Color Pixel Theory and Image Representation

Pre-work: Computer Vision

Agenda

- Basics of Images
- Representation of Images
- PIL (Python Imaging Library)

Basics of Images

Basics of Images

- An image is made up of small square-like boxes or elements called **pixels**.

- Every image is simply a combination of multiple pixels, each of which has its own color.

- As seen below, an image is simply a combination of multiple pixels with individual colors.

- Every pixel in an image has an intensity value which ranges from 0 to 255.
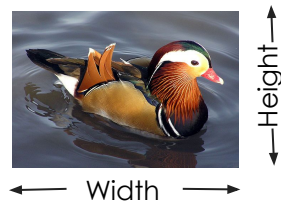 This is known as the **Pixel Intensity Value**.
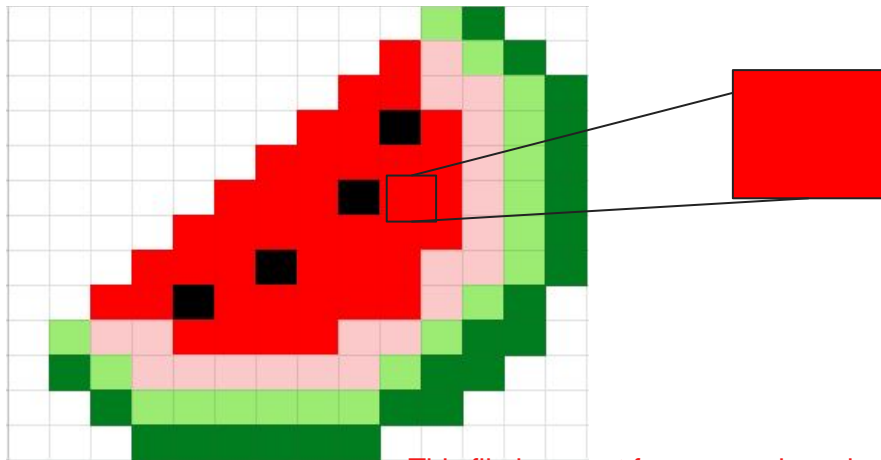
Basics of Images

- Images have three major components:

| |
|---|
| **Size** |
| This represents the height and width of an image. It is usually measured by number of pixels. |
| **Color Space** |
| This represents the different possible color spaces, like Grayscale, RGB, HSV. The image of the duck on the right is represented in RGB color space. |
| **Channels** |
| This explains the attributes of a color space - For example, RGB has three color channels: Red, Green and Blue. |

Height

Width

Grayscale

RGB
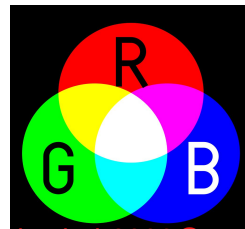
The RGB Color Space

Basics of Images

- The RGB image can be broken down into three different channels as shown below:
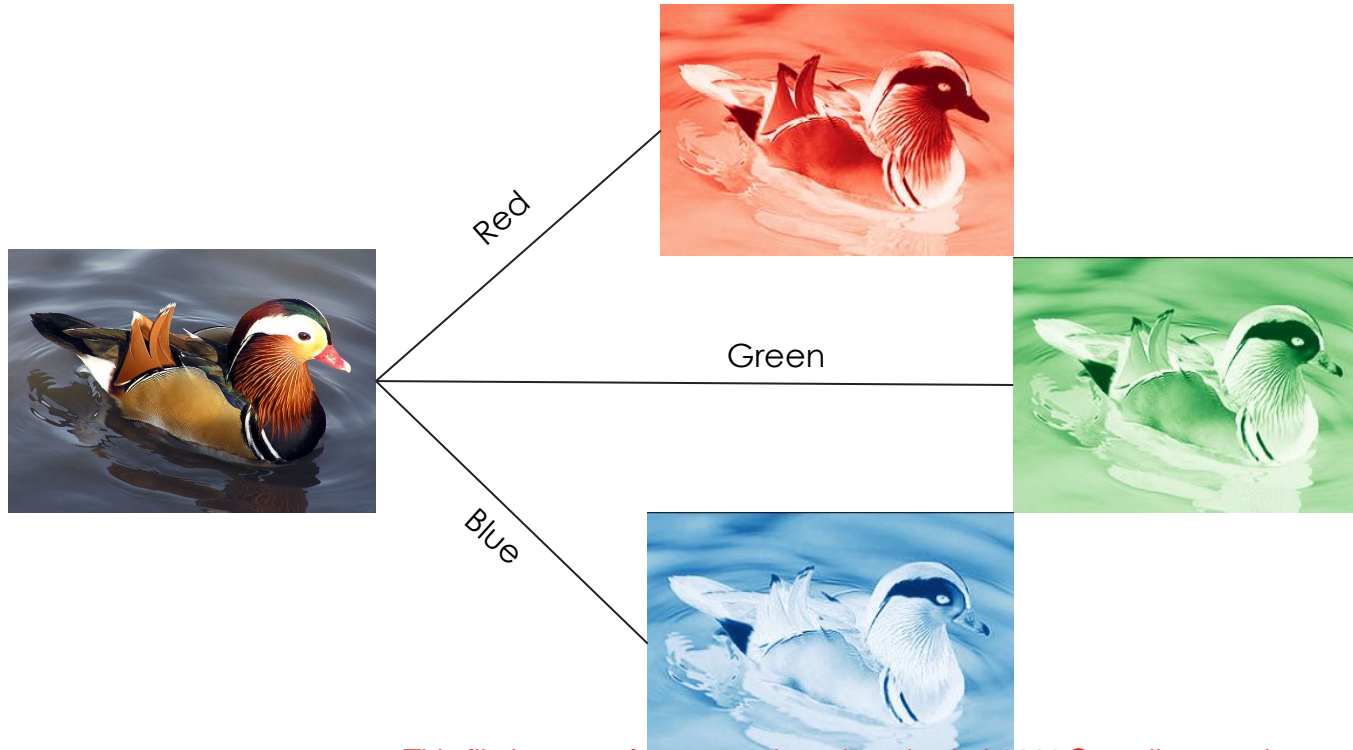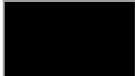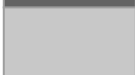

