Encoding color images

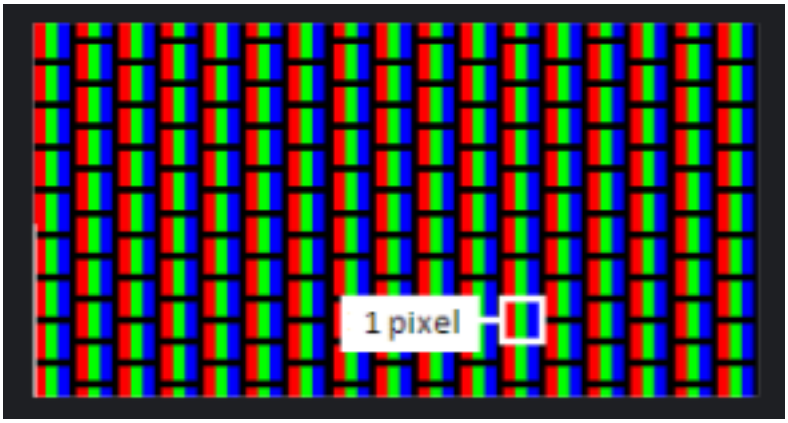
In the video below, the Instagram co-founder explains how images are represented in binary, how digital images are created.

https://www.youtube.com/watch?v=15aqFQQVBWU

We have previously learned that ***Additive color synthesis*** is the creation of colours by combining the three primary colours in light : ***Red, Green and Blue (RGB system)*** and that digital images are based on the RGB colour system.

I-Pixels

Digital images are organised as a grid of coloured squares called ***pixels*** (short for 'picture elements'). Image files like JPEG, GIF and PNG are images made up of pixels and are named ***bitmap images*** or ***bitmaps***.



A single pixel is made up of 3 dots : one Red, one Green and one Blue which have different values that give the ***colour depth***.

The more pixels on the screen, the higher the ***resolution*** and the better the quality of the picture will be. Resolution is a measure of pixel density, usually measured in dots per inch (dpi). The higher the image resolution, the more memory is needed to store the graphic.

II-Bits and binary encoding

Each colour of an image is stored as a ***binary number***

A ***binary system (or binary code)*** is a number system that contains two ***bits, 0 and 1*** ; also known as base 2.

A ***bit*** (short for 'binary digit') is the smallest unit of data in a computer. A bit has therefore a single binary value, either 0 or 1.

The colour depth is therefore given by the number of bits used to indicate the colour of a single pixel.
The number of bits indicates how many colours are available for each pixel.

Each colour RGB is commonly represented by ***8 bits (= 1 byte)***. Other possibilities exist, the more bits a pixel occupies, the more different colours can be represented by this pixel.

So there are $2^8 = 256$ possibilities, colours for each dot of a single pixel: the values range from 0 to 255.
0 will be very dark and 255 will be very bright.

Bit Values

In a binary number, a bit's value depends on its position, starting from the right. Like tens, hundreds, and thousands in a decimal number, a bit's value grows by a power of two as it goes from right to left, as shown in the following chart:

Binary number	1	1	1	1	1	1	1	1
value	$2^7 = 128$	$2^6 = 64$	$2^5 = 32$	$2^4 = 16$	$2^3 = 8$	$2^2 = 4$	$2^1 = 2$	$2^0 = 1$

Conversion

To convert a binary string into a decimal number, multiply the value of each bit by its weight, then add together the products. Therefore, the binary string 0101, in decimal, becomes:

```
2^3x0 + 2^2x1 + 2^1x0 + 2^0x1
= 8x0 + 4x1 + 2x0 + 1x1
= 5
```

The Byte

The **byte** (shortened to the uppercase **B**) is a unit of information composed of 8 bits. It can be used to store, among other things, a character, such as a letter or number.

Grouping numbers in clusters of 8 makes them easier to read, much as grouping numbers in threes helps to make thousands clearer when working in base-10. For example, the number "1,256,245" is easier to read than "1256245".

A 16-bit unit of information is usually called a **word**. A 32-bit unit of information is called a **double word** (sometimes called a **dword**). For a byte, the smallest number possible is 0 (represented by eight zeroes, 00000000), and the largest is 255 (represented by eight ones, 11111111), making for 256 different possible values.

2^7 =128	2^6 =64	2^5 =32	2^4 =16	2^3 =8	2^2 =4	2^1 =2	2^0 =1
0	0	0	0	0	0	0	0
1	1	1	1	1	1	1	1

Examples : In RGB system, with 8 bits per dot, so 24 bits for a single pixel
Red : (255, 0, 0)
Green : (0, 255, 0)
Blue : (0, 0, 255)
Turquoise : (64, 224, 208) equal to (01000000, 11100000, 11010000)

Explanations : **01000000** = $2^7 \times \mathbf{0} + 2^6 \times \mathbf{1} + 2^5 \times \mathbf{0} + 2^4 \times \mathbf{0} + 2^3 \times \mathbf{0} + 2^2 \times \mathbf{0} + 2^1 \times \mathbf{0} + 2^0 \times \mathbf{0}$
= 0+64+0+0+0+0+0+0
= 64

Note : Rather than binary, digital artists often use the ***hexadecimal number*** system to represent colours.
For example the same colour Turquoise can be represented by 6 hexadecimal digits : (40, E0, D0)