

## 2.2. Image Resolution

August 26, 2024

**Jarrian Vince G. Gojar**

Instructor I

*College of Information and Communications Technology, Sorsogon State University, Philippines*

### 1 Image Resolution

Image Resolution is the number of pixels in an image. It is usually expressed as width x height. For example, an image with resolution 1920x1080 has 1920 pixels in width and 1080 pixels in height.

### 2 Setup

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

### 3 Initial Setup

```
[ ]: # Importing the required libraries
import cv2
import numpy as np
import matplotlib.pyplot as plt

# Reading the image
image_path = '../assets/images/parrot.jpg'

# Reading an image using OpenCV
original_image = cv2.imread(image_path)
grayscale_image = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)

# Converting the image to RGB
original_image = cv2.cvtColor(original_image, cv2.COLOR_BGR2RGB)
gray_image = cv2.cvtColor(grayscale_image, cv2.COLOR_BGR2RGB)

# Displaying the image
print("Displaying the Original Image and the Grayscale Image")
plt.figure("Original Image and Grayscale Image")

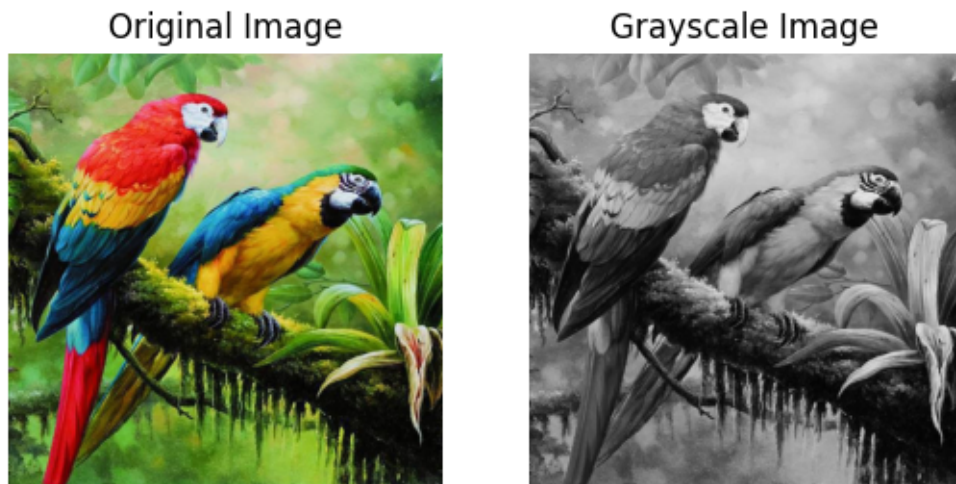
plt.subplot(1, 2, 1)
```

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

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

plt.show()
```

Displaying the Original Image and the Grayscale Image



## 4 Types of Resolution in Images

There are mainly two types of resolution in an image:

- **Intensity Resolution**
  - It is the number of bits used to represent the intensity of each pixel in an image. It is also known as pixel depth or color depth. It is usually measured in bits per pixel (bpp).
  - The intensity resolution of an image is the number of bits used to represent the intensity of each pixel in an image. It is also known as pixel depth or color depth.
- **Spatial Resolution**
  - It is the number of pixels used to represent an image. It is usually measured in pixels per inch (ppi) or dots per inch (dpi).
  - The spatial resolution of an image is the number of pixels used to represent an image. It is usually measured in pixels per inch (ppi) or dots per inch (dpi).

## 4.1 Intensity Resolution

The intensity resolution on an image refers to number of gray levels that can be distinguished in the image. The intensity resolution is determined by the number of bits used to represent the intensity levels.

For example, an 8-bit image has an intensity resolution of  $2^8 = 256$  gray levels. A 12-bit image has an intensity resolution of  $2^{12} = 4096$  gray levels. Quiet commonly a pixel in a grayscale image is represented by 8 bits, which means that the intensity resolution is 256 gray levels. This is a convenient number of gray levels to work with, as it is a power of 2 and can be represented by a single byte. It also falls into the range of human perception. In other words, if an image had 1000 gray levels, the human eye would not be able to distinguish between all of them and common display devices would not be able to display all of them.

```
[ ]: # Displaying the image in different intensity resolutions
# 256 gray levels or 8-bit image
g_img_8_bit = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)

# 64 gray levels or 6-bit image
g_img_6_bit = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)
g_img_6_bit = cv2.normalize(g_img_6_bit, None, 0, 63, cv2.NORM_MINMAX)

# 16 gray levels or 4-bit image
g_img_4_bit = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)
g_img_4_bit = cv2.normalize(g_img_4_bit, None, 0, 15, cv2.NORM_MINMAX)

# 4 gray levels or 2-bit image
g_img_2_bit = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)
g_img_2_bit = cv2.normalize(g_img_2_bit, None, 0, 3, cv2.NORM_MINMAX)