Red

Green

Blue

The RGB Color Scheme

- The colors of an image are determined by its pixel values. An RGB image has 3 color channels - Red, Green and Blue. Here each channel has a pixel value ranging from 0 to 255. For example, the number 0 in a channel means there is no color, and 255 means there is 100% color. If a pixel value is represented by [255,255,0], it means that we have 100% Red and Green colors, and there is no Blue color.

- **The higher the pixel intensity value, the more the brightness of the color.**

Color	Color Name	Pixel Intensity Values - RGB Color Space		
		R (Red)	**G** (Green)	**B** (Blue)
	Red	255	0	0
	Green	0	255	0
	Blue	0	0	255
	Yellow	255	255	0
	Cyan	0	255	255

Grayscale Colors

- Grayscale colors are special in the RGB color scheme, because **every grayscale color** (from white to black to all shades of gray) **always has equal values for R, G and B.**

- Due to this, grayscale colors can be represented by a single number as opposed to the three numbers that three color channels require in RGB. **A grayscale image hence has only one channel,** where the pixel values range from 0 to 255. The pixel value 0 represents black and the value 255 represents white. The method of conversion of images from RGB, HSV, etc to a grey shaded image is called Grayscaling.

Color	Color Name	Pixel Intensity Values - RGB Color Space		
		R (Red)	**G** (Green)	**B** (Blue)
	Black	0	0	0
	White	255	255	255
	Dim Gray	100	100	100
	Very Light Gray	200	200	200

Grayscale Images

- Why is Grayscaling important in computer vision?

  Using a grayscale image over an RGB image helps in **dimensionality reduction** as an RGB image has 3 channels, whereas a grayscale image only has one. This helps with computational cost for the algorithm.

- In order to convert any color into its grayscale equivalent, one conversion formula often used is to simply add up the R, G and B values, and divide by 3 (the arithmetic average), as that would redistribute the total intensity of the three channels into each channel equally, hence creating a grayscale color.
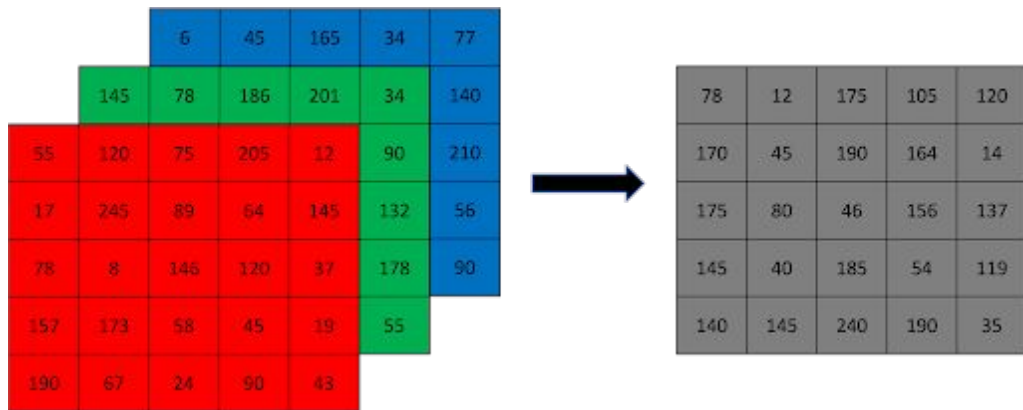

