POLITECNICO
MILANO 1863

DIPARTIMENTO DI ELETTRONICA
INFORMAZIONE E BIOINGEGNERIA

D E
I B

2023

Dipartimento di Elettronica, Informazione e Bioingegneria

Computer Graphics

Milano, 2023

Computer Graphics

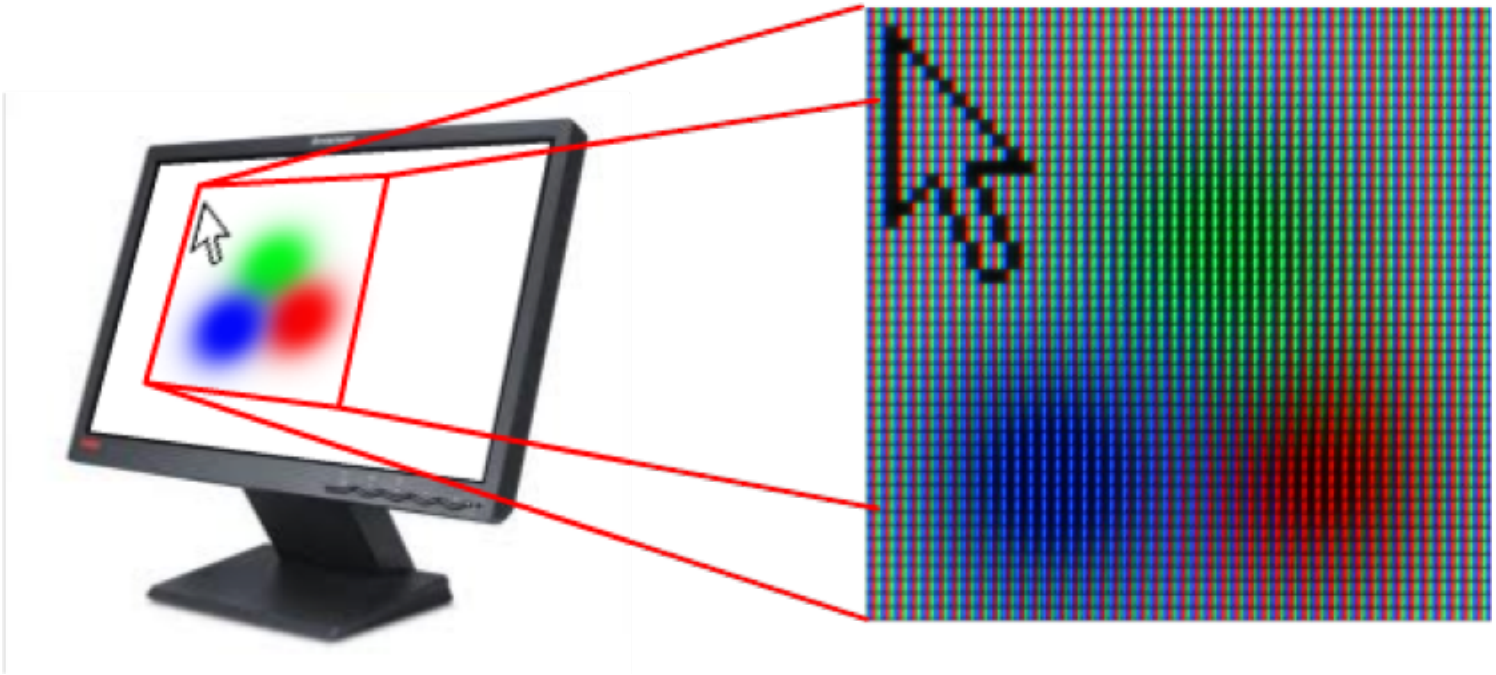
- Graphics adapters, colors and 2D drawings

Graphic adapters

Graphics adapters divide the screen into a matrix of individually assignable elements called *pixels* (picture elements).

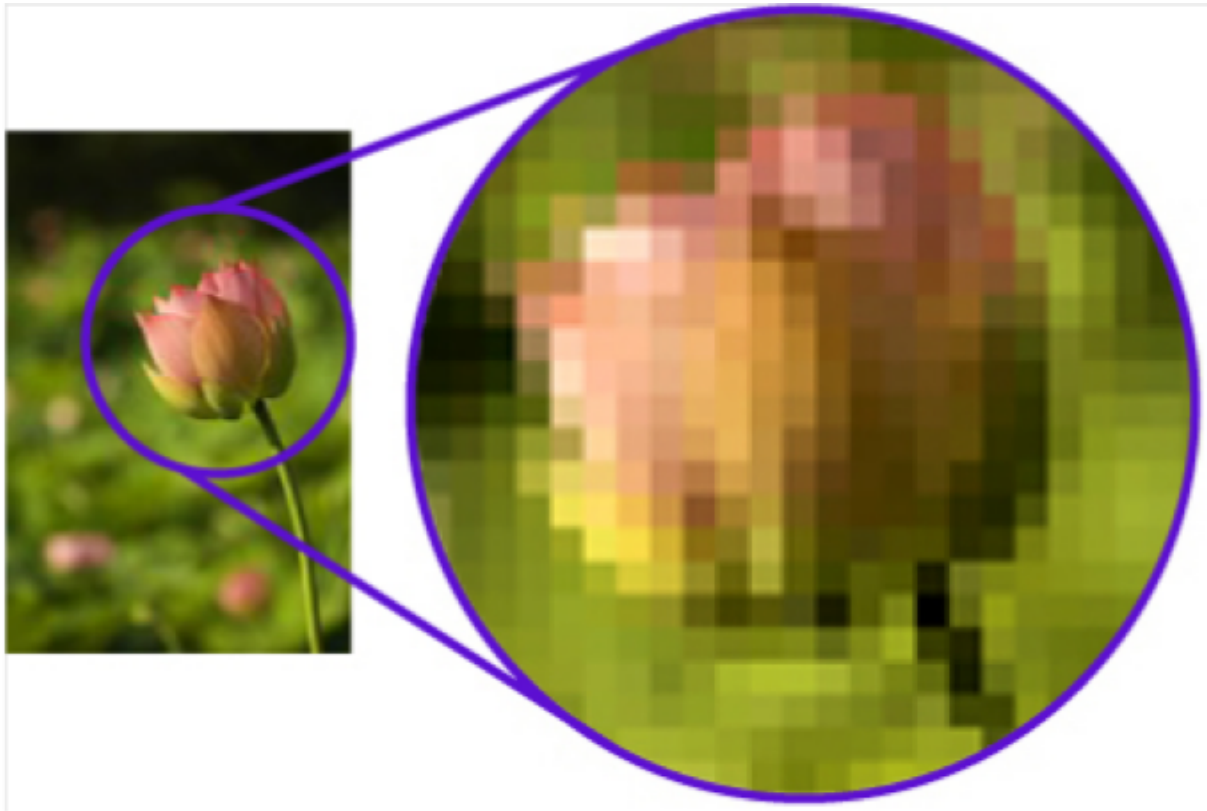
Raster graphics

They decide which color the monitor displays in any of its pixels.



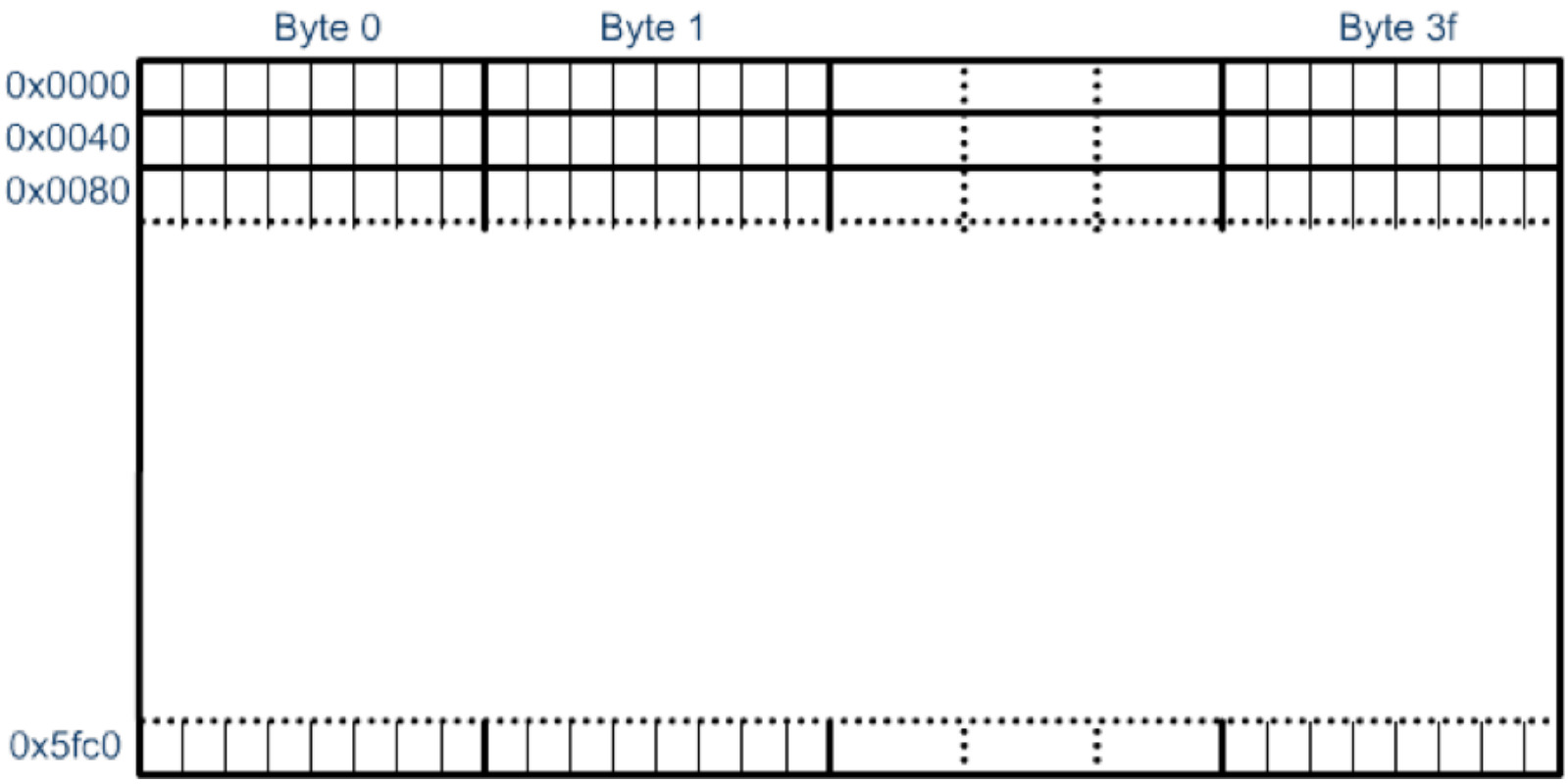
Raster graphics

If the color of the pixels comes from a spatial sampling of an image, the graphic adapter can reproduce it on the monitor.



