3 - 4.5. Contrast Manipulation

September 27, 2024

Jarrian Vince G. Gojar
Instructor I
College of Information and Communications Technology, Sorsogon State University, Philippines

1 Introduction

In an image, `contrast` refers to the difference in luminance or color that makes an object (or its representation in an image) distinguishable. In other words, contrast is the difference between the darkest and lightest areas of an image. The greater the difference, the higher the contrast. It is defined by the following formula:

$$\text{Contrast} = \frac{L_{\max} - L_{\min}}{L_{\max} + L_{\min}}$$

where:

- $L_{\max}$ is the maximum luminance value in the image.
- $L_{\min}$ is the minimum luminance value in the image.

Read More:

- [Contrast of an Image](#)

2 Setup

```
[ ]: %pip install opencv-python opencv-contrib-python numpy matplotlib
```

3 Initial Setup

```
[2]: # Import Libraries
import cv2
import numpy as np
import matplotlib.pyplot as plt

# Asset Root
asset_root = '../../assets/'

# Image Path
```

```
image_path = asset_root + '/images/hunting_lion.jpg'

# Read Image and convert to RGB
input_image = cv2.cvtColor(cv2.imread(image_path), cv2.COLOR_BGR2RGB)
gray_image = cv2.cvtColor(input_image, cv2.COLOR_RGB2GRAY)

# Display Both Image
plt.figure("Hunting Lion")

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

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

plt.show()
```


Original Image   Grayscale Image

4   Contrast Manipulation

Contrast Manipulation is the process of changing the contrast of an image. It can be done by changing the pixel values of the image. In image processing, contrast manipulation is used to enhance the visibility of an image by increasing the difference between the lightest and darkest areas of the image.

There are several ways to manipulate the contrast of an image. Some of the common methods are:

1. **Histogram Equalization**: It is a method used to improve the contrast of an image by redistributing the intensity values of the image. It is done by mapping the intensity values of the image to a new range of values.

2. **Contrast Stretching**: It is a method used to increase the contrast of an image by stretching the intensity values of the image to cover the full range of values.

3. **Gamma Correction**: It is a method used to adjust the contrast of an image by changing the intensity values of the image using a gamma correction function.

4. **Histogram Matching**: It is a method used to adjust the contrast of an image by matching the histogram of the image to a reference histogram.

In this book, we will focus on `Histogram Equalization` to manipulate the contrast of an image.

Read More:

- Contrast Manipulation
- Histogram Equalization
- Histogram Equalization in OpenCV
- Contrast Stretching
- Contrast Stretching in OpenCV
- Gamma Correction
- Histogram Matching

5 Histogram Equalization

`Histogram Equalization` is a method used to improve the contrast of an image by redistributing the intensity values of the image. It is done by mapping the intensity values of the image to a new range of values.

The histogram of an image represents the distribution of intensity values in the image. The histogram equalization process involves calculating the `cumulative distribution function (CDF)` of the histogram and then mapping the intensity values of the image to a new range of values based on the `CDF`.

It can be done using the following steps:

1. Calculate the histogram of the image.
2. Calculate the `cumulative distribution function (CDF)` of the histogram using the formula:

$$CDF(i) = \sum_{j=0}^{i} histogram(j)$$

3. Normalize the `CDF` to cover the full range of intensity values using the formula:

$$CDF_{normalized} = \frac{CDF - CDF_{min}}{CDF_{max} - CDF_{min}} \times 255$$

4. Map the intensity values of the image to the new range of values using the normalized `CDF`.

```python
[3]: # Histogram Equalization on a Color Image
     # Convert to YCrCb
     ycrcb_image = cv2.cvtColor(input_image, cv2.COLOR_RGB2YCrCb)

     # Equalize the histogram of the Y channel
     def equalize_hist(image):
         image_copy = np.copy(image)
         image_copy[:, :, 0] = cv2.equalizeHist(image_copy[:, :, 0])
         return image_copy

     # Convert the YCrCb image back to RGB format
     equalized_image = cv2.cvtColor(equalize_hist(ycrcb_image), cv2.COLOR_YCrCb2RGB)

     # Display Both Image
     plt.figure("Hunting Lion")

     plt.subplot(1, 2, 1)
     plt.imshow(input_image)
     plt.title("Original Image")
     plt.axis('off')

     plt.subplot(1, 2, 2)
     plt.imshow(equalized_image)
     plt.title("Equalized Image")
     plt.axis('off')

     plt.show()
```

