05 April 2025 11:29

Image dimensions

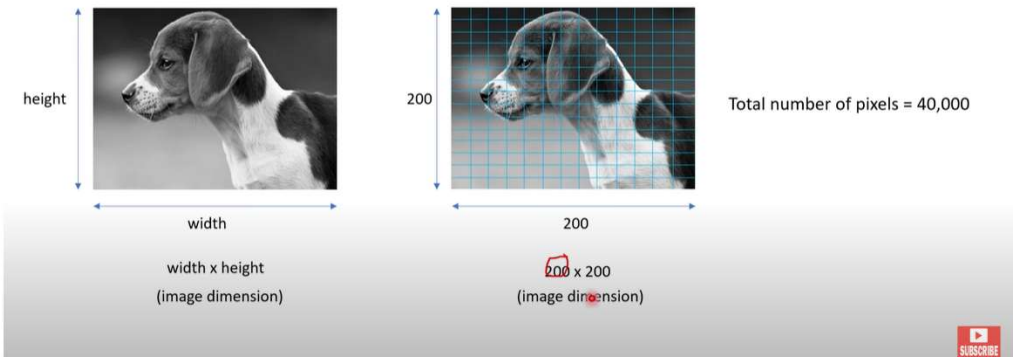
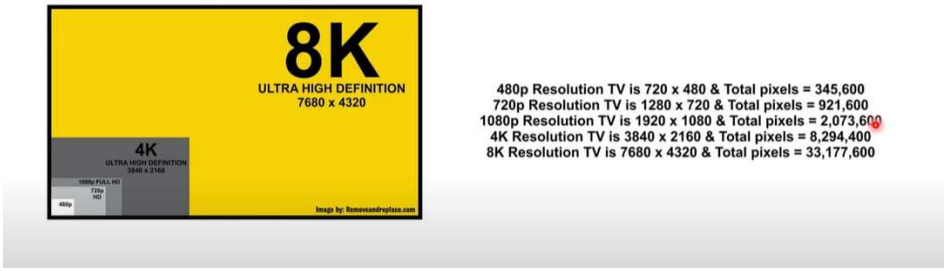


Image resolution



480p Resolution TV is 720 x 480 & Total pixels = 345,600
720p Resolution TV is 1280 x 720 & Total pixels = 921,600
1080p Resolution TV is 1920 x 1080 & Total pixels = 2,073,600
4K Resolution TV is 3840 x 2160 & Total pixels = 8,294,400
8K Resolution TV is 7680 x 4320 & Total pixels = 33,177,600