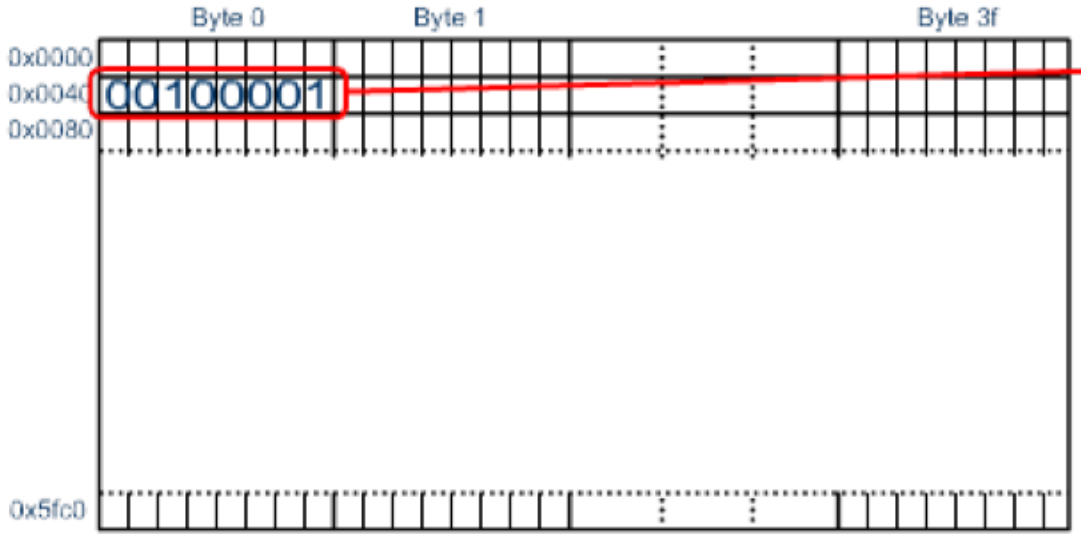
Raster graphics

The graphic adapter has a special memory, called *Video Memory* (VRAM).



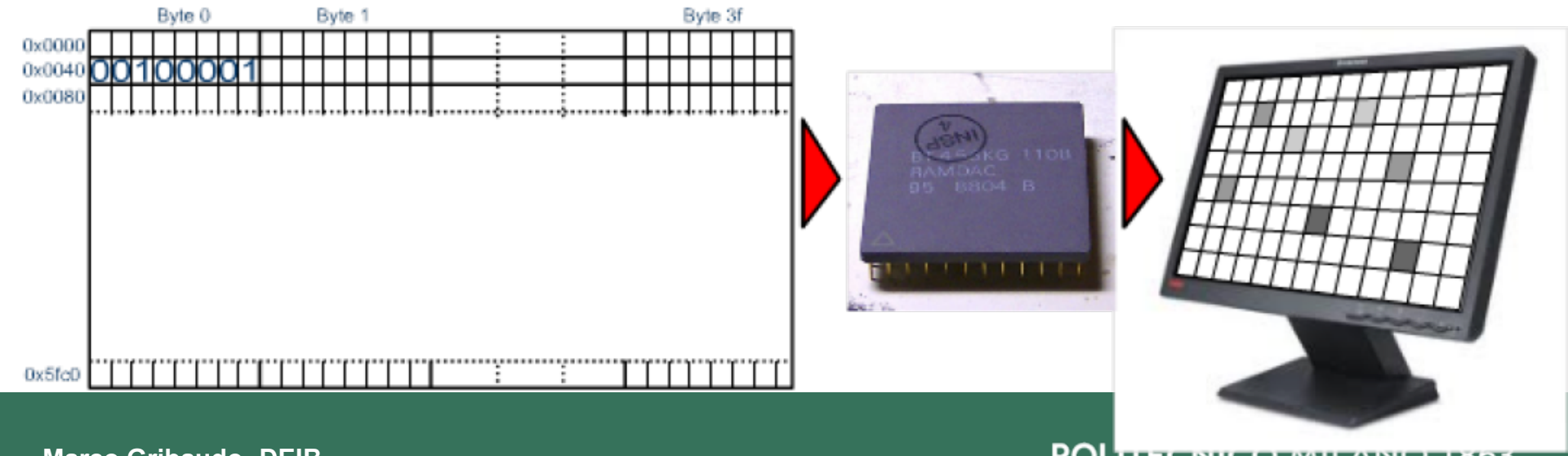
Raster graphics

A part of the VRAM, called the *screen buffer* or *frame buffer*, stores an encoding of the color associated to each pixel shown on screen.



Raster graphics

A special component on the graphics card (called RAMDAC in case of analog displays) uses the information stored in VRAM to transfer the image to the display.



Modern graphic adapters

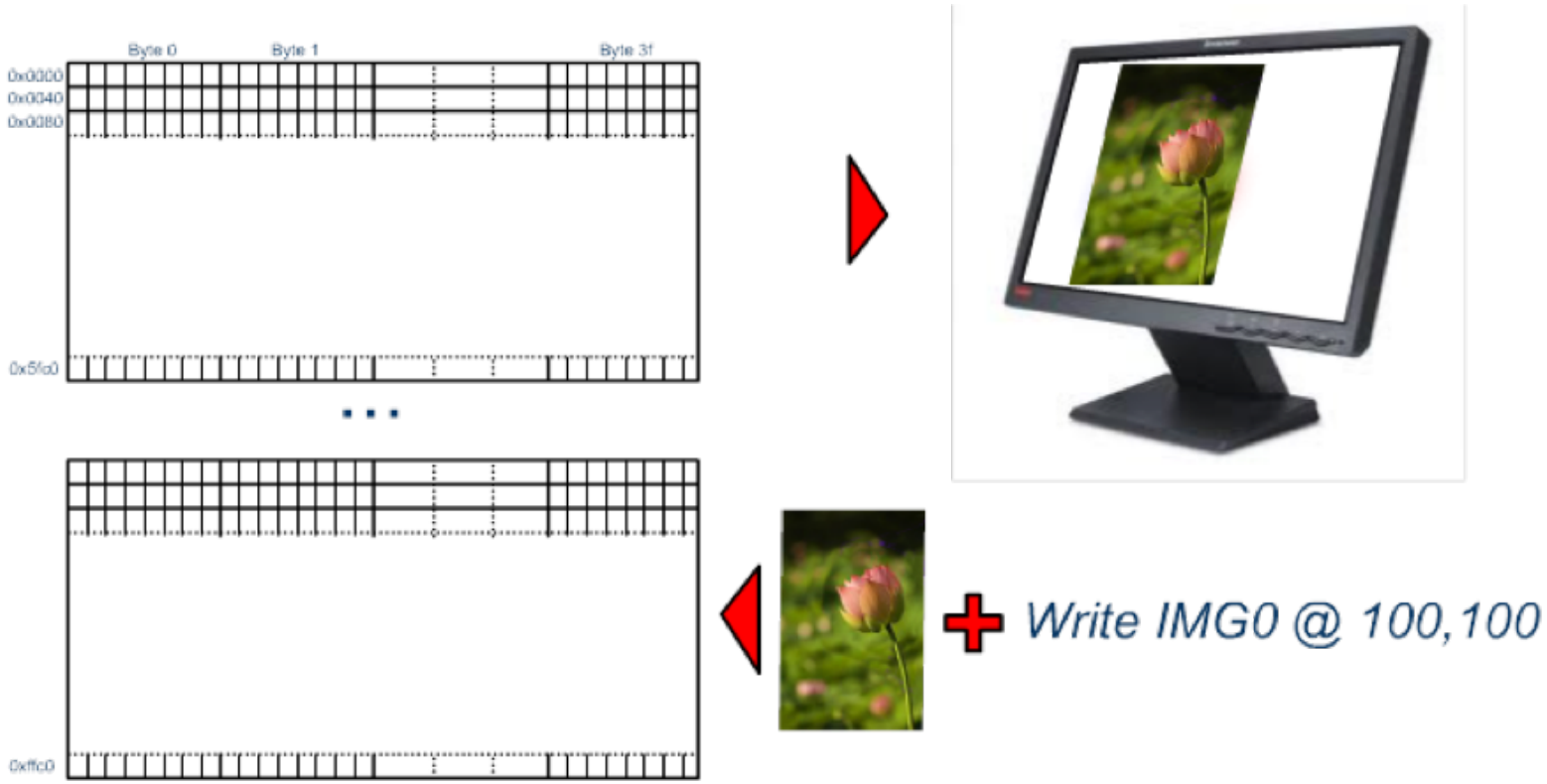
The user basically never writes directly on the screen buffer, but it stores the images in different areas of the VRAM.

Then, the user composes the image shown on screen by sending a list of commands to the graphic adapter: each command can perform tasks like:

- Draw points, lines and other figures
- Write text
- Transfer raster images from the VRAM to the screen buffer
- Perform 3D projections
- Deform and add effects to the images

Modern graphic adapters

With the commands and the data in the VRAM, the adapter composes the image, and sends it to the display.



Complex issues

Interacting with the attached screens can be very complex due to the presence of more than one display, ...

Complex issues

… more than one graphic adapter, …

Complex issues

... or to the ability to show more than one image on the same screen (stereoscopy).

Complex issues

Graphics adapters are usually co-designed with their software drivers: the programmer never actually accesses the registers of the video card, but always interfaces using libraries provided by the hardware manufacturer.

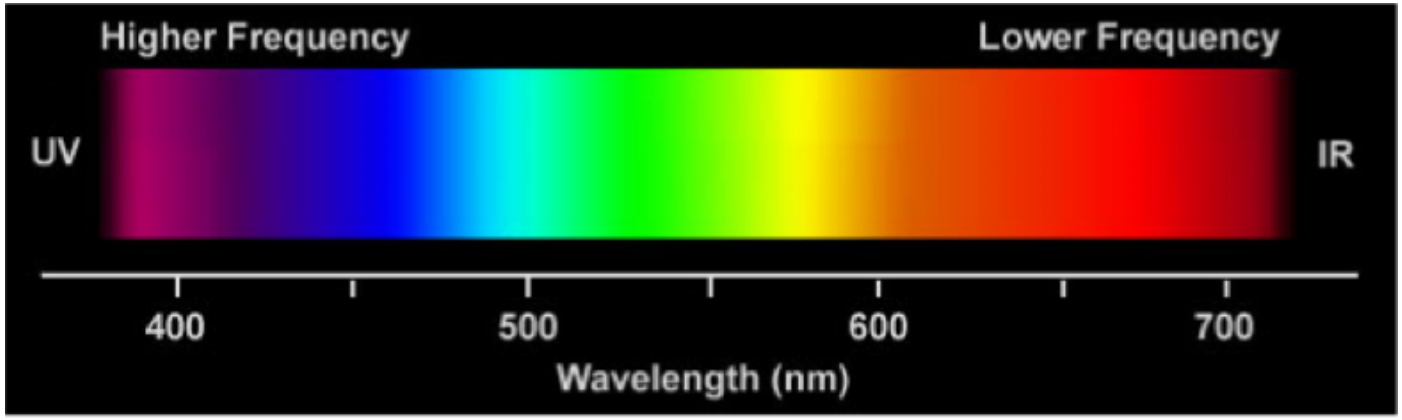
Vulkan is an example of standard set of procedures that unify the way in which functionalities offered by the graphic cards are accessed by the application code.

Color vision

Color-coding is one of the most important topics in computer graphics: it defines how a color on-screen is transformed into a sequence of bits.