RGB

Grayscale

The HSV Color Scheme

- HSV stands for Hue Saturation Value. It has three main components which can be described as:
  - **Hue:** It is the color segment or color portion of the image. It is expressed in degrees so the values range from 0 to 360 degrees.
  - **Saturation:** It describes the amount of gray shade in a particular color. It is expressed in percentage so it ranges from 0 to 100 percent. 0 represents the highest gray shade, 100 appears as pure color.
  - **Value:** It represents the intensity or brightness of a color, it is also expressed in percentage so it ranges from 0 to 100 percent.
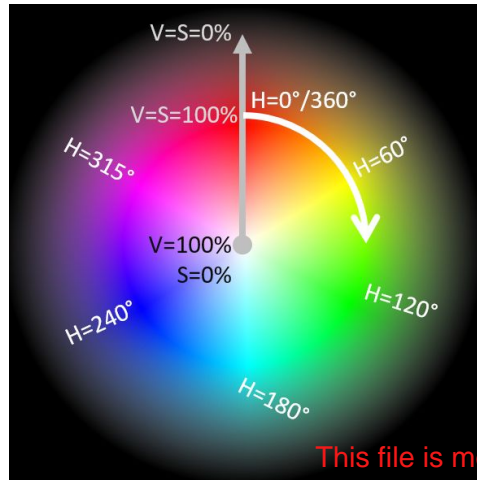
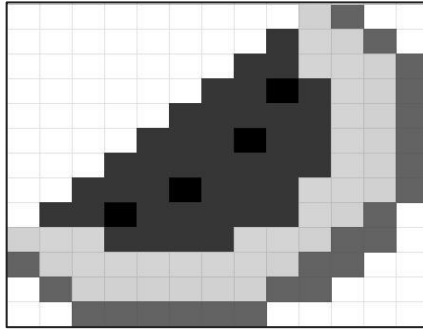# Representation of Images

Representation of Images as Arrays

- **A grayscale image can be represented using a 2D array.** This is because each grayscale pixel would have just one channel and hence one number for its pixel intensity value, so an image, which is just a 2D array of pixels, would mathematically just be a 2D array of pixel intensity values.

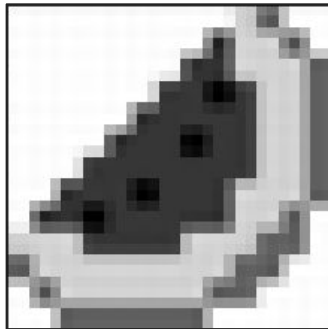A Grayscale image only consists of a 2D array of grayscale pixels, such as the pixel highlighted below
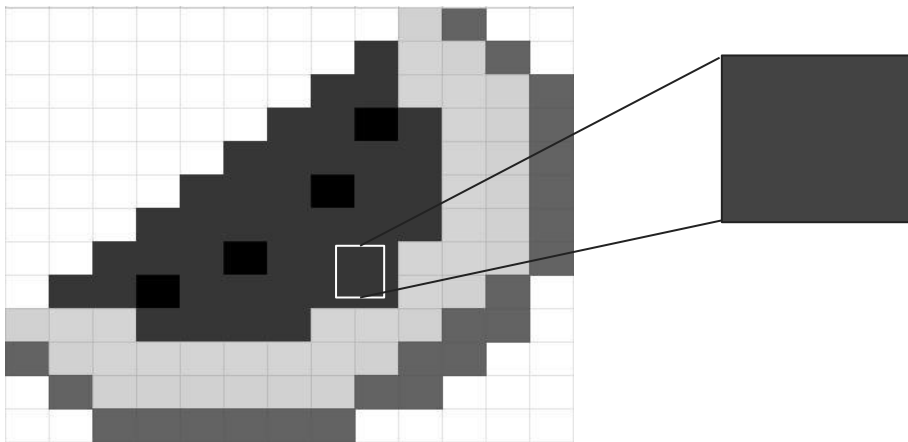
Representation of Images as Arrays

440 x 340

30 x 30

The 2D Array Representation of the Image

Matrix of size 30 x 30

Representation of Images as Arrays

- Representing **multiple grayscale images** on the other hand, would require using multiple 2D arrays, or in other words, **a 3D array.** The extra dimension present (generally at the beginning) would show the number of sample images, while the other two dimensions would be the height and width of each image. This can be represented as:

Shape: (no. of samples, height, width)
Ex: (3,16,16)

Columns (Width)

No. of Samples

Rows (Height)

78	12	175	105	120		
170	78	12	175	105	120	
175	170	78	12	175	105	120
145	175	170	45	190	164	14
140	145	175	80	46	156	137
	140	145	40	185	54	119
		140	145	240	190	35

Representation of Images as Arrays

- **A single RGB image is also a 3D array with the depth dimension always having a value of 3,** since each pixel of the 2D image has three channels (R, G and B).

