

Introducción – Python

Imágenes, píxeles y paletas de colores

Auxiliar N°1

CC3501 – Modelación y Computación Gráfica para Ingenieros

Pablo Pizarro R. @ppizarror.com

Contenidos de hoy

- Refrescar la memoria (Python).
- Aprender a manipular imágenes utilizando numpy, dimensionando la parte matricial del problema y sus aplicaciones.

Imágenes

- Tipos de representación:
 - Raster
 - Imagen vectorial



Imágenes – Representación vectorial



- Modelo paramétrico de cada figura representada (Funciones, líneas, cuadrados, círculos, curvas).
- Resolución infinita, no hay pérdida de nitidez al hacer zoom.
- Puede tener menor tamaño de almacenado.
- Clásico: Formato SVG (Scalable Vector Graphics).

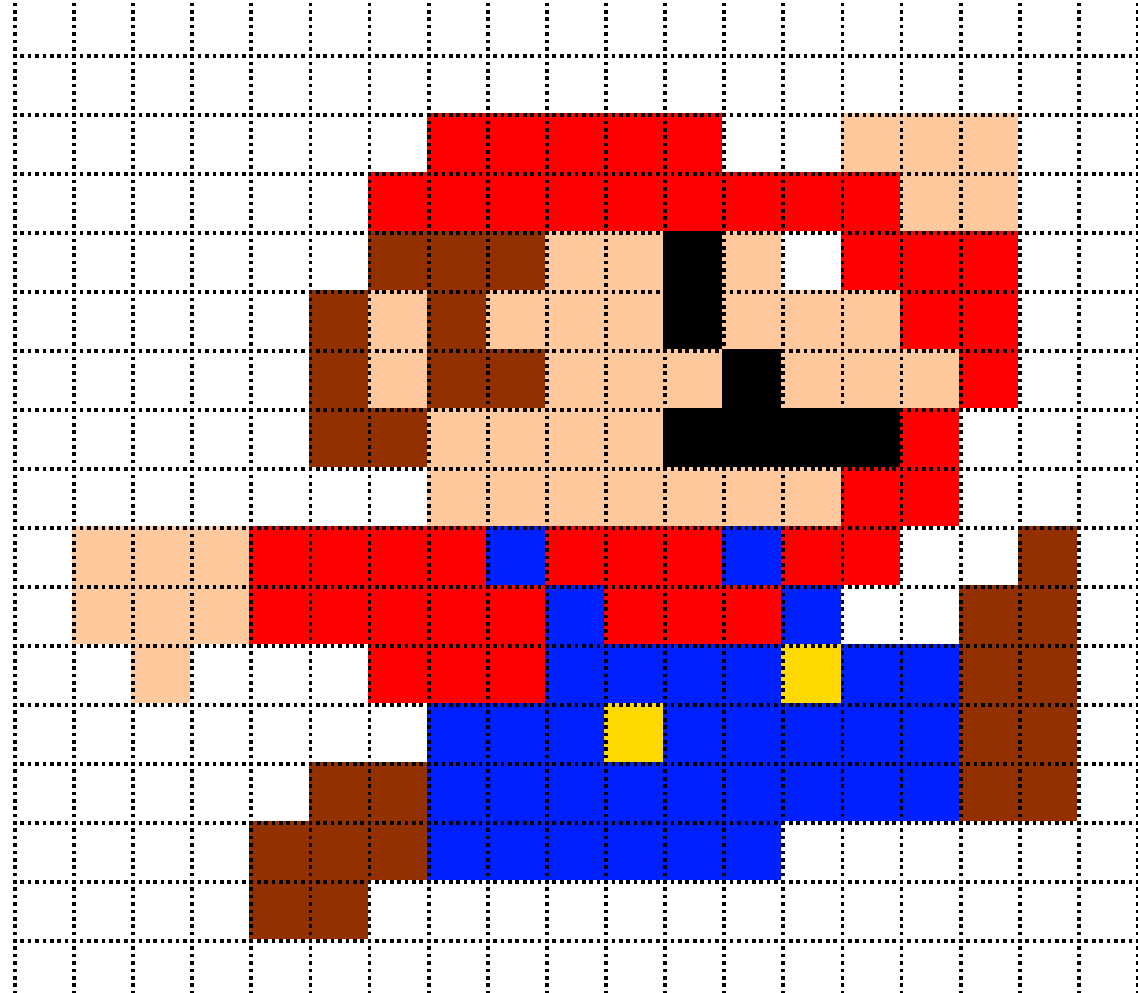
Imágenes – Representación vectorial



Imágenes – Representación raster



- Una imagen se representa utilizando una matriz 2D donde cada celda tiene asignado un color.
- Cada celda se conoce como pixel.