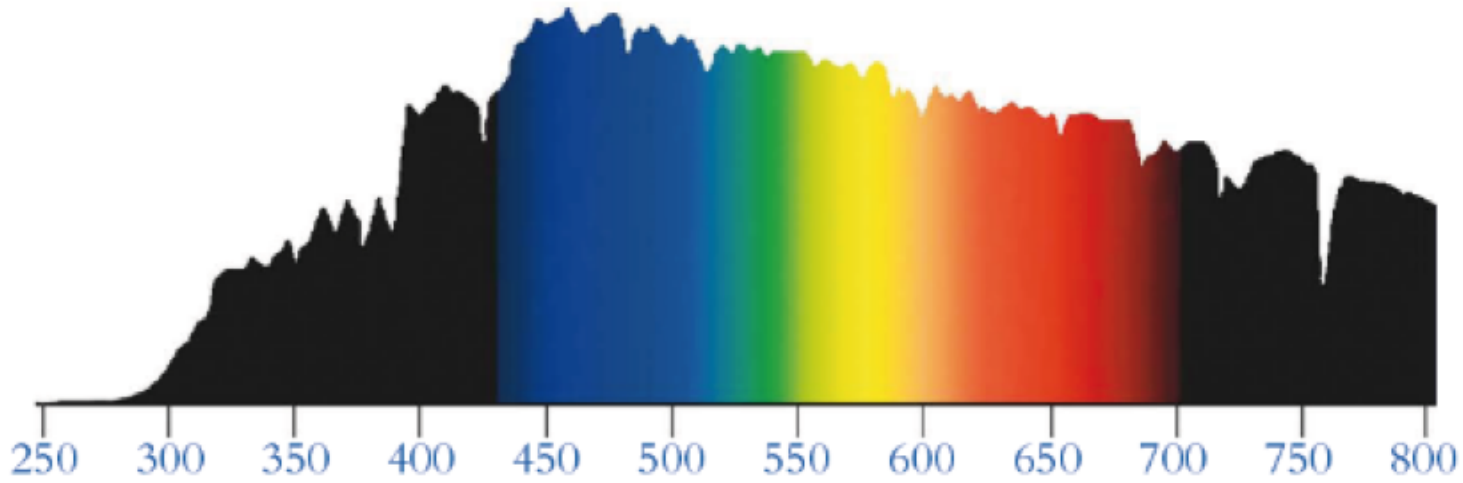
Usually on-screen colors are coded using a system called RGB (Red-Green-Blue).

From the physics, we know that the color of the light is determined by the wavelength of the photons that transmit it.



Color vision

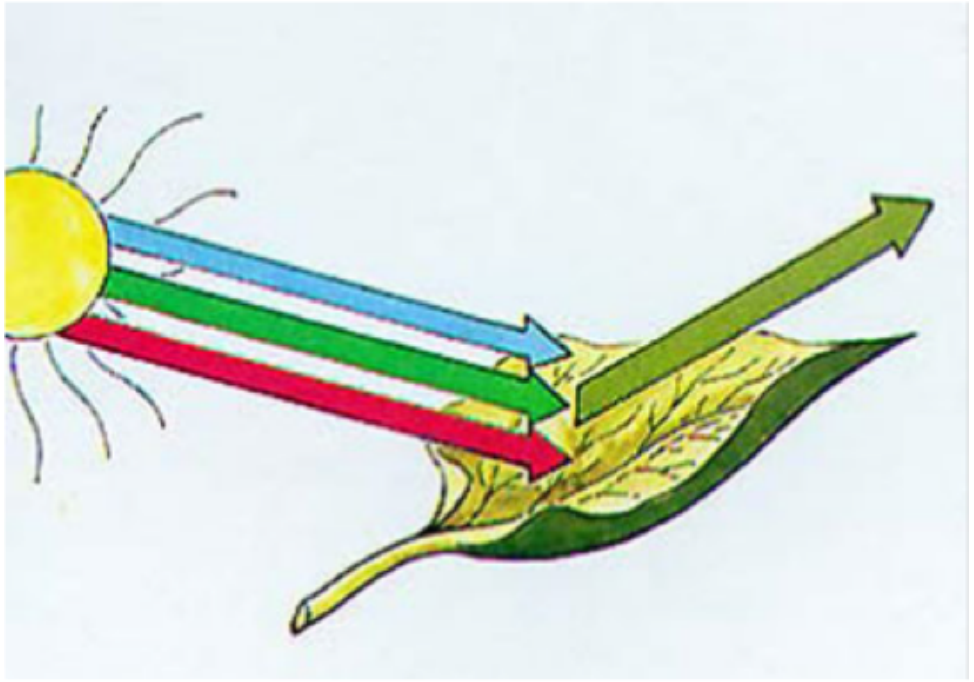
A light source emits a large number of photons, of different wavelengths. For example the figure below shows the "spectrum" of the light emitted by the sun.



Color vision

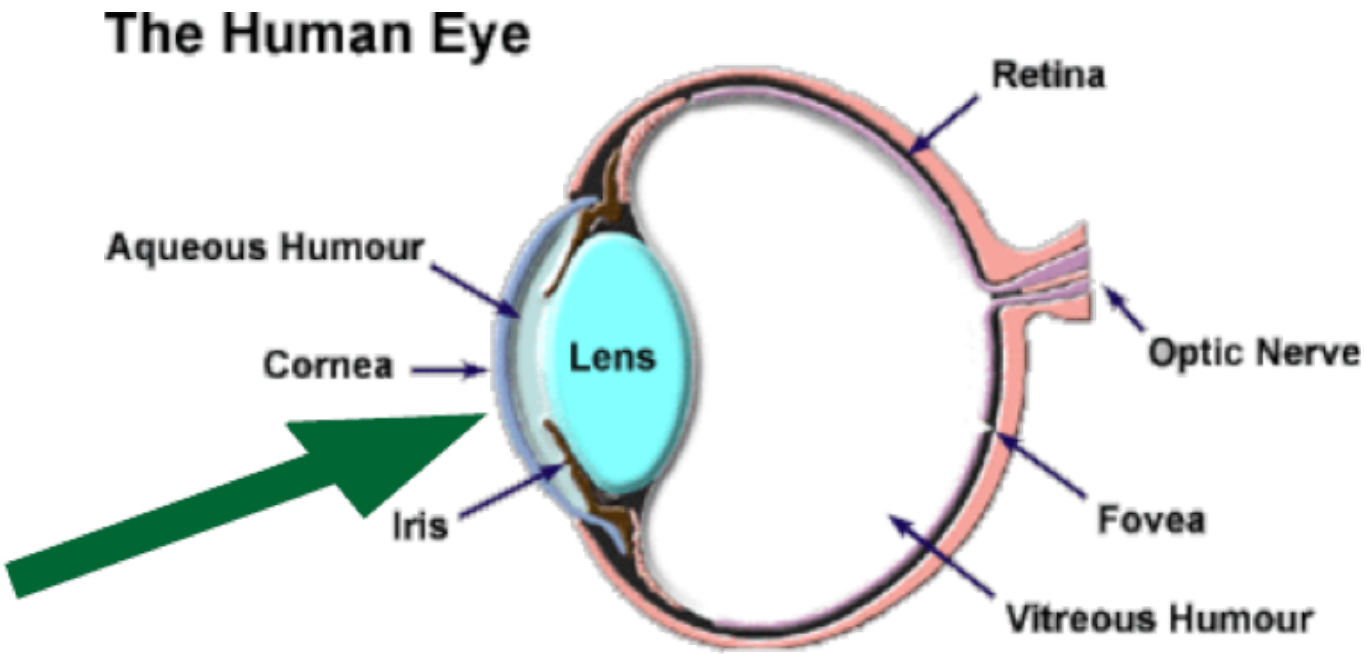
All the objects, depending on their composition, reflect or absorb the various wavelengths at different intensities.

Reflected photons determine the main colors of an element.



Color vision

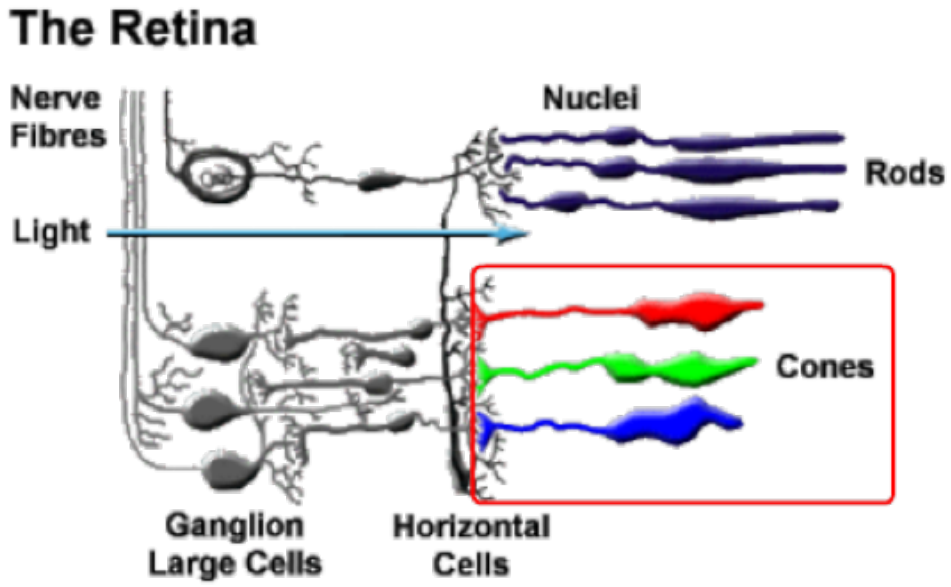
Reflected photons are focused through the lens of the human eye on the retina.



