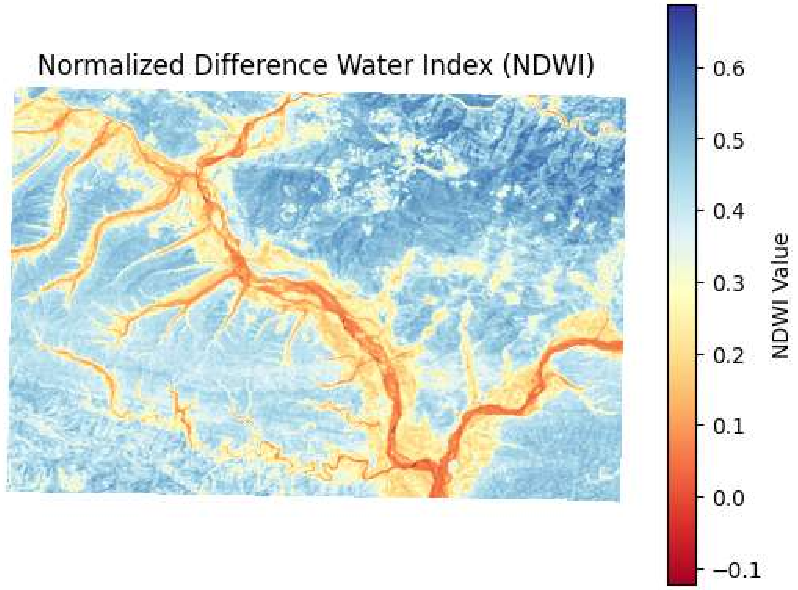
```
# StepStep 1: Read satellite imagery
with rasterio.open('X_test.tif') as src:
    # Read different bands from the satellite image
    nir_band = src.read(4)  # Near-infrared band (indexing starts from 1)
    green_band = src.read(3)  # Green band
    blue_band = src.read(2)  # Blue band
    red_band = src.read(1)  # Red band

plt.imshow(np.dstack((red_band, green_band, blue_band)))
plt.title('Original Satellite Image')
plt.axis('off')
```

```
(-0.5, 2508.5, 1676.5, -0.5)
```



```
# Step 2: Compute NDWI (Normalized Difference Water Index)
ndwi = np.where((nir_band + green_band) == 0., 0,
 (nir_band - green_band) / (nir_band + green_band))

plt.imshow(ndwi, cmap='RdYlBu')
plt.title('Normalized Difference Water Index (NDWI)')
plt.title('Normalized Difference Water Index (NDWI)')
plt.colorbar(label='NDWI Value')
plt.axis('off')
```

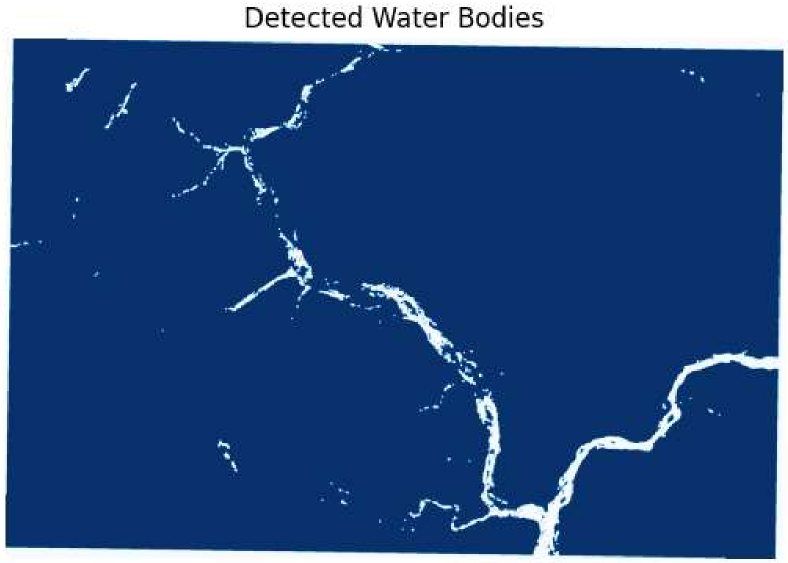
```
(-0.5, 2508.5, 1676.5, -0.5)
```



```
# Step 3: Thresholding
threshold = 0.1
water_mask = np.where(ndwi > threshold, 1, 0)
```

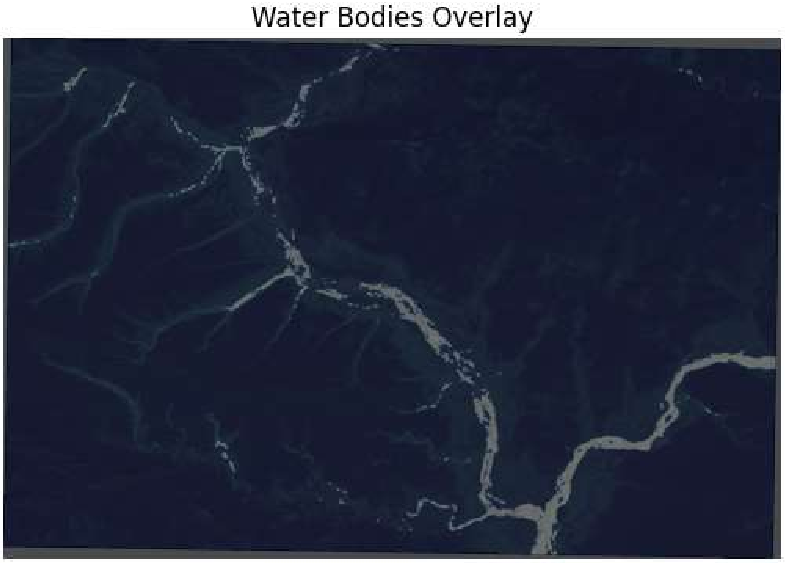
```
# Step 4: Post-processing (Optional)
kernel = np.ones((5, 5), np.uint8)
water_mask = cv2.morphologyEx(water_mask.astype(np.uint8),
                              cv2.MORPH_CLOSE, kernel)

plt.imshow(water_mask, cmap='Blues')
plt.title('Detected Water Bodies')
plt.axis('off')
```

```
(-0.5, 2508.5, 1676.5, -0.5)
```



```
#step 5:Overlay water mask with transparency
plt.imshow(np.dstack((red_band, green_band, blue_band)))
plt.imshow(water_mask, cmap='Blues', alpha=0.3)
plt.title('Water Bodies Overlay')
plt.axis('off')
```

```
(-0.5, 2508.5, 1676.5, -0.5)
```


Water Bodies Overlay