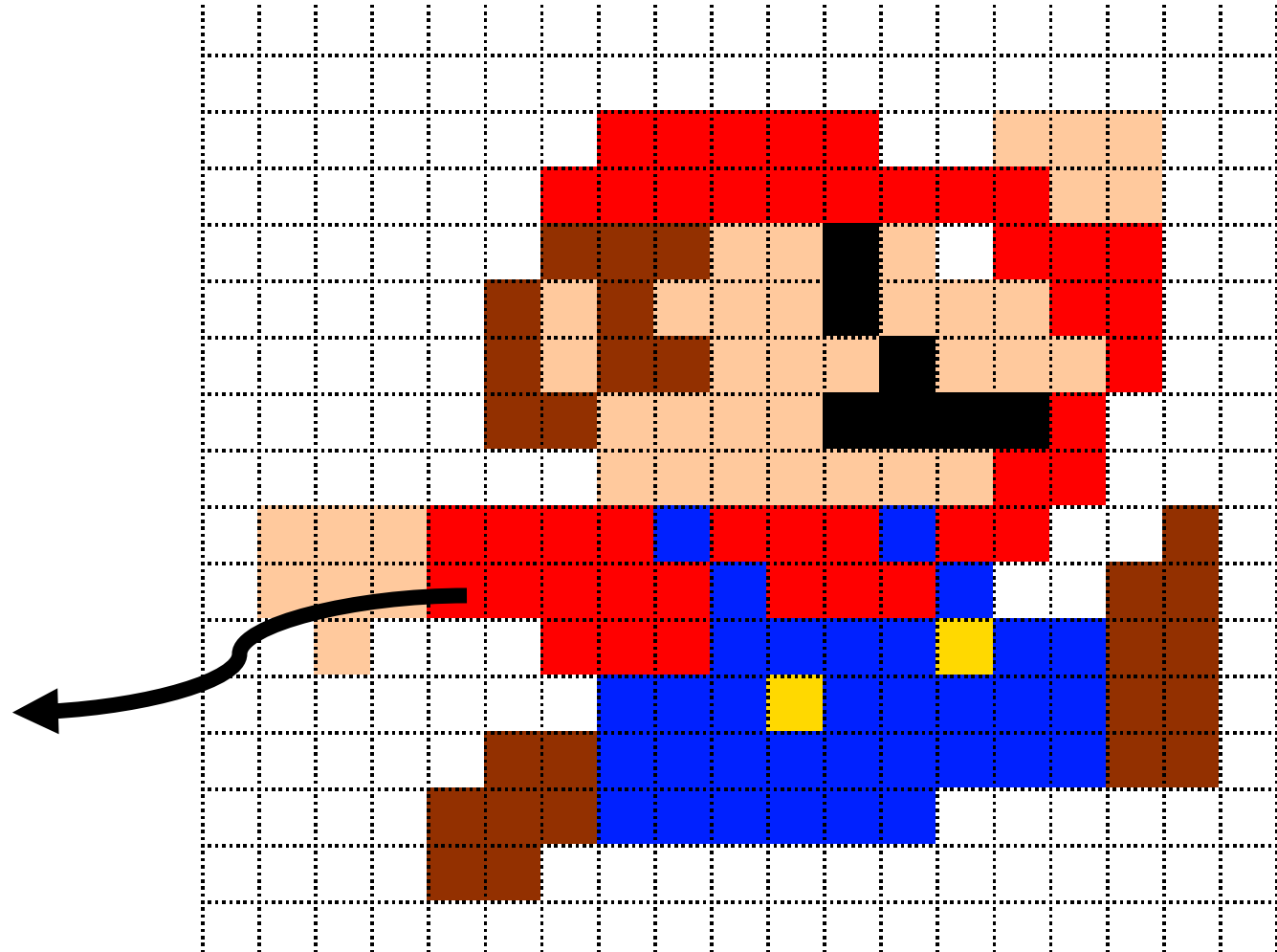


Imágenes – Representación raster



- Una imagen se representa utilizando una matriz 2D