Color vision

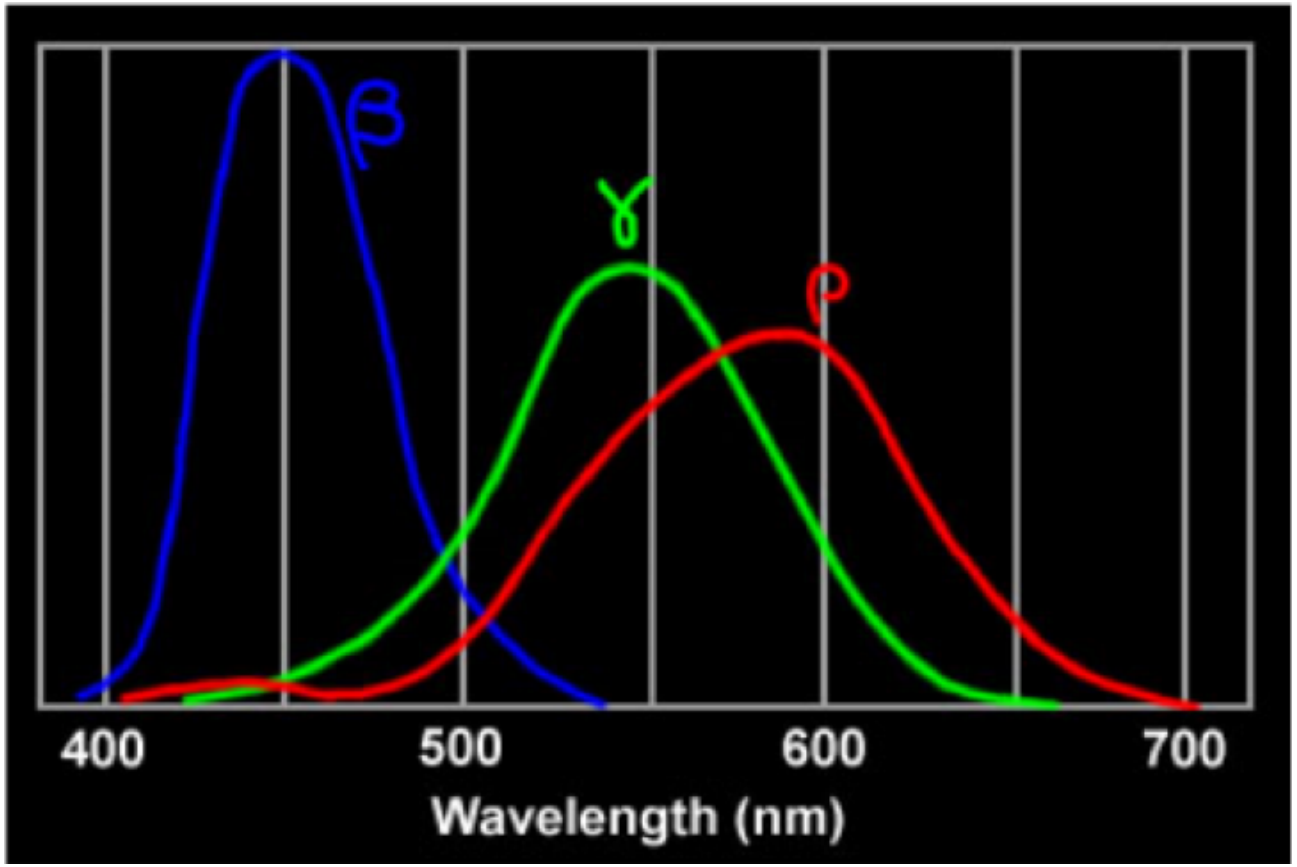
On the retina there are special cells called *rods*, sensible to light intensity, and *cones*, sensible to light color. These cells allow the brain to see.

There are three different types of cones (sometimes called ρ, β, γ), all equally distributed over the retina.



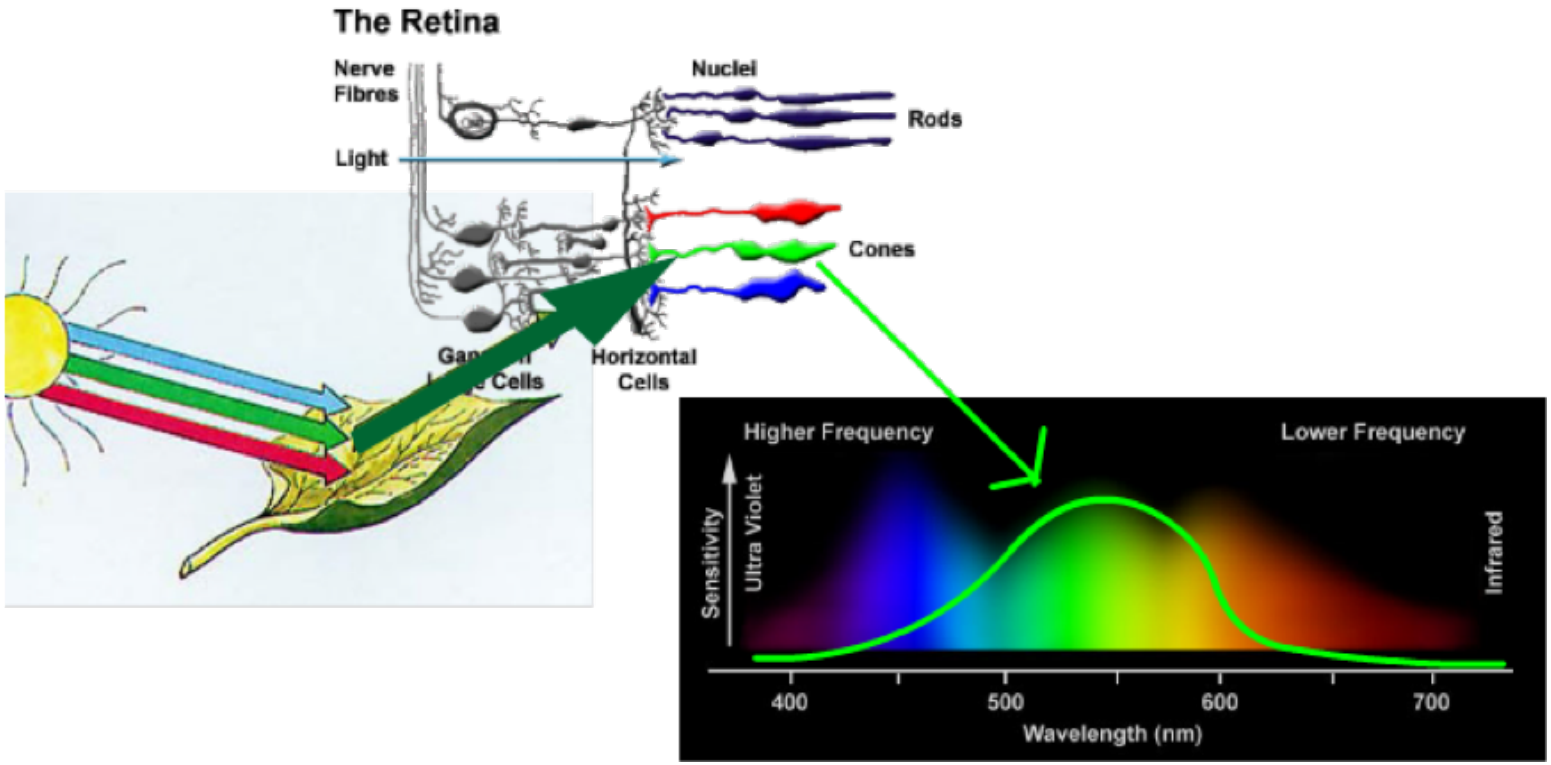
Color vision

Each type of cone is sensible to a different portion of the light spectrum.



Color vision

The brain combines the stimuli arriving from the different cones, and allows the vision of a given color.

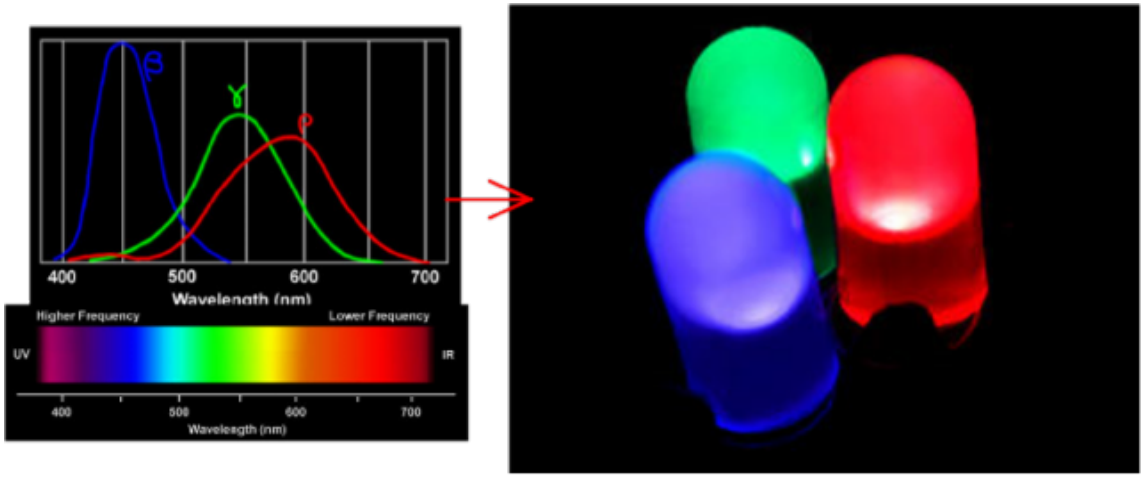


Color reproduction

Color reproduction uses the inverse principle of the human vision.

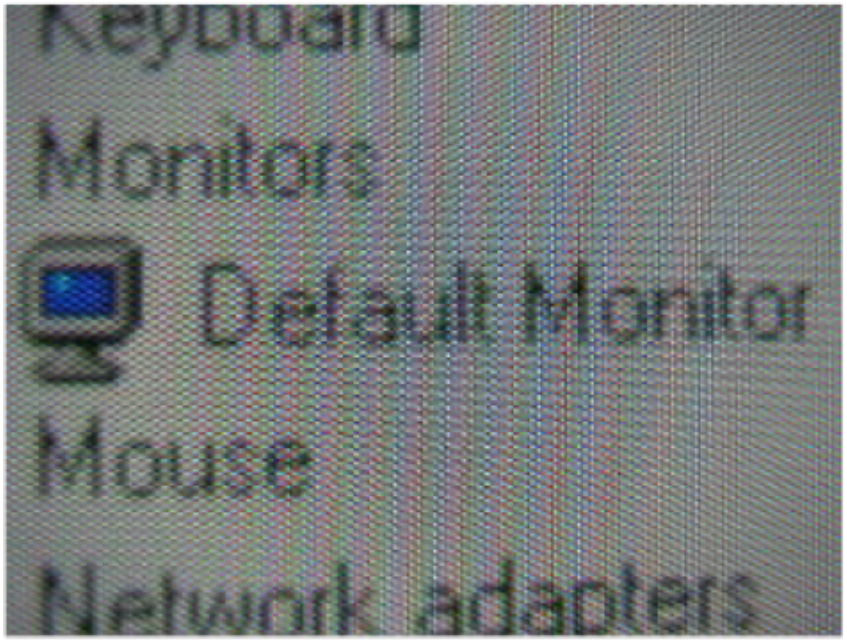
In particular it associates a different emitter for each color that a particular type of cone can perceive.

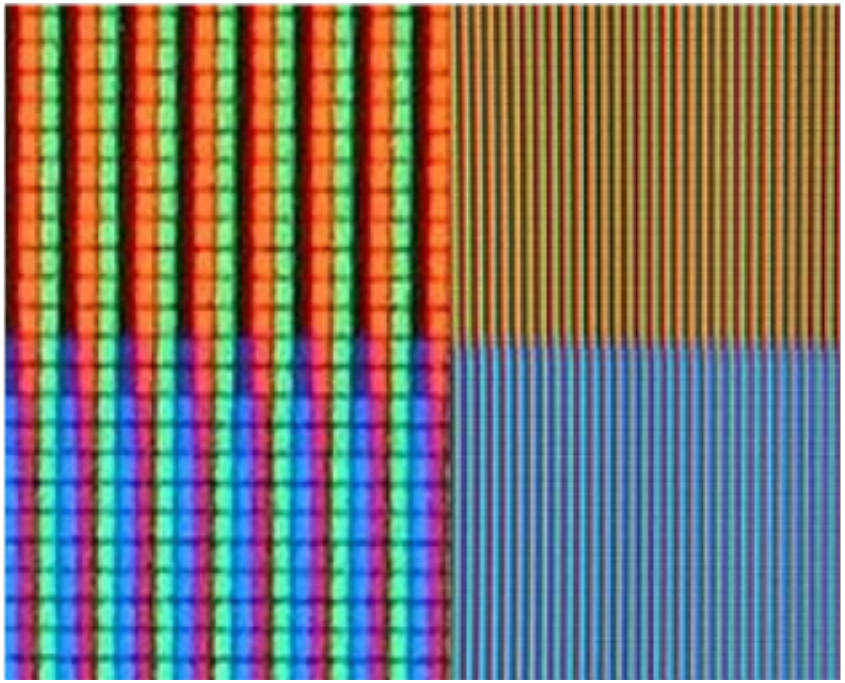
Since the wavelength perceived by the three types of cones mainly corresponds to the red, green and blue colors, different hues are constructed by mixing lights of these three primary colors.