# 2 gray levels or 1-bit image
g_img_1_bit = cv2.imread(image_path, cv2.IMREAD_GRAYSCALE)
g_img_1_bit = cv2.normalize(g_img_1_bit, None, 0, 1, cv2.NORM_MINMAX)

# Displaying the image
print("Displaying the Image in Different Intensity Resolutions")
plt.figure("Image in Different Intensity Resolutions")

plt.subplot(2, 3, 1)
plt.imshow(g_img_8_bit, cmap='gray')
plt.title("8-bit Image")
plt.axis('off')

plt.subplot(2, 3, 2)
plt.imshow(g_img_6_bit, cmap='gray')
plt.title("6-bit Image")
plt.axis('off')
```

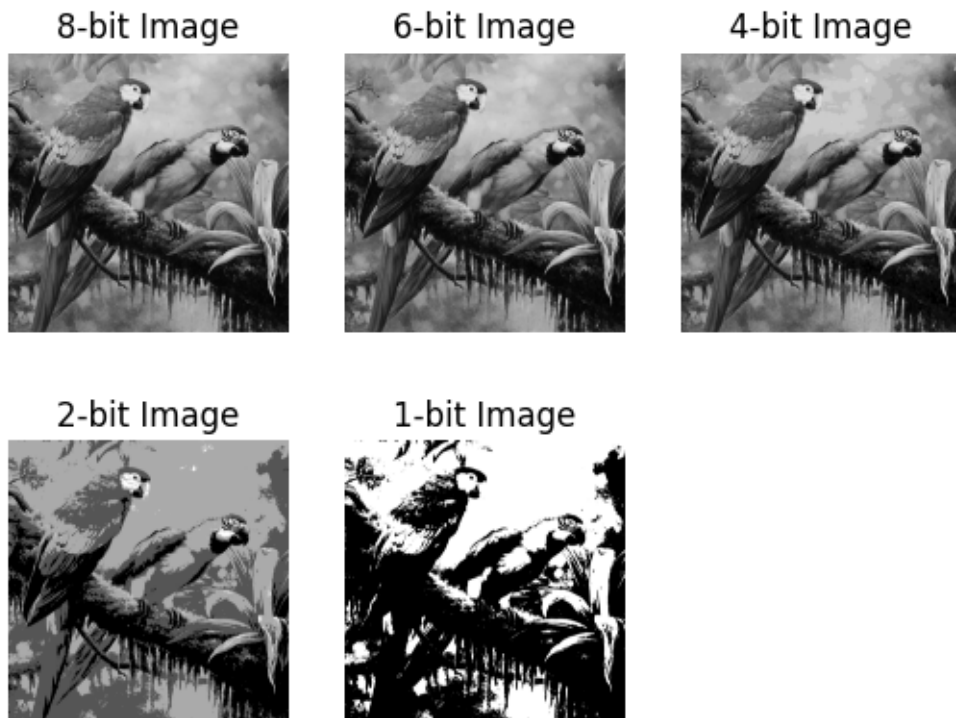
```
plt.subplot(2, 3, 3)
plt.imshow(g_img_4_bit, cmap='gray')
plt.title("4-bit Image")
plt.axis('off')

plt.subplot(2, 3, 4)
plt.imshow(g_img_2_bit, cmap='gray')
plt.title("2-bit Image")
plt.axis('off')

plt.subplot(2, 3, 5)
plt.imshow(g_img_1_bit, cmap='gray')
plt.title("1-bit Image")
plt.axis('off')

plt.show()
```

Displaying the Image in Different Intensity Resolutions



The above code reduces the intensity resolution of the grayscale image to 64, 16, 4, 2, and 1 gray levels. The intensity resolution of the grayscale image is reduced by normalizing the pixel values to the desired range. To reduce the intensity resolution of the grayscale image, the `cv2.normalize()` function is used.

**Important codes to remember:**

- `cv2.normalize()`
  - This function normalizes the input array to a given range.
  - Syntax: `cv2.normalize(src, dst, alpha, beta, norm_type, dtype, mask)`
  - Parameters:
    - \* `src`: input array
    - \* `dst`: output array of the same size as `src`
    - \* `alpha`: lower bound of the normalization range
    - \* `beta`: upper bound of the normalization range
    - \* `norm_type`: normalization type
    - \* `dtype`: type of the output array
    - \* `mask`: optional mask
  - Returns: normalized array
  - Reference: [cv2.normalize](#)

## 4.2 Spatial Resolution

Spatial resolution is often equated to the number of pixels in an image, but this is not exactly correct. It is possible to have an image with  $2 \times 2$  block of pixels with the same intensity value. While the number of pixels may be  $V \times H$ , where  $V$  is the number of pixels in the vertical direction and  $H$  is the number of pixels in the horizontal direction, the spatial resolution would only be  $\frac{V}{2} \times \frac{H}{2}$ .

The process of reducing the spatial resolution of an image is called **downsampling**. This process involves reducing the number of pixels in an image. The process of increasing the spatial resolution of an image is called **upsampling** or **interpolation**. This process involves increasing the number of pixels in an image by inserting new pixels between the existing pixels.