donde cada celda tiene asignado un color.
- Cada celda se conoce como pixel.

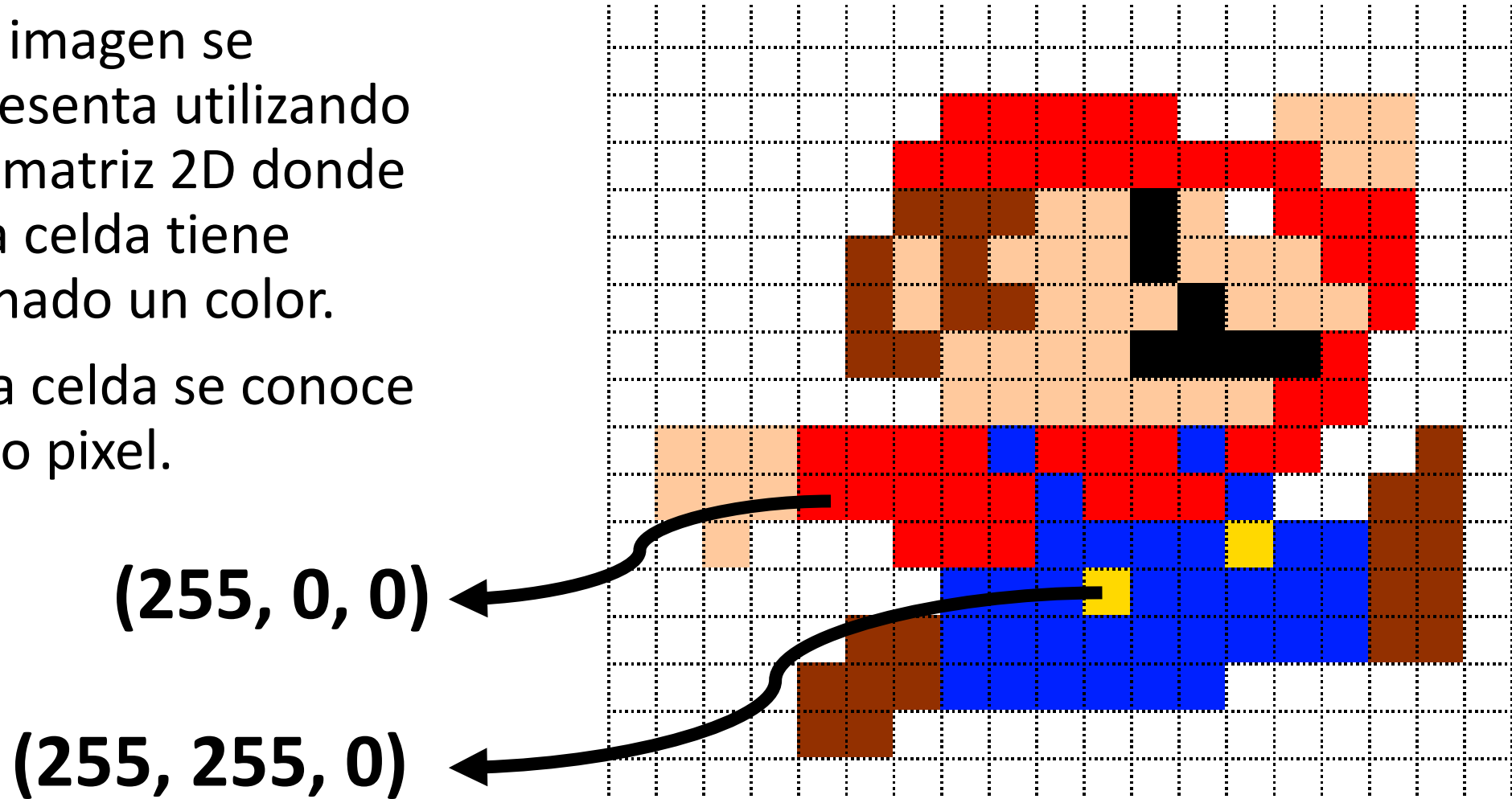
(255, 0, 0)