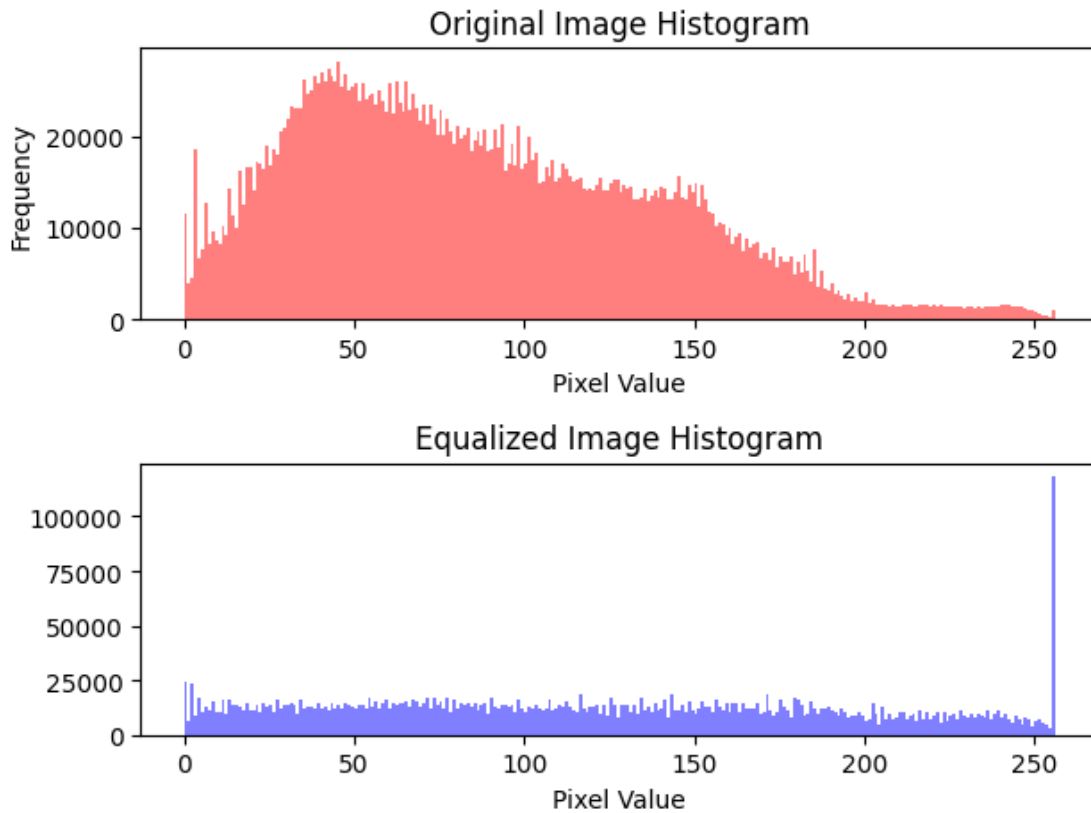

Original Image   Equalized Image

```
[4]: # Display Histograms
     plt.figure("Hunting Lion (Histograms)")

     plt.subplot(2, 1, 1)
     plt.hist(input_image.flatten(), 256, [0, 256], color='r', alpha=0.5)
     plt.title("Original Image Histogram")
     plt.xlabel("Pixel Value")
     plt.ylabel("Frequency")

     plt.subplot(2, 1, 2)
     plt.hist(equalized_image.flatten(), 256, [0, 256], color='b', alpha=0.5)
     plt.title("Equalized Image Histogram")
     plt.xlabel("Pixel Value")

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

The histogram above shows the intensity distribution of the original and equalized image. The equalized image has a more uniform intensity distribution compared to the original image. This results in an image with better contrast and brightness.

Read More:

- Histogram Equalization
- Histogram Equalization in OpenCV

6 Summary

- `Contrast` is the difference in brightness between objects or regions.
- `Contrast Manipulation` is the process of changing the contrast of an image.
- `Histogram Equalization` is a technique used to improve the contrast of an image by redistributing the intensity values.
- `Histogram` is a graphical representation of the distribution of pixel intensities in an image.
- `Contrast Stretching` is a technique used to improve the contrast of an image by stretching the intensity values.
- `Gamma Correction` is a technique used to improve the contrast of an image by applying a power-law transformation.
- `Histogram Matching` is a technique used to improve the contrast of an image by matching the histogram of the input image to the histogram of a reference image.

7 References

- Thomas G. (2022). Graphic Designing: A Step-by-Step Guide (Advanced). Larsen & Keller. ISBN: 978-1-64172-536-1
- Singh M. (2022). Computer Graphics and Multimedia. Random Publications LLP. ISBN: 978-93-93884-95-4
- Singh M. (2022). Computer Graphics Science. Random Publications LLP. ISBN: 978-93-93884-03-9
- Singh M. (2022). Computer Graphics Software. Random Publications LLP. ISBN: 9789393884114
- Tyagi, V. (2021). Understanding Digital Image Processing. CRC Press.
- Ikeuchi, K. (Ed.). (2021). Computer Vision: A Reference Guide (2nd ed.). Springer.
- Bhuyan, M. K. (2020). Computer Vision and Image Processing. CRC Press.
- Howse, J., & Minichino, J. (2020). Learning OpenCV 4 Computer Vision with Python 3: Get to grips with tools, techniques, and algorithms for computer vision and machine learning. Packt Publishing Ltd.
- Kinser, J. M. (2019). Image Operators: Image Processing in Python. CRC Press.