

EXPERIMENT NO. 2**Title:**

To perform Histogram Equalization.

Objective:

To enhance the contrast of a given grayscale image using the Histogram Equalization technique and analyze the results

Brief Theory:

Histogram Equalization is a technique used in image processing to improve the contrast of an image by spreading out the most frequent intensity values. It works by mapping the cumulative distribution function (CDF) of pixel intensities to a uniform distribution. This transformation enhances the visibility of details in an image.

Mathematical Explanation:

1. Compute the histogram of the original image.
2. Compute the cumulative sum of the histogram values.
3. Normalize the cumulative sum to get the transformation function.
4. Map the original intensity values using this function to obtain the enhanced image.

Histogram Equalization is widely used in applications such as medical imaging, remote sensing, and surveillance.

Steps: -**Step 1: Import Necessary Libraries**

First, we need to import the libraries required for Image Processing Essentials on Google Colab.

Code: -

```
import cv2
import numpy as np
import matplotlib.pyplot as plt
from google.colab.patches import cv2_imshow
```

Step 2: Read the Image

We will read the sample image from the local directory.

Code: -

```
img = cv2.imread('sample.jpg')
```

imread() reads the image from source.

Step 3: Convert the Sample image to grayscale image

Code: -

```
gray_img = cv2.cvtColor(img, cv2.COLOR_BGR2GRAY)
```

Step 4: Perform Histogram Equalization

Code: -

```
equalized_img = cv2.equalizeHist(gray_img)
```

Step 5: Plot the grayscale image and its histogram

Code: -

```
plt.figure(figsize=(12, 6))

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

plt.subplot(2, 2, 2)
plt.title("Histogram of Original Image")
plt.hist(img.ravel(), bins=256, range=(0, 256), color='blue')
plt.xlabel("Pixel Intensity")
plt.ylabel("Frequency")
```

Step 6: Plot the Equalized image and its histogram

Code: -

```
plt.subplot(2, 2, 3)
plt.title("Equalized Image")
plt.imshow(equalized_img, cmap='gray')
plt.axis('off')

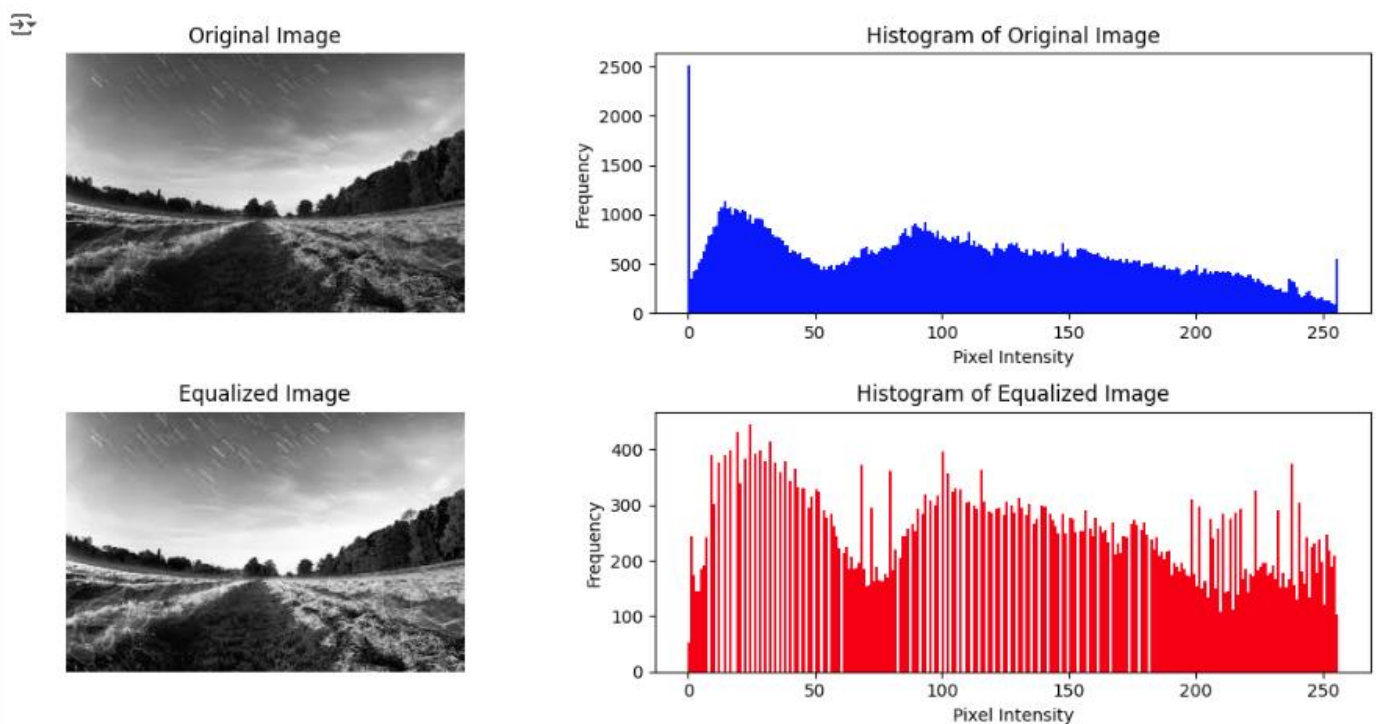
plt.subplot(2, 2, 4)
plt.title("Histogram of Equalized Image")
plt.hist(equalized_img.ravel(), bins=256, range=(0, 256),
color='red')
plt.xlabel("Pixel Intensity")
plt.ylabel("Frequency")
```