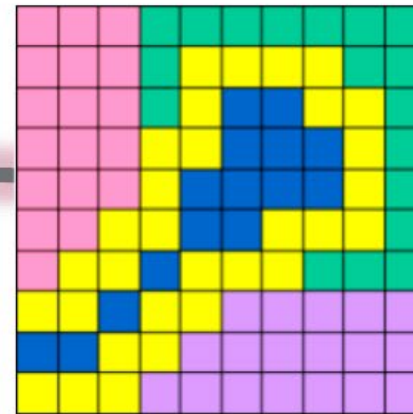
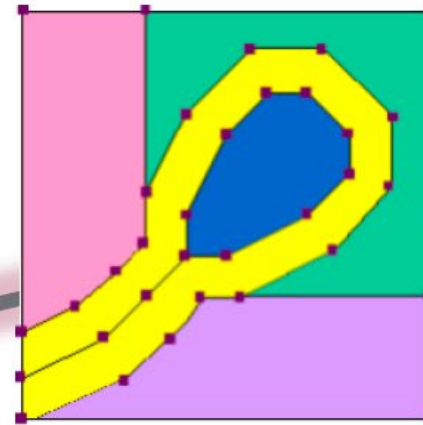
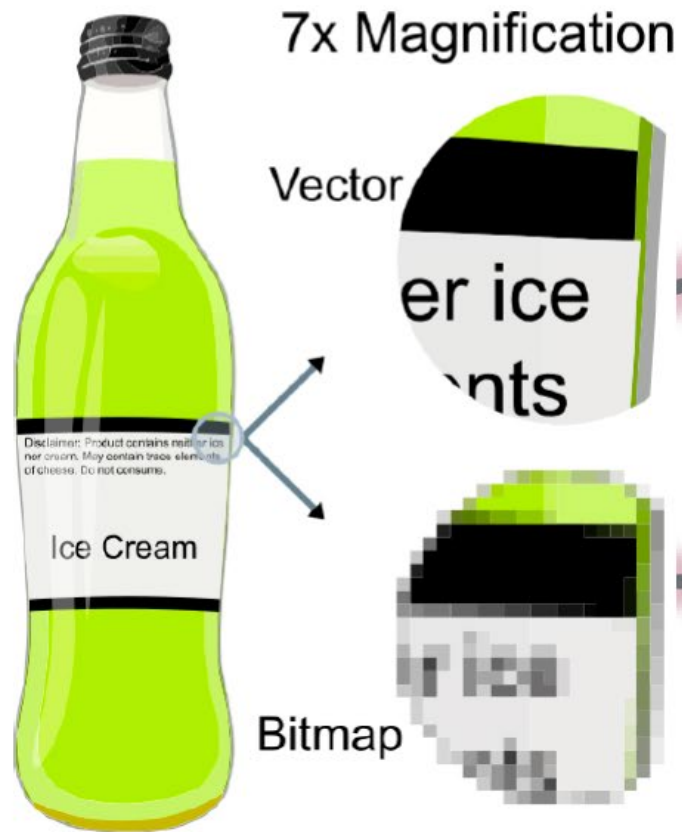
Imágenes – Representación raster



