

AP CSP

SouthLake Christian Academy
2022-2023

Image Data

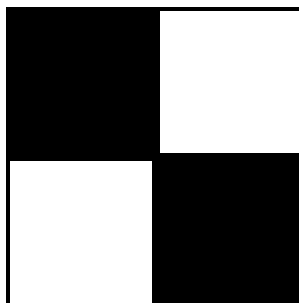
Black and White Images

Apart from numbers, we use many different types of data. If we are using Photoshop, our data is in the form of *images*.

Remember, in binary, `1` means an “on” state, and `0` means an “off” state.

For black and white images, `1` represent white, and `0` represents black. Images are constructed by a bunch of tiny squares or dots called **pixels** (picture elements), and each pixel is either white or black – `1` or `0`.

So if an image has a height and width of 2 pixels, `0b0110` can be interpreted as



Notice that we constructed the image from left to right, top to bottom.

Color Images

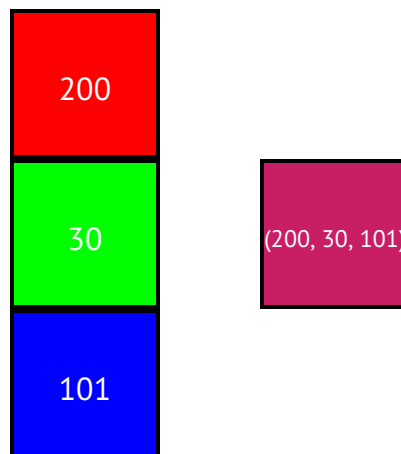
There are several ways to talk about color. One way is to discuss a color’s hue, saturation, and brightness values (HSV format). Another way is to look at a color’s red, green, and blue values – RGB format. If a color is slightly transparent, then we can look at its alpha value as well – RGBA.

Every color is made up of some parts red, some parts green, and some parts blue. A pure red color has its red component fully “turned on”, and its green and blue components are both fully “turned off”.

To represent color, a computer will set aside some bits for the red, green and blue parts. Typically, each component gets a byte. Therefore, it will take 3 bytes (**24 bits**) to represent just one color.

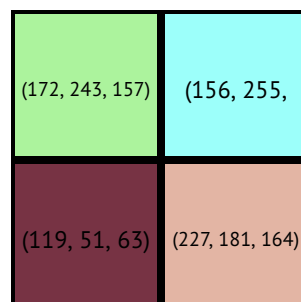
In 24-bit RGB, each component is described as some number between `0` and `255`, where `0` means we have the least of that component and `255` means we have the most of that component.

For example, blend together 200 parts of red, 30 parts of green, and 101 parts of blue, and you get a nice magenta-y color.



What the computer sees is the binary equivalents for 200 (`0b11001000`), 30 (`0b00011110`), and 101 (`0b01100101`) smushed together: `0b110010000001111001100101` represents 1 pixel of that magenta color.

The 2x2 pixel 24-bit color image below would require $2 * 2 * 24 = 96$ bits total.

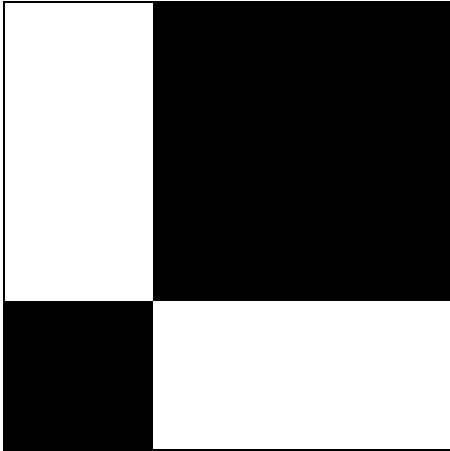


Key Takeaway

Computers do not interpret data; they just consume binary numbers. Humans interpret binary as either decimal numbers, black and white images, or color images. The value of the data does

not have to change, but its representation can change wildly, depending on context.

For example, `0b100100011` can be interpreted as *either*



or `582`, depending if we are in Photoshop or the Calculator App.

Extra Resources

- [BW Images Explore](#)
- [Color Images Explore](#)
- [Slides \(https://docs.google.com/viewer?url=https://github.com/APCSP-SLCA/slides/raw/main/images/slides.pdf\)](https://docs.google.com/viewer?url=https://github.com/APCSP-SLCA/slides/raw/main/images/slides.pdf)