Step 7: Show the plots

Code: -

```
plt.tight_layout()
plt.show()
```

Output:



Explanation:

1. We import necessary libraries and read the image in grayscale mode.
2. The original image is displayed using Matplotlib.
3. The `cv2.equalizeHist()` function is applied to enhance contrast.
4. The equalized image is displayed alongside the original image.
5. The histograms before and after equalization are plotted to observe the effect of contrast enhancement.

Conclusion:

Histogram Equalization significantly improves the contrast of an image by redistributing intensity values. This method is useful in various applications where image clarity is crucial, such as medical imaging and object detection.

Practice Questions:

1. Apply Histogram Equalization on a colored image by converting it to YCrCb color space.
2. Implement Adaptive Histogram Equalization using OpenCV's `cv2.createCLAHE()` function.
3. Compare the results of Global and Adaptive Histogram Equalization.
4. Perform Histogram Equalization on an underexposed image and observe the changes.
5. Write a function to implement Histogram Equalization manually without using OpenCV's built-in function.

Expected Oral Questions

1. What is the purpose of Histogram Equalization?
2. How does Histogram Equalization affect an image's contrast?
3. What are the limitations of Histogram Equalization?
4. How does Adaptive Histogram Equalization differ from Global Histogram Equalization?
5. Why do we use the cumulative distribution function (CDF) in Histogram Equalization?
6. In which real-world applications is Histogram Equalization commonly used?

FAQs in Interviews

1. What is Histogram Equalization in image processing?

A: It is a contrast enhancement technique that redistributes intensity values to improve image clarity.

2. Why does Histogram Equalization work better on low-contrast images?

A: Because it spreads out intensity values, making details more distinguishable.

3. Can Histogram Equalization be applied to color images?

A: Yes, but it is typically applied to the luminance channel in YCrCb or LAB color spaces.

4. What is the drawback of Histogram Equalization?

A: It can amplify noise and may not work well for images with already high contrast.

5. How does Adaptive Histogram Equalization differ from standard Histogram Equalization?

A: Adaptive Histogram Equalization applies contrast enhancement to small regions instead of the entire image.

6. Can Histogram Equalization be reversed?

A: No, it is a non-reversible transformation as pixel intensity mapping changes permanently.