- Una imagen se representa utilizando una matriz 2D donde cada celda tiene asignado un color.
- Cada celda se conoce como pixel.



Imágenes – Raster vs vector



Problema de hoy

- Reforzar contenidos de Python.
- ¿Qué necesitamos?
 - Python 3 (3.7)
 - Numpy (Librería de Python)
 - Matplotlib (Librería de Python)
- Numpy permite el manejo de matrices, similar a las características que ofrece Matlab.
- Matplotlib es una librería que permite realizar gráficos.

Problema de hoy

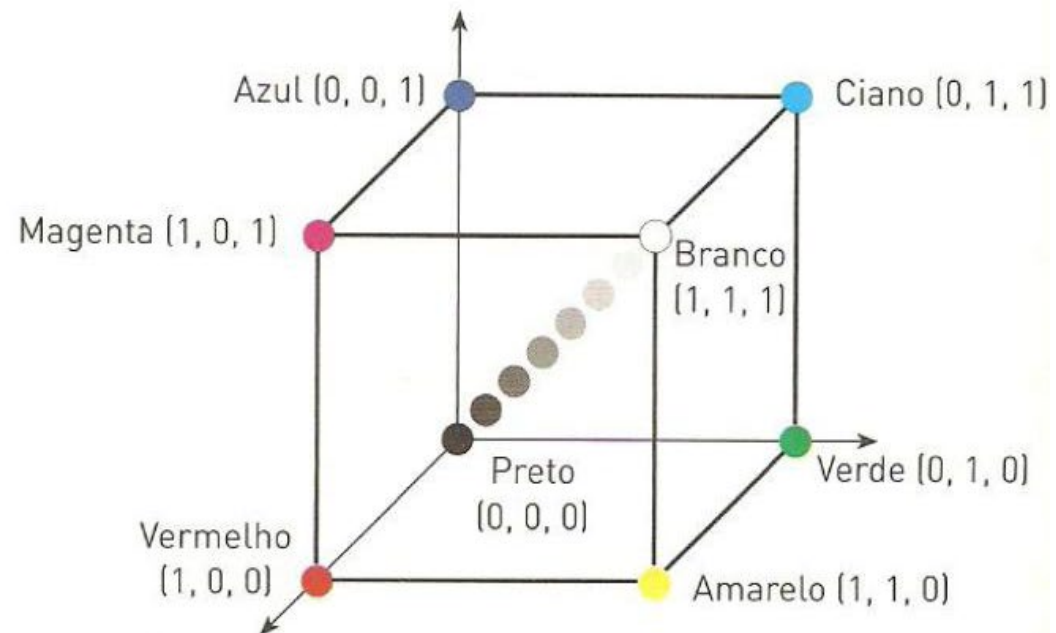
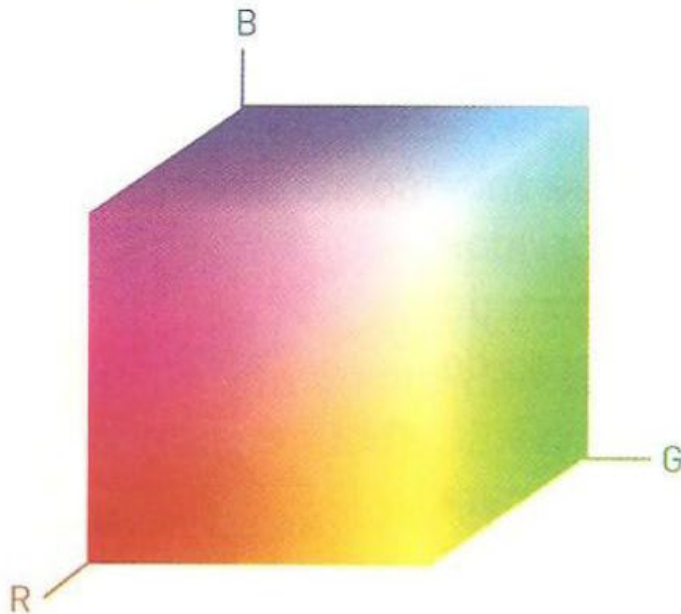
- Revisar `ex_color_palette.py`: Permite crear una paleta de colores a partir de una imagen.
- Para cada pixel de una imagen, si un color no existe en la paleta este se añade y se asocia la posición del color en la paleta al pixel.
- Recordar que un color se define con sus componentes rojo, verde y azul. En numpy los colores están entre (0,0,0) y (1,1,1).
 - Ej, (0.12, 0.4, 0), etc.