Color reproduction

Looking a display from a sufficient distance, the human eye automatically mixes the primary hues, recreating the stimuli corresponding to a combined color.

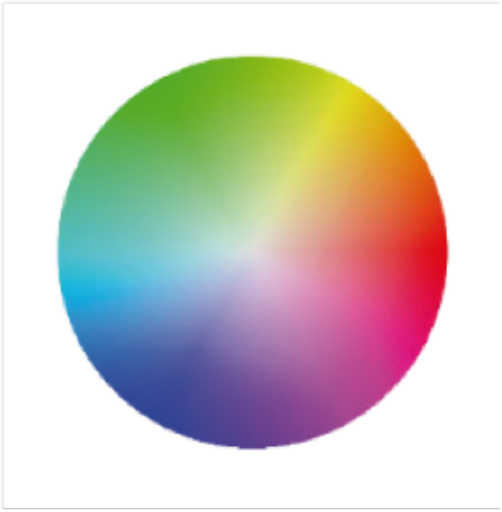




Color composition

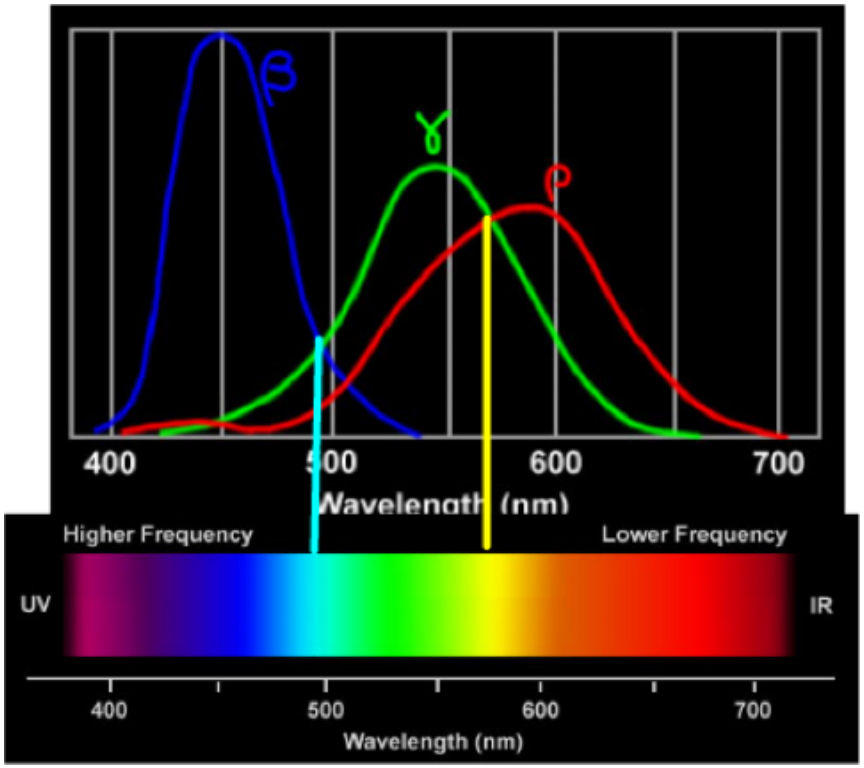
Mixing two primary colors, we obtain the *yellow*, *cyan* or *magenta*.
Mixing all the three primaries, we obtain white.

Mixing the three colors in different proportion, we can obtain all the possible hues.



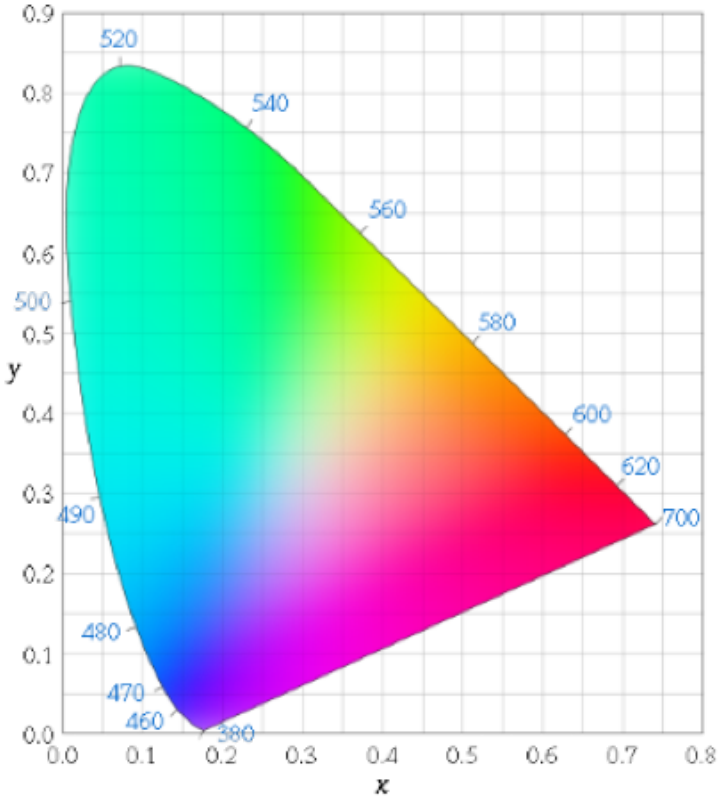
Color composition

If we compare the wavelengths of the photon to which the cones are sensible with the frequency spectrum, we can understand why cyan is obtained by mixing blue and green, and yellow is obtained mixing red and green.



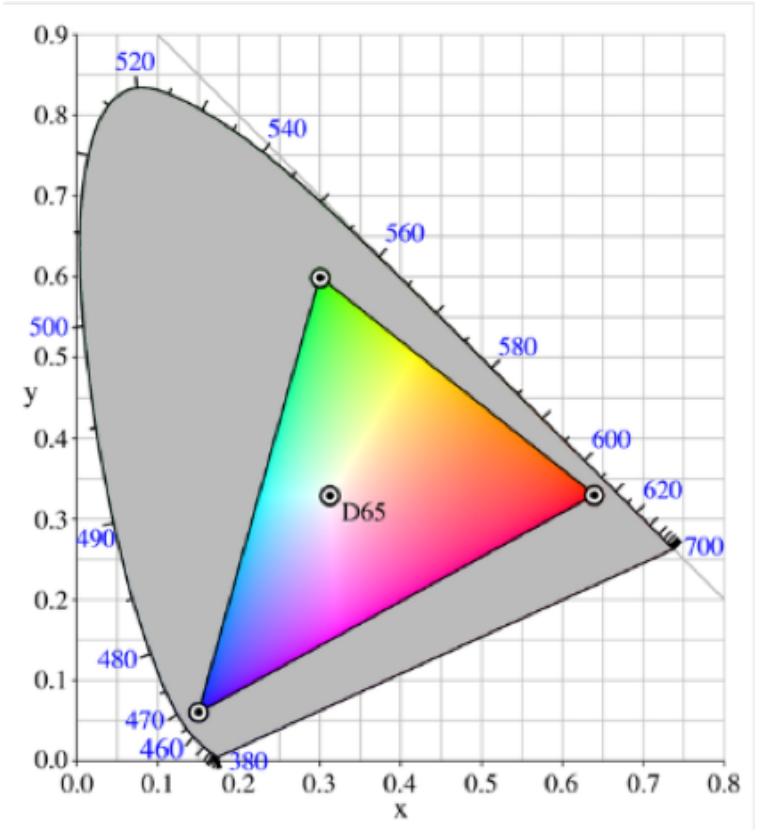
Color range

The spectrum that the human eye can see is quite large, and it is characterized by a parabolic shape.



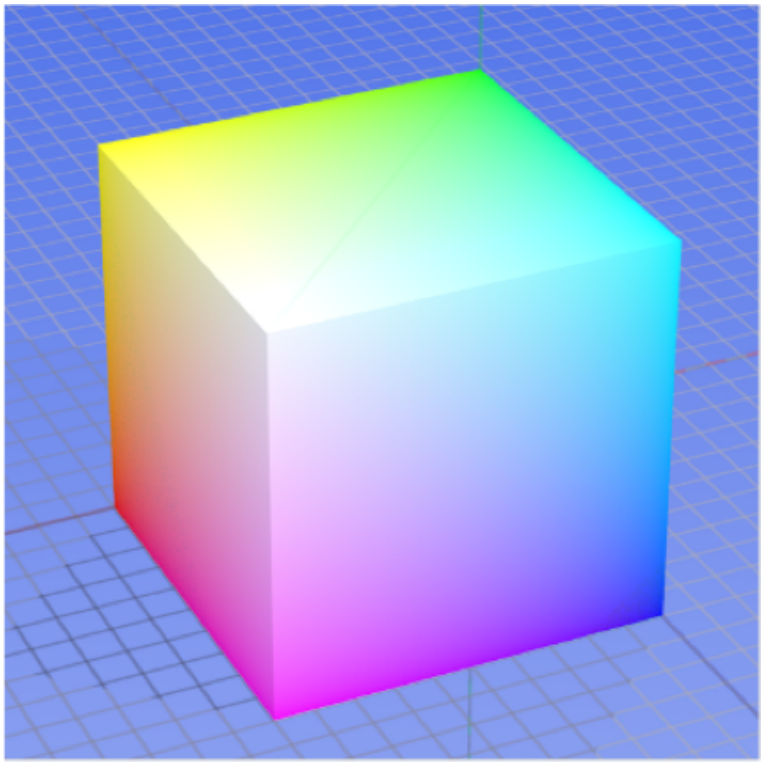
Color range

The spectrum that a monitor can reproduce is just a small portion of the one that the eye can see.



