3 - 4.4. Brightness Manipulation

September 27, 2024

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

1 Introduction

`Brightness Manipulation` is the process of changing the brightness of an image. It is a common image processing task that can be used to improve the visibility of an image or to add special effects to an image.

2 Setup

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

3 Initial Setup

[4]:
```
# Import Libraries
import cv2
import matplotlib.pyplot as plt

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

# Image Path
image_path = asset_root + '/images/parrot.jpg'

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

# Display Both Image
plt.figure("Parrot")

plt.imshow(input_image)
plt.title("Original Image")
plt.axis('off')
```

```
plt.show()
```

4 Brightness Manipulation

To increase the brightness of an image, we can add a positive value to each pixel of the input image. The following code snippet increases the brightness of the input image by adding a constant value of `100` to each pixel of the input image.

Brightness Manipulation can be done using the following formula:

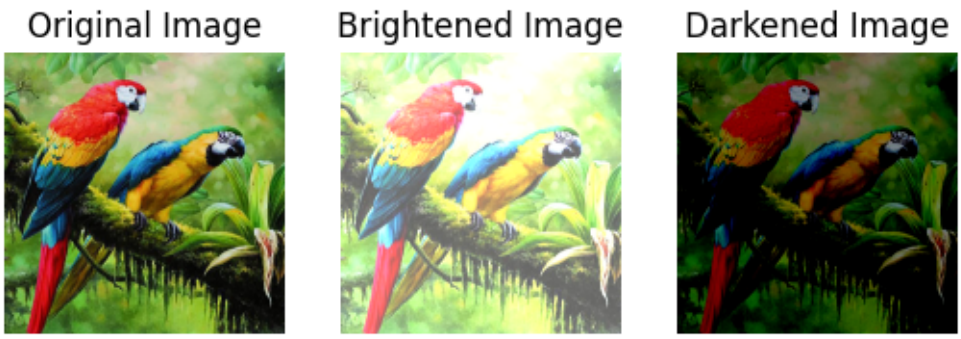
$$new\_pixel\_value = old\_pixel\_value + brightness\_factor$$

where:

- `new_pixel_value` is the pixel value of the output image.
- `old_pixel_value` is the pixel value of the input image.
- `brightness_factor` is a constant value that is added to each pixel of the input image.

[5]:
```
# Increase Brightness
brightness_factor = 100
brightened_image = cv2.add(input_image, brightness_factor)
```

```
# Decrease Brightness
darkened_image = cv2.subtract(input_image, brightness_factor)

# Display Both Image
plt.figure("Parrot")

plt.subplot(1, 3, 1)
plt.imshow(input_image)
plt.title("Original Image")
plt.axis('off')

plt.subplot(1, 3, 2)
plt.imshow(brightened_image)
plt.title("Brightened Image")
plt.axis('off')

plt.subplot(1, 3, 3)
plt.imshow(darkened_image)
plt.title("Darkened Image")
plt.axis('off')

plt.show()
```



In the above code snippet, the `cv2.add()` function is used to add the `brightness_factor` to each pixel of the input image. The `cv2.add()` function automatically clips the pixel values to the range `[0, 255]` after adding the `brightness_factor` to each pixel of the input image.

To decrease the brightness of an image, we can subtract a positive value from each pixel of the input image. The following code snippet decreases the brightness of the input image by subtracting a constant value of `100` from each pixel of the input image.

Read More:

- Brightness Manipulation

5 Summary

- `Brightness Manipulation` is the process of changing the brightness of an image.
- `Brightness` is a measure of the intensity of light in an image.
- To increase the brightness of an image, we can add a positive value to each pixel of the input image.
- To decrease the brightness of an image, we can subtract a positive value from each pixel of the input image.

6 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.