Problema de hoy

- Galería de ejemplos matplotlib: <https://matplotlib.org/gallery.html>
- Animaciones con paletas de colores de Mark Ferrari:
<http://www.effectgames.com/demos/canvascycle/>
- Reducir el número de colores en una imagen en Photoshop y Gimp:
<http://www.mapwel.net/manual/2150colorreduction.htm>

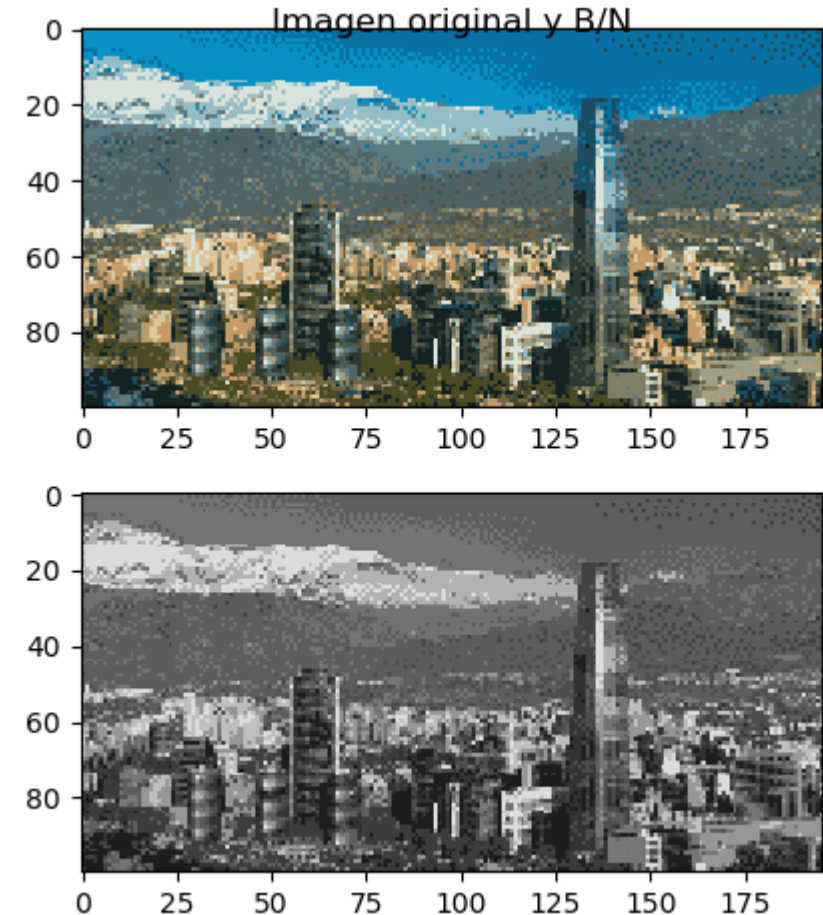
Problema de hoy – P1

- 1. Calcule una paleta de colores en blanco y negro (mismo valor en RGB) utilizando el promedio de las componentes RGB de la paleta de colores de la imagen. Presente la imagen en escala de grises resultante. (FACIL)