Types of Images

Grayscale image

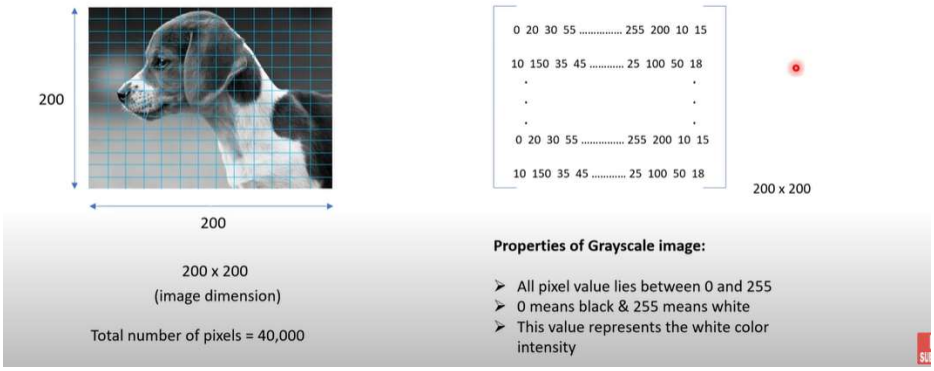
1 channel

RGB image

3 channels:
- Red
- Green
- Blue

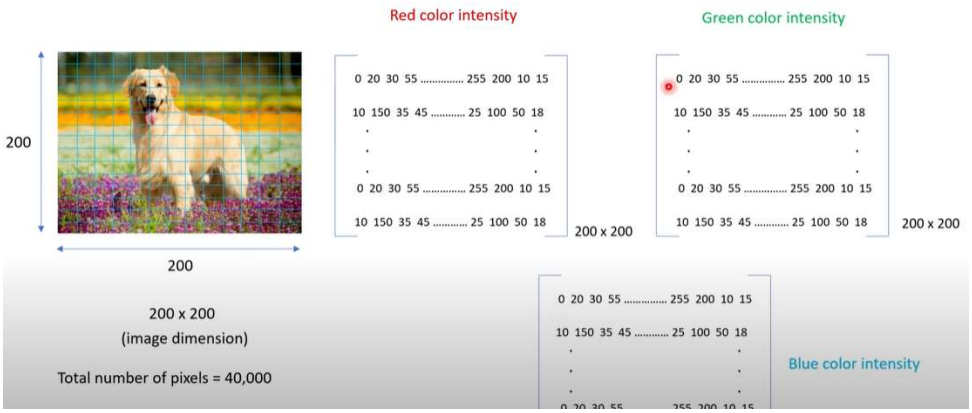
RGB image is heavy so sometimes it is converted into grayscale

Grayscale image



0 means black and 255 means white

RGB image



RGB image have 3 matrices with red, green and blue intensity

RGB image

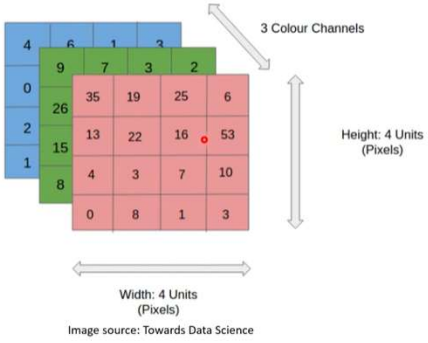


Image processing in python

1. You can use opencv , matplotlib.image or pillow to get the image
2. You need to resize all the images to one single dimension to ensure consistency in dataset (Using pillow)
3. RGB to grayscale

3. opencv

```
[4] import matplotlib.image as mping
    import matplotlib.pyplot as plt
```

3. opencv

```
[4] import matplotlib.image as mping
    import matplotlib.pyplot as plt
```

```
[5] # loading the image using matplotlib

    img = mping.imread('/content/puppy image.jpeg')
```

```
[6] type(img)
```

```
numpy.ndarray
```

```
[8] img.shape
```

```
(2000, 3000, 3)
```

```
[10] # displaying the image from numpy array

     img_plot = plt.imshow(img)
     plt.show()
```

Resizing the images using pillow

```
[11] from PIL import Image
```

```
[13] img = Image.open("/content/puppy image.jpeg")
```

```
[14] img_resize = img.resize((200,200))
```

```
[17] img_resize.save("dog image resized.jpg")
```

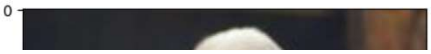
```
[18] img = mping.imread("/content/dog image resized.jpg")
```

```
[20] img.shape
```

```
(200, 200, 3)
```

```
[21] img_plot = plt.imshow(img)
     img_plot
```

```
<matplotlib.image.AxesImage at 0x7c380cc23390>
```



RGB to grayscale image

```
[22] import cv2
```

```
[32] img = cv2.imread("/content/puppy image.jpeg")
```

```
[26] type(img)
```

```
numpy.ndarray
```

```
[27] img.shape
```

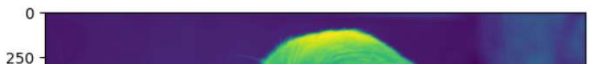
```
(2000, 3000, 3)
```

```
[33] grayscale_img = cv2.cvtColor(img, cv2.COLOR_RGB2GRAY)
```

```
[29] grayscale_img.shape
```

```
(2000, 3000)
```

```
img = plt.imshow(grayscale_img)
plt.show()
```



Transfer Learning

Transfer Learning is a Deep Learning technique where we use a pre-trained model. This pre-trained model is trained for one task and can be re-trained for a similar task with a smaller dataset.

Transfer Learning gives higher accuracy compared to training models from scratch.

Examples of Pre-Trained Models:

- VGG-16
- ResNet50
- Inceptionv3
- MobileNet V2