



Ejercicio 1 – Espacios de color

Este ejercicio tiene como objetivo que os empecéis a familiarizar con OpenCV, el trabajo a bajo nivel de los píxeles de una imagen y su librería para cambios de espacios de color. Todo ello visto en el Tema 2: Formación de la imagen.

La defensa del ejercicio se hará en clase, y hay que entregar un archivo **cv_node_p1.cpp** con el código generado que deberás subir al Aula Virtual.

Puntos totales posibles del ejercicio: 10

Instrucciones

Utilizando el simulador con Tiago, y partiendo del ejemplo **02_02_pixels**, se pide crear un programa que muestre en la parte superior un texto como el que muestra la figura:



Para incluir un texto en una imagen se proporciona la siguiente función:

```
// Write text in an image
cv::putText(image, text, cv::Point(10, 20),
            cv::FONT_HERSHEY_SIMPLEX, 0.5, cv::Scalar(0, 0, 255));
```

Visión Artificial

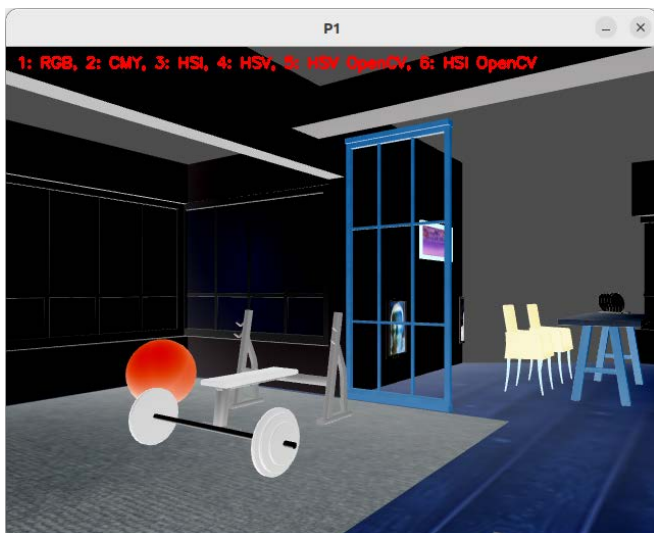
Mientras que para leer la tecla que se ha pulsado sobre la imagen, se usa una de las siguientes funciones:

```
// Read key
int key = cv::waitKey(int delay); // waiting time for a key pressed
int key = cv::pollKey();
```

Dentro de esta función, o en otra aparte, se pide que en cada una de las 6 opciones se haga lo siguiente con la imagen capturada por Tiago:

- Opción 1: Mostrar la imagen en formato de color RGB.
- Opción 2: Mostrar la imagen en formato de color CMY haciendo la conversión píxel a píxel.
- Opción 3: Mostrar la imagen en formato de color HSI haciendo la conversión píxel a píxel.
- Opción 4: Mostrar la imagen en formato de color HSV haciendo la conversión píxel a píxel. Siendo V el valor máximo de los tres canales **$\max(\text{Canal_R}, \text{Canal_G}, \text{Canal_B})$**
- Opción 5: Mostrar la imagen en formato de color HSV utilizando la función **cvtColor** de OpenCV.
- Opción 6: Mostrar la imagen en formato de color HSI utilizando la función **cvtColor** de OpenCV para obtener los canales H y S, y calculando el canal I manualmente.

A continuación, se muestran unas capturas de lo que debe mostrar el resultado final para diferentes opciones:



Opción 1

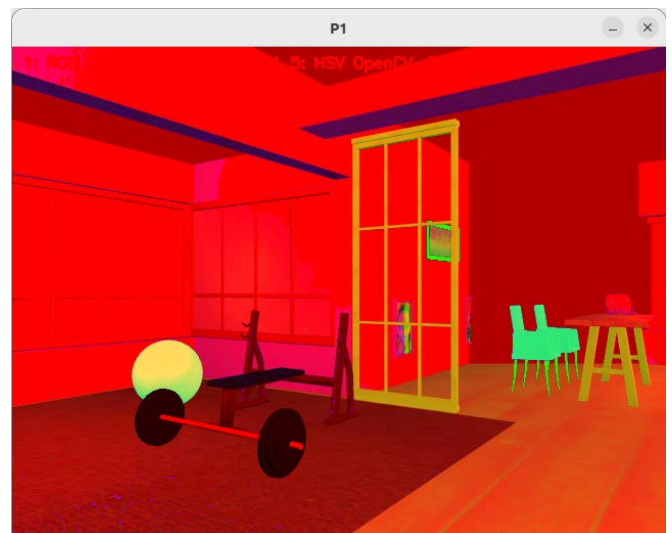


Ayuda

putText: [enlace](#)

waitKey: [enlace](#)

pollKey: [enlace](#)



Opción 4