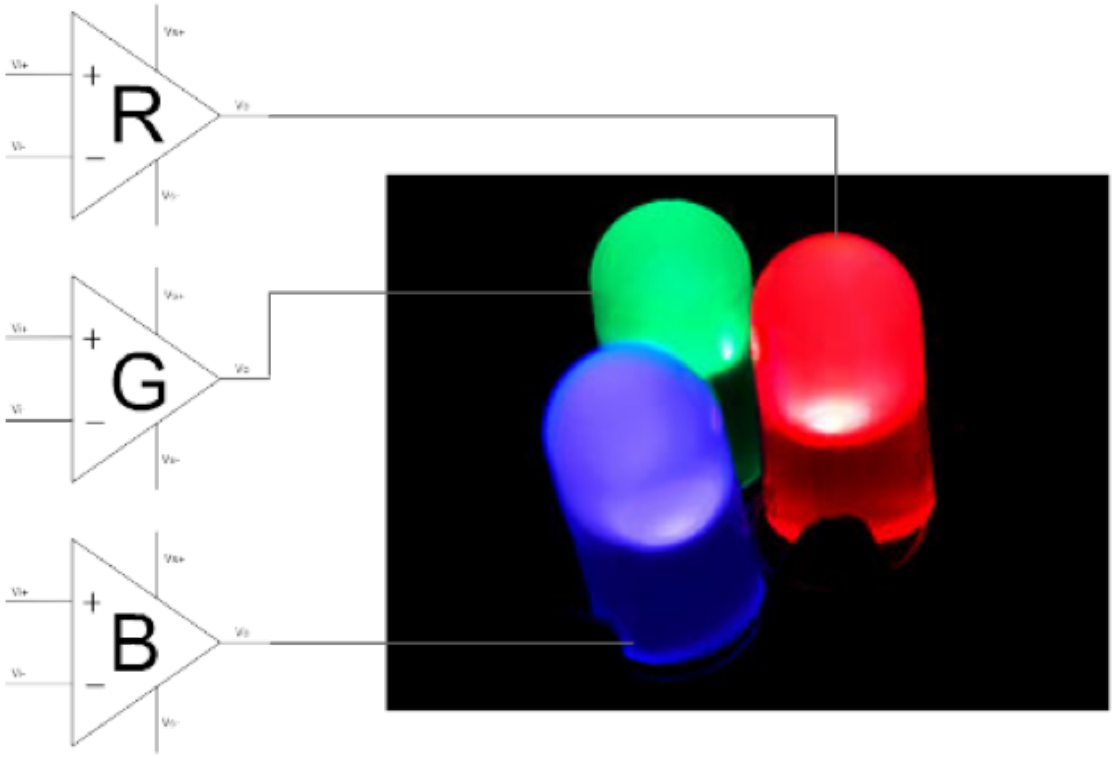
Color range

The set of the colors that a monitor can reproduce corresponds to a cube, where the red, green and blue components are placed on the x, y and z-axis.



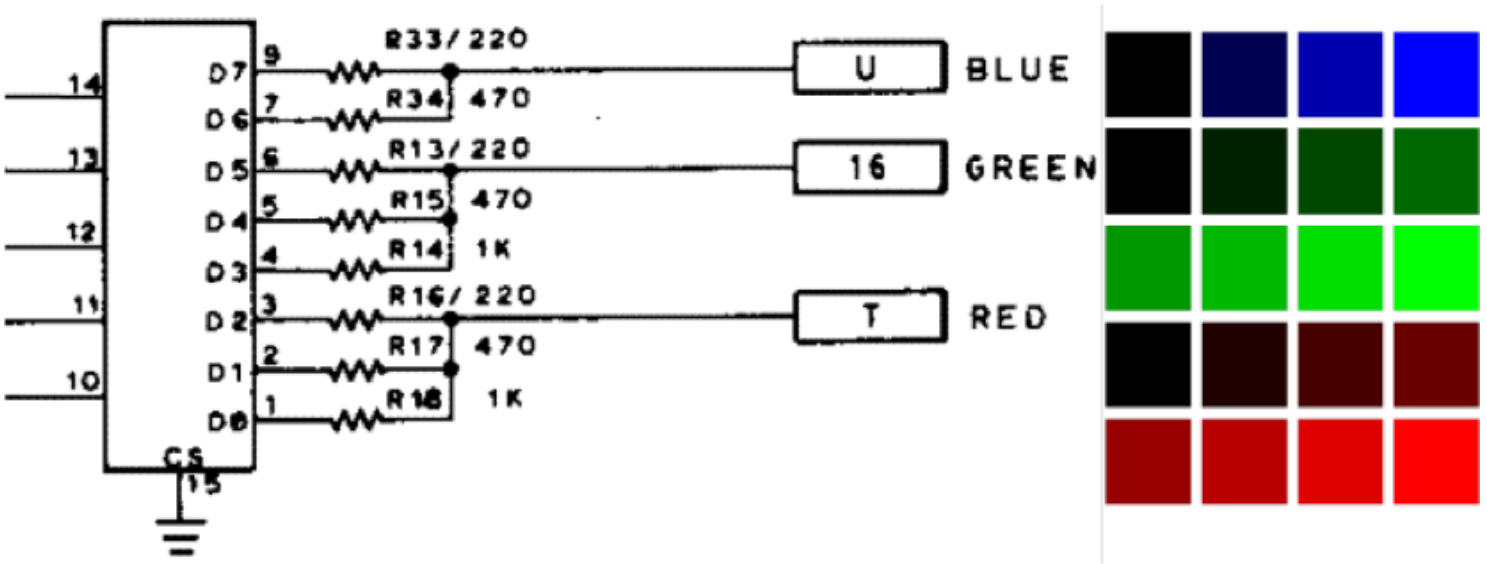
Color synthesis

The levels of red, green and blue translate into three electrical signals whose intensity controls the light emitted by the screen for each of the primary colors.



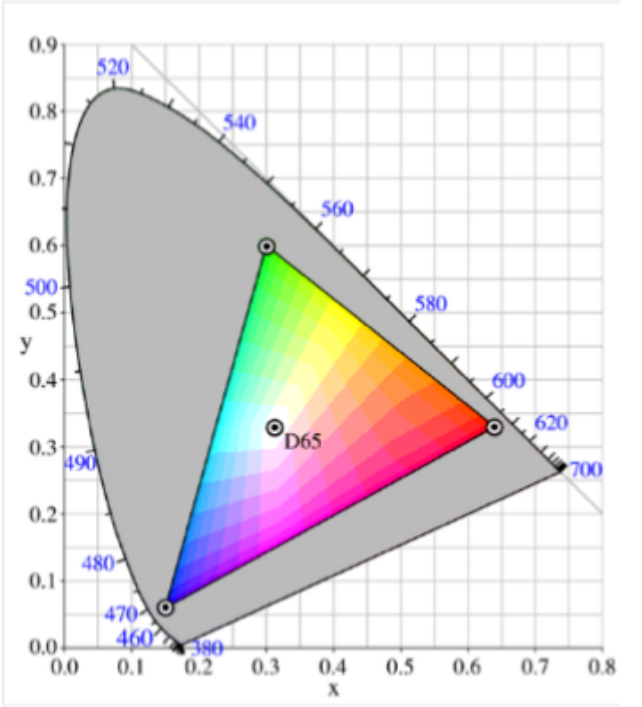
Color synthesis

Being the system digital, those signals are usually reproduced with a DAC conversion, and are thus quantized.



Color quantization

Quantization further reduces the number of colors that can be displayed.



Color profiles

Different monitors can associate different emitted frequencies of the spectrum to each primary color, and encode their level in a different ways. *Color profiles* allow to compensate for these effects, and produce a uniform behavior on different adapters.

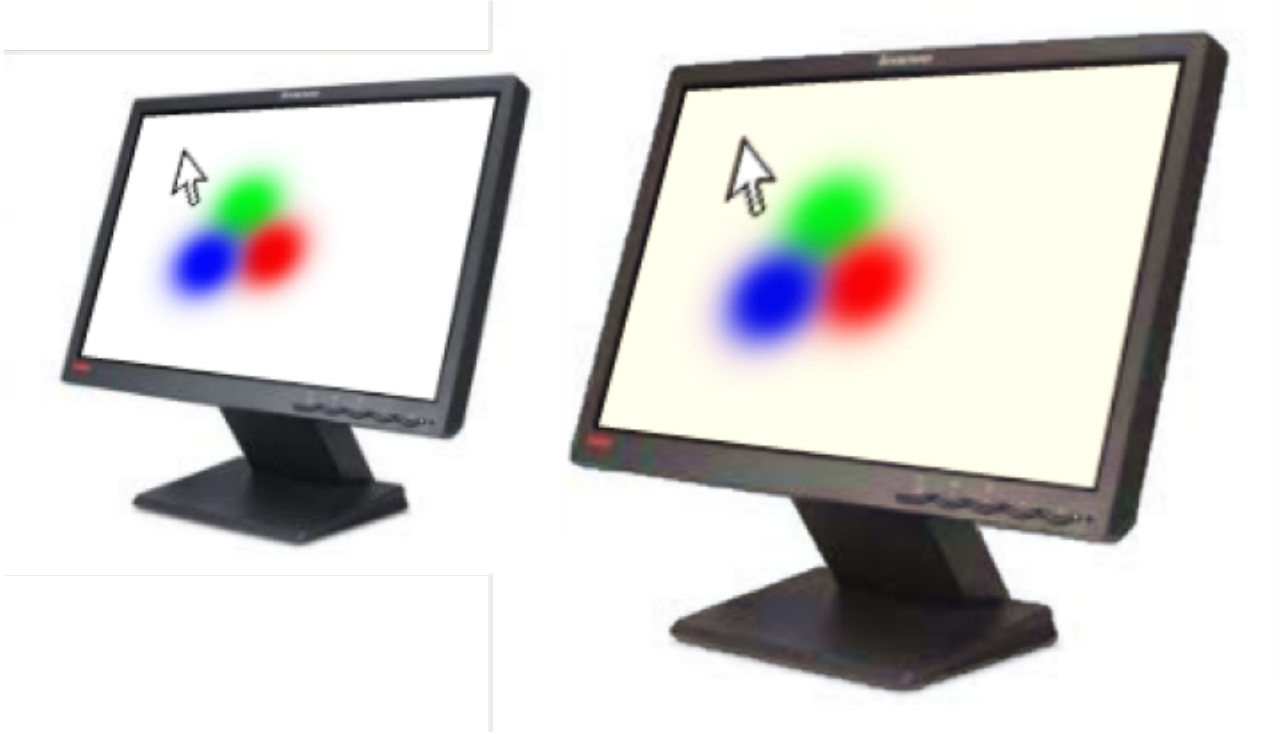


Image resolution

The *resolution* of an image defines the density of pixels in a metrical unit (generally measured in DPI – Dots Per Inch).

Display resolution

When we are dealing with raster graphic devices (as opposed to printed images), the density is relative to the monitor size.

Display resolution

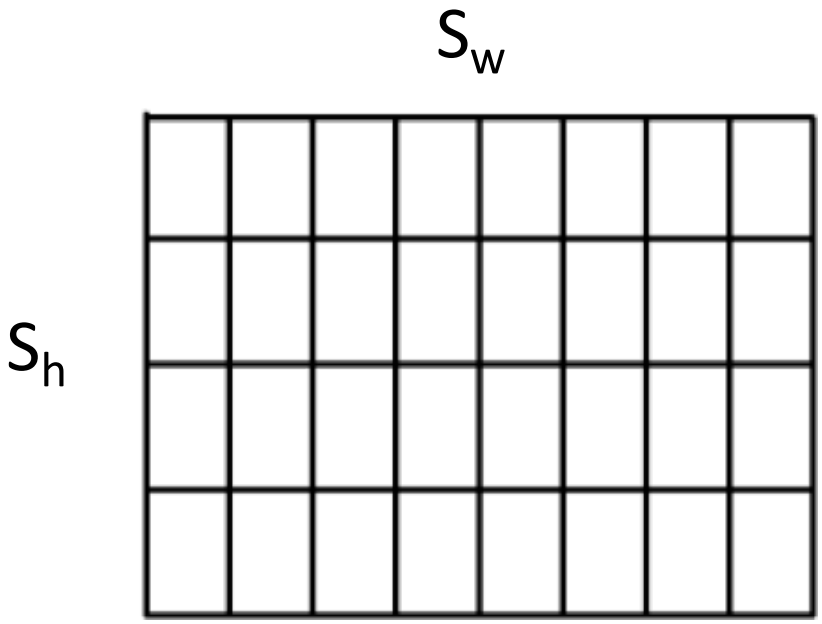
The resolution of a screen thus defines the number of pixels displayed on the horizontal (S_w) and vertical directions (S_h).