- However, **representing multiple RGB images would require a 4D array**, because an extra dimension is required to show the number of sample images. For example:

Shape: (samples, height, width, color channels) e.g. (5,16,16,3)
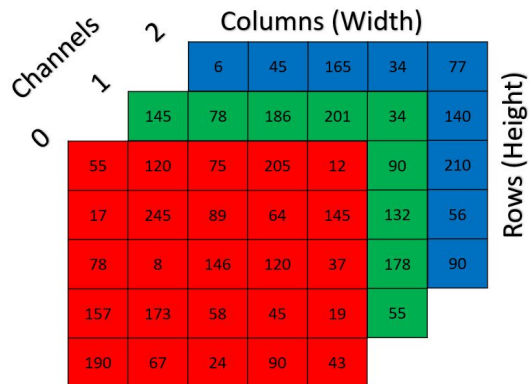
A single RGB Image can also be represented as a 3D array with a depth of 3, on account of having 3 channels (R, G and B)

Pixel Normalization

- So to recap, in **Grayscale Images**, **each pixel can be represented by a single number.** However in **RGB colored images**, **each pixel has to be represented by a vector of three numbers,** for the three primary color channels: red, green, and blue.

- As we also saw earlier, the pixel intensity values of the RGB digital color space **vary from 0 to 255**.

- These pixel intensity values representing the image, can also be **normalized / rescaled** into a range from **[0,1]**, as this helps reduce the storage used for each image's pixel values.

- This kind of normalization / scaling is preferred for neural networks in computer vision, since computational cost is always an important consideration in Deep Learning.
  It is implemented using a **rescaling ratio** by which each pixel can be multiplied in order to achieve the desired range. An example of such a ratio is **1/255** (about 0.0039).

Common Image Characteristics

- There are certain standard image resolution and aspect ratios that are often used with images in real-world applications.

- **The Aspect Ratio** of an image is a term used to describe the ratio of the width of an image to its height. It is usually denoted with two numbers separated by a colon.

  A few common image aspect ratios are 1:1, 3:2, 5:4 and 16:9.

- **The Resolution** of an image, on the other hand, is a term that describes how many pixels the image consists of.

  For example: An image with a width of 640 pixels and a height of 480 pixels is said to have a resolution of 640x480, which is over 0.3 MP (Megapixels).

  The higher the number of pixels in an image, the higher its resolution.

PIL (Python Imaging Library)

PIL (Python Imaging Library)

- Manipulating the pixel intensity values of an image (also called **Filtering**), is an important part of the image pre-processing stage of Computer Vision.

- Performing image manipulation tasks manually through code can be a tedious task, so libraries such as **PIL** (Python Imaging Library) and **OpenCV,** which have in-built pixel alteration functions, are often used to achieve such tasks.

- The PIL library consists of methods that can extract the pixelmap from an image and change pixel intensities by iterating over each pixel value.

**Python
Imaging
Library**

RGB to Grayscale Conversion

- Simple averaging formula to convert an RGB image into a Grayscale image:
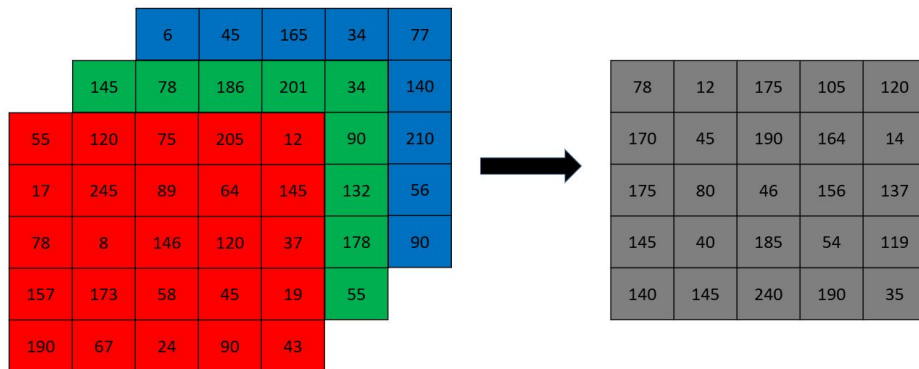 **Grayscale=** **(R+G+B)/3**

- While the above method achieves equal intensity redistribution, a more research-oriented formula, taking into account the increased sensitivity of the human eye to green over the other colors, has been developed that uses a weighted average of the pixel intensity values instead:
  Grayscale = (0.299*R + 0.587*G + 0.114*B)

Simple averaging formula: RGB to Grayscale Conversion

Thank You