```
[ ]: # Downsampling the image up to 8x8
downsampled_image_512 = cv2.resize( grayscale_image, (512, 512))
downsampled_image_256 = cv2.resize( grayscale_image, (256, 256))
downsampled_image_64 = cv2.resize( grayscale_image, (64, 64))
downsampled_image_16 = cv2.resize( grayscale_image, (16,16))
downsampled_image_8 = cv2.resize( grayscale_image, (8, 8))

# Displaying the image
print("Displaying the Downsampling Image")
plt.figure("Downsampling Image")

plt.subplot(2, 3, 1)
plt.imshow( grayscale_image, cmap='gray')
plt.title(f"Original ({grayscale_image.shape[0]}x{grayscale_image.shape[1]})")
plt.axis('off')

plt.subplot(2, 3, 2)
plt.imshow( downsampled_image_512, cmap='gray')
plt.title(f"Image (512x512)")
plt.axis('off')
```

```
plt.subplot(2, 3, 3)
plt.imshow(downsampled_image_256, cmap='gray')
plt.title(f"Image (256x256)")
plt.axis('off')

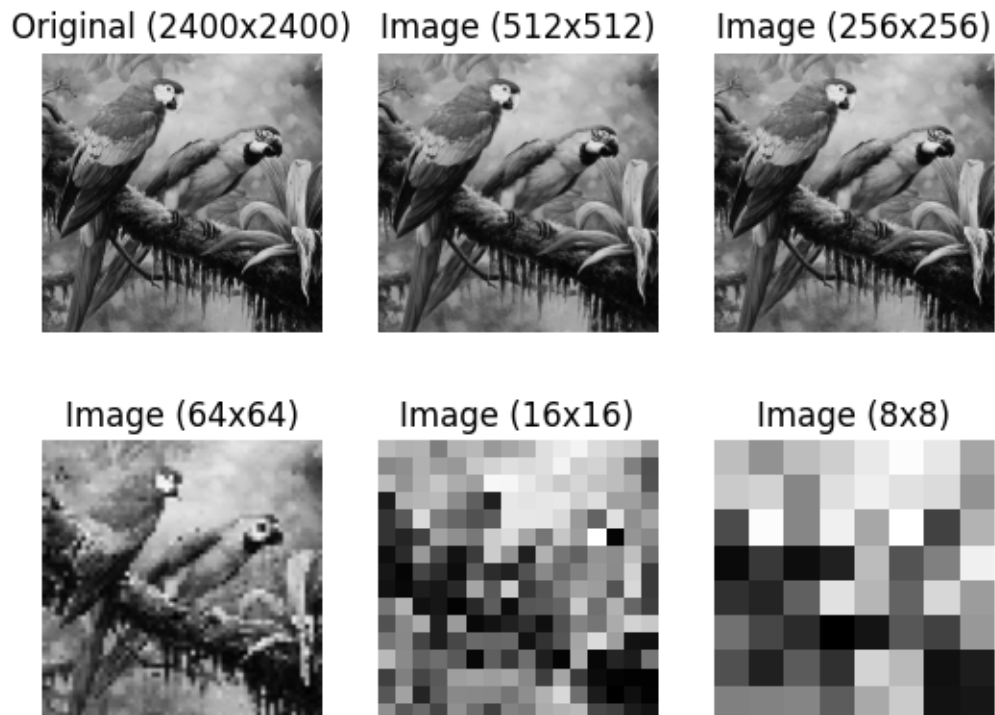
plt.subplot(2, 3, 4)
plt.imshow(downsampled_image_64, cmap='gray')
plt.title(f"Image (64x64)")
plt.axis('off')

plt.subplot(2, 3, 5)
plt.imshow(downsampled_image_16, cmap='gray')
plt.title(f"Image (16x16)")
plt.axis('off')

plt.subplot(2, 3, 6)
plt.imshow(downsampled_image_8, cmap='gray')
plt.title(f"Image (8x8)")
plt.axis('off')

plt.show()
```

Displaying the Downsampling Image



The code above downsamples the image to different resolutions. The original image is  $512 \times 512$  pixels. The image is then downsampled to  $256 \times 256$ ,  $64 \times 64$ ,  $16 \times 16$ , and  $8 \times 8$  pixels.

## 5 Summary

Image Resolution has two types: Intensity Resolution and Spatial Resolution. Intensity Resolution refers to the number of gray levels that can be distinguished in an image. Spatial Resolution refers to the number of pixels used to represent an image. Intensity Resolution is usually measured in bits per pixel (bpp). Spatial Resolution is usually measured in pixels per inch (ppi) or dots per inch (dpi). Downsampling is the process of reducing the number of pixels in an image. Upsampling or Interpolation is the process of increasing the number of pixels in an 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.