S_w

S_h

Pixel aspect ratio

Pixel might not have a square shape, and usually the horizontal resolution is different from the vertical one.



2D graphics primitives

Procedures that draw simple geometric shapes, based on a 2D coordinates system, are called *2D Graphics Primitives*.

A lot of theory has been developed about 2D graphics primitives.

However, current graphics adapters are able to draw everything supporting just three types of primitives:

- Points
- Lines
- Filled triangles

2D graphics primitives

Primitives draw and connect points on the screen identified a set of *coordinates*, defined with a vector of two components.

These coordinates are integer numbers whose units are *pixels*.

An important part of this course will deal with coordinate transformations and we will see a lot of different reference systems:

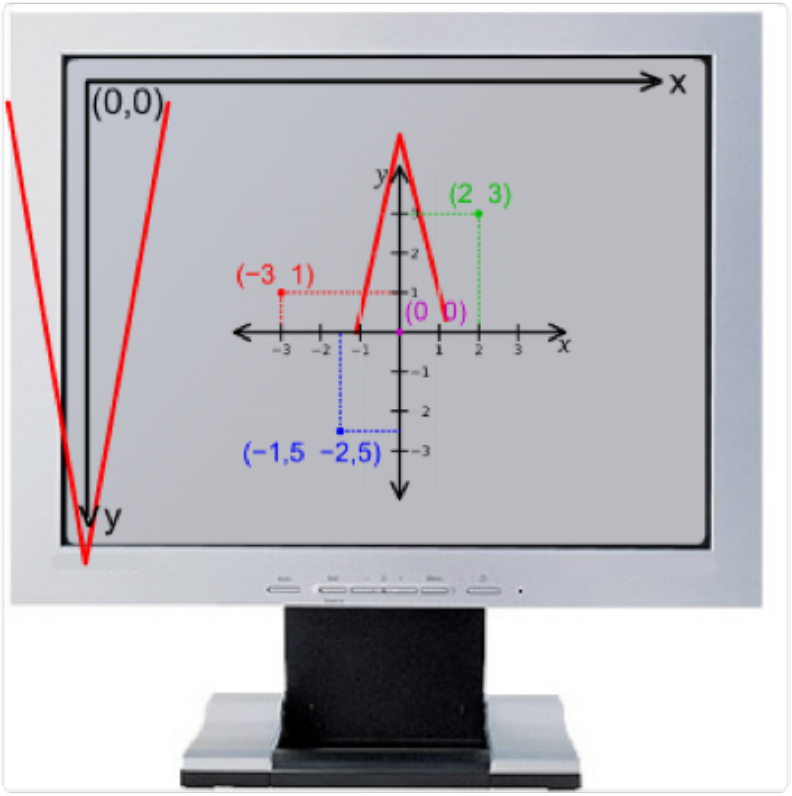
- This set of coordinates is called *pixel coordinates* (or *hardware* or *device coordinates*).

Pixel coordinates

The coordinate system is *Cartesian*, with the origin at the top-left corner, the *x-axis* increasing horizontally left to right, and the *y-axis* increasing vertically top to bottom.

Pixel coordinates

Note that the coordinate system is left-handed, since it has the y-axis in the opposite direction compared to the classical one.

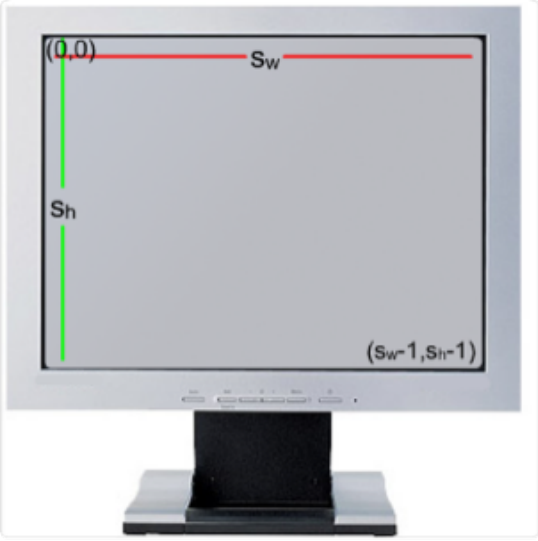


Pixel coordinates

The *x* and *y* integer values of the coordinates are in the range:

$0 <= x <= s_w - 1$ (where s_w is the screen width in pixel)

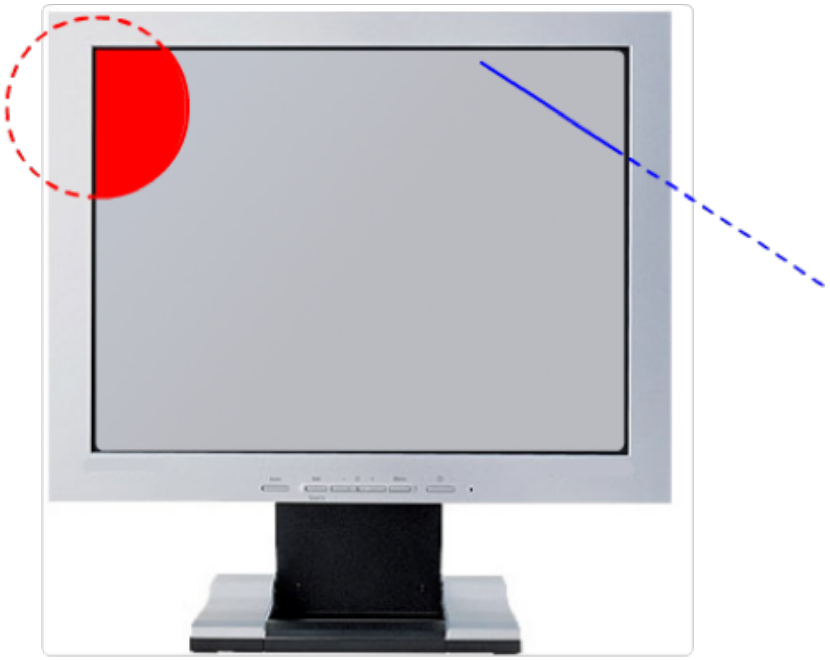
$0 <= y <= s_h - 1$ (where s_h is the screen height in pixel)



Clipping

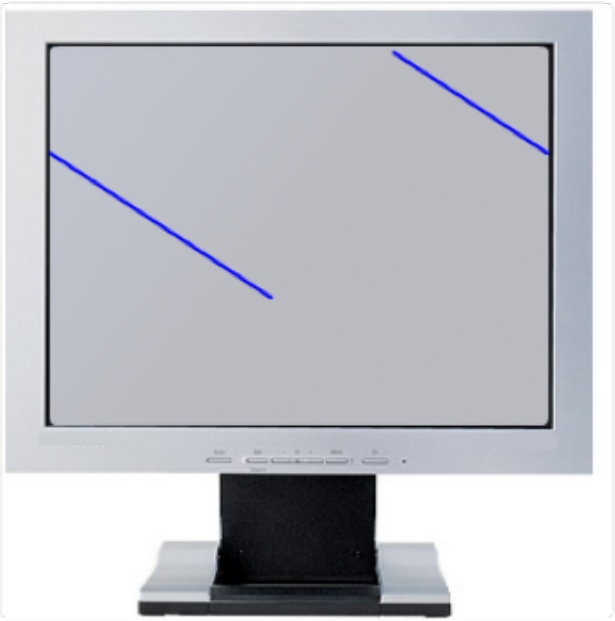
Coordinates might go outside the screen boundaries.

The process of cutting the primitives to display only their visible fraction is called *clipping*.



Clipping

If no clipping is performed, depending on the hardware, the primitive may either wrap-around, or write some un-allocated memory space (usually causing unrecoverable errors).



Normalized Screen Coordinates

Current displays are able to show different resolutions, and are available in different sizes and form factors.

Applications want to display the same content regardless of the resolution, the size and the shape of the screen. They also want to exploit all the features of the display at their best.

A special coordinate system, called *Normalized Screen Coordinates* is used to address points on screen in a device independent way.

Normalized Screen Coordinates

Screens might have different aspect ratio...

4 : 3

16 : 9

Normalized Screen Coordinates

... pixels might not have a square shape ...

Normalized Screen Coordinates

... but the goal is to represent correctly the same scene, regardless of the actual resolution, adding or subtracting features depending on the available space.

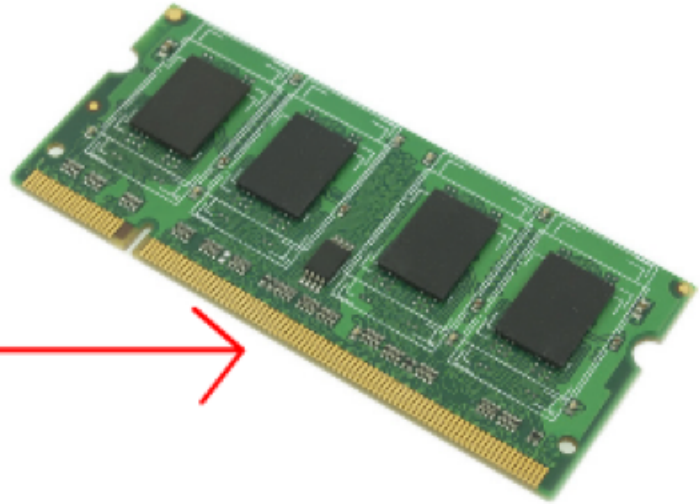
Gears of War (Microsoft Game Studios - 2006)

Normalized Screen Coordinates

The same applies also to windows in conventional O.S., which might be freely resized by the user.

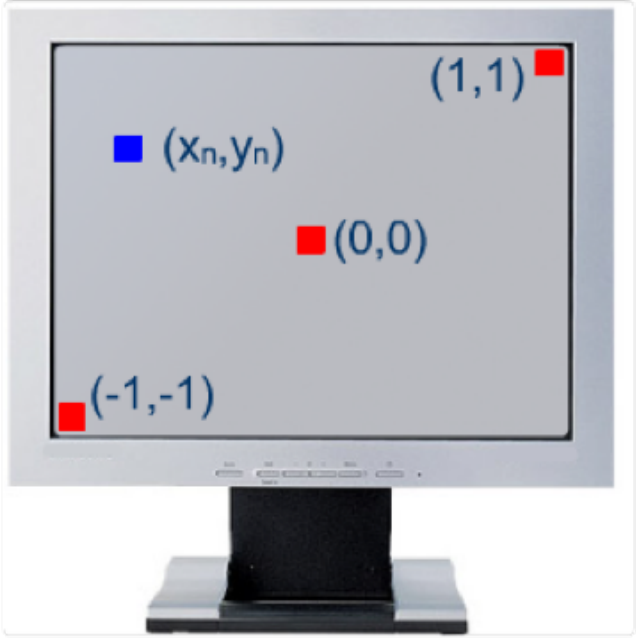
Normalized Screen Coordinates

Many applications render the images to memory: also in this case, the proportions of the display area can be arbitrary.



Normalized Screen Coordinates

Normalized Screen Coordinates are a Cartesian coordinate system, where *x* and *y* values range between two canonical values (generally between *-1* and *1*, but other standards such as *0* and *1* also exists), and axes oriented along a specific direction.

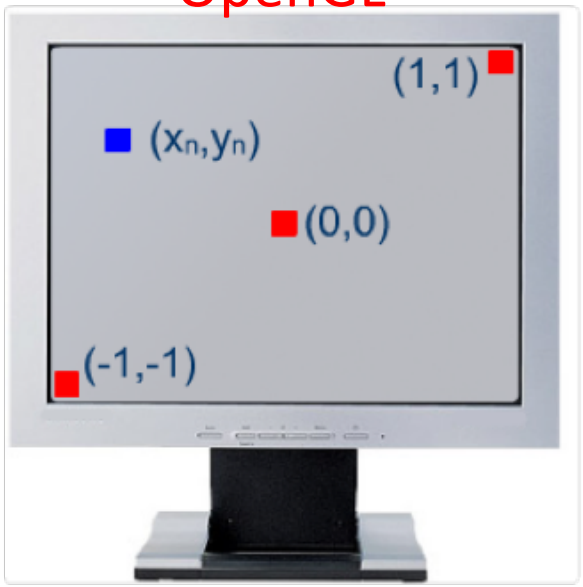


Normalized Screen Coordinates

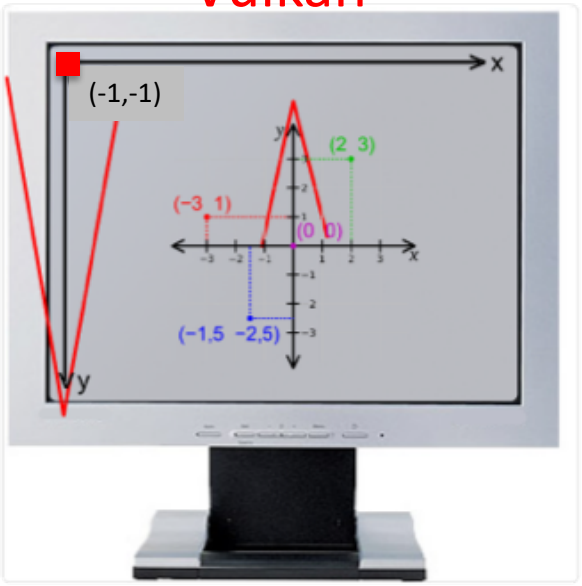
For example, both OpenGL and Vulkan uses normalized screen coordinates in the [-1, 1] range.

However, OpenGL has the *y-axis* going up, while Vulkan follows the convention of pixel coordinates with the *y-axis* pointing down.

OpenGL



Vulkan



Normalized Screen Coordinates

If we know that the screen (or window, or memory area) resolution is s_w x s_h pixels, then pixel coordinates (x_s, y_s) can be derived from the normalized screen coordinates (x_n, y_n) with a simple relation. For example, for Vulkan we have:

$$x_s = (s_w - 1) * (x_n + 1) / 2 ;$$
$$y_s = (s_h - 1) * (y_n + 1) / 2 ;$$

We use $s_w - 1$ and $s_h - 1$ since pixel coordinates are respectively in the $[0, s_w - 1]$ and $[0, s_h - 1]$ range.

Normalized Screen Coordinates

As we introduced, screen buffers are usually accessed using their drivers and specific software libraries.

The primitives used to draw on the screen (or on one of its windows) automatically perform the computations to determine the correct pixels on the screen starting from normalized screen coordinates.

Software always uses Normalized Screen Coordinates, unless it works at a very low level in which it has to directly communicate with the frame buffer.

Points

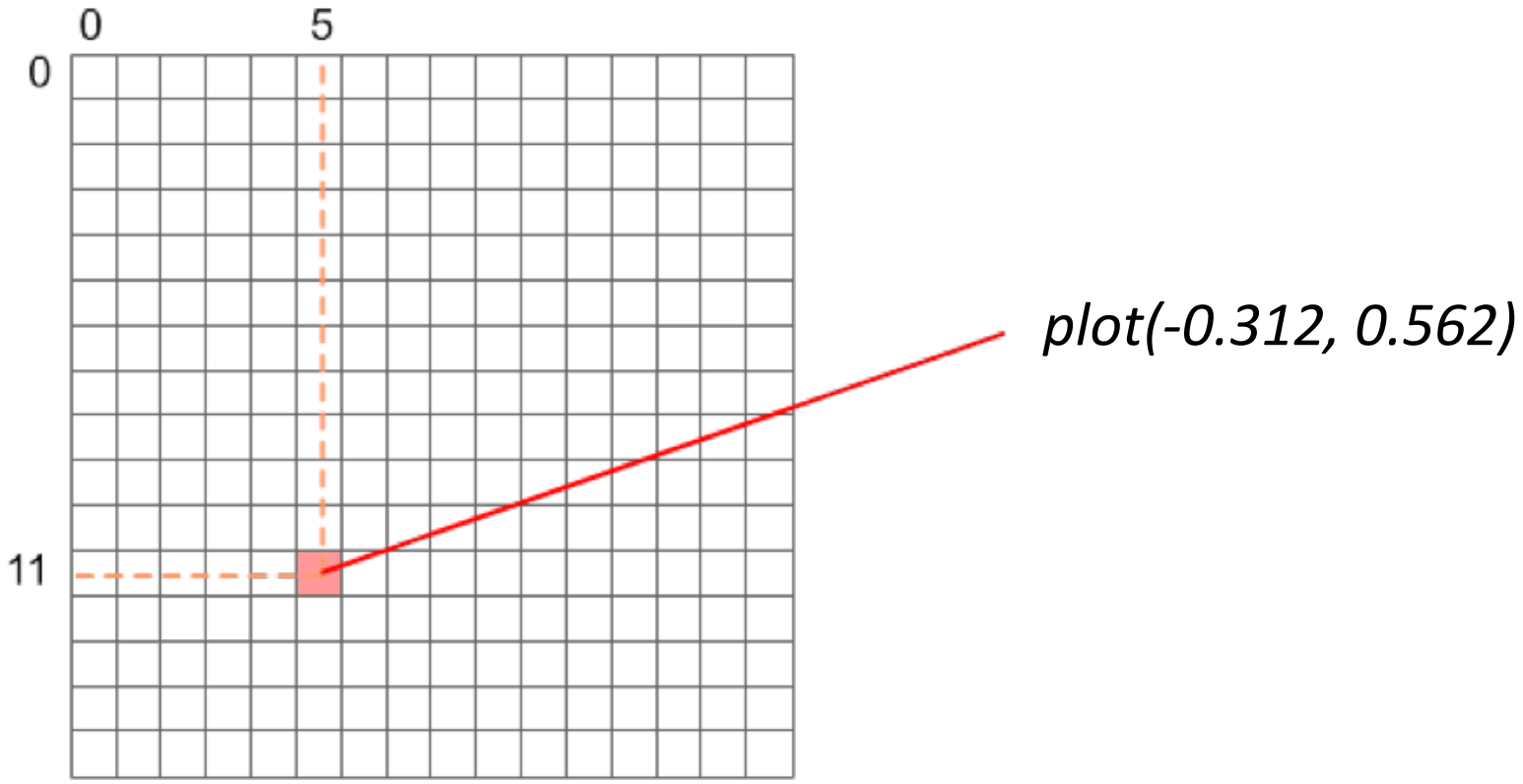
The simplest 2D primitive is the *point*.

Drawing a point corresponds to setting a pixel on the screen, in a given position and with a given color.

The graphic primitive that draws a point is usually called *plot()*.

Points

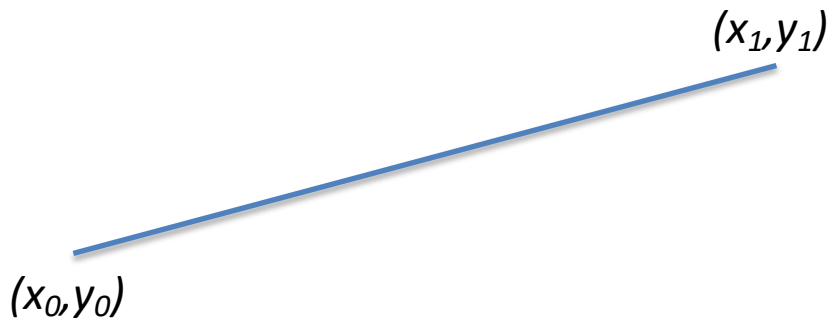
The primitive takes as parameters the coordinates of the pixel it needs to set.



Lines

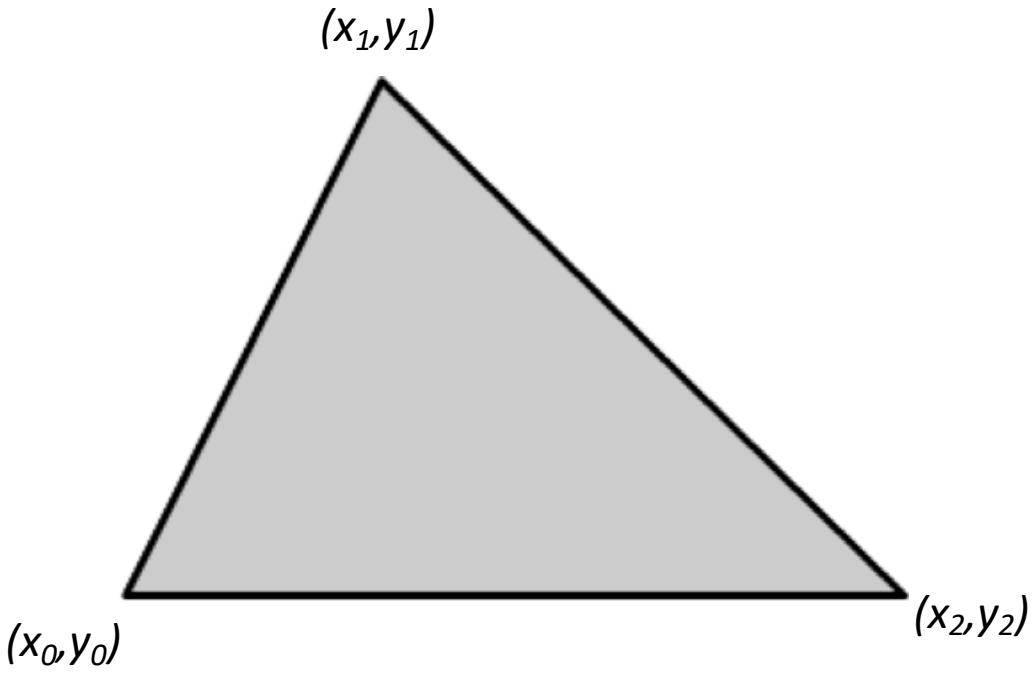
The line primitive connects two points on the screen with a straight segment.

It requires the coordinates (x_0,y_0) of the starting point and (x_1,y_1) of the end.



Triangles

As we will see, filled triangles are the basis of 3D computer graphics. They are specified with the coordinates of their three vertices (x_0,y_0), (x_1,y_1) and (x_2,y_2).



Marco Gribaudo

Associate Professor

CONTACTS

Tel. +39 02 2399 3568

marco.gribaudo@polimi.it

https://www.deib.polimi.it/eng/home-page

(Remember to use the phone, since mails might require a lot of time to be answered. Microsoft Teams messages might also be faster than regular mails)