Problema de hoy – P1

- 1. Calcule una paleta de colores en blanco y negro (mismo valor en RGB) utilizando el promedio de las componentes RGB de la paleta de colores de la imagen. Presente la imagen en escala de grises resultante.

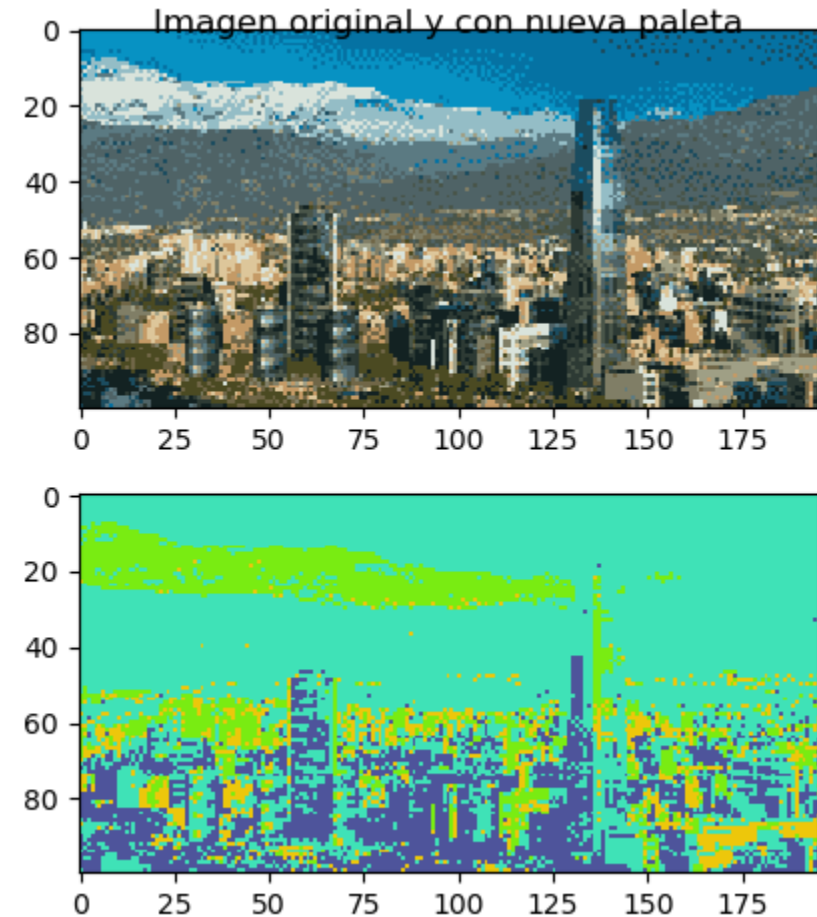


Problema de hoy – P2

- 2. Utilizando la escala de grises anterior. Escoja 3 valores v_1 , v_2 y v_3 entre 0 y 1 en orden estrictamente creciente. Escriba una nueva función que modifique la paleta de colores, asociando distintos colores para valores entre 0 y v_1 , entre v_1 y v_2 , entre v_2 y v_3 , y entre v_3 y 1. (FACIL)

Problema de hoy – P2

- 2. Utilizando la escala de grises anterior. Escoja 3 valores v_1 , v_2 y v_3 entre 0 y 1 en orden estrictamente creciente. Escriba una nueva función que modifique la paleta de colores, asociando distintos colores para valores entre 0 y v_1 , entre v_1 y v_2 , entre v_2 y v_3 , y entre v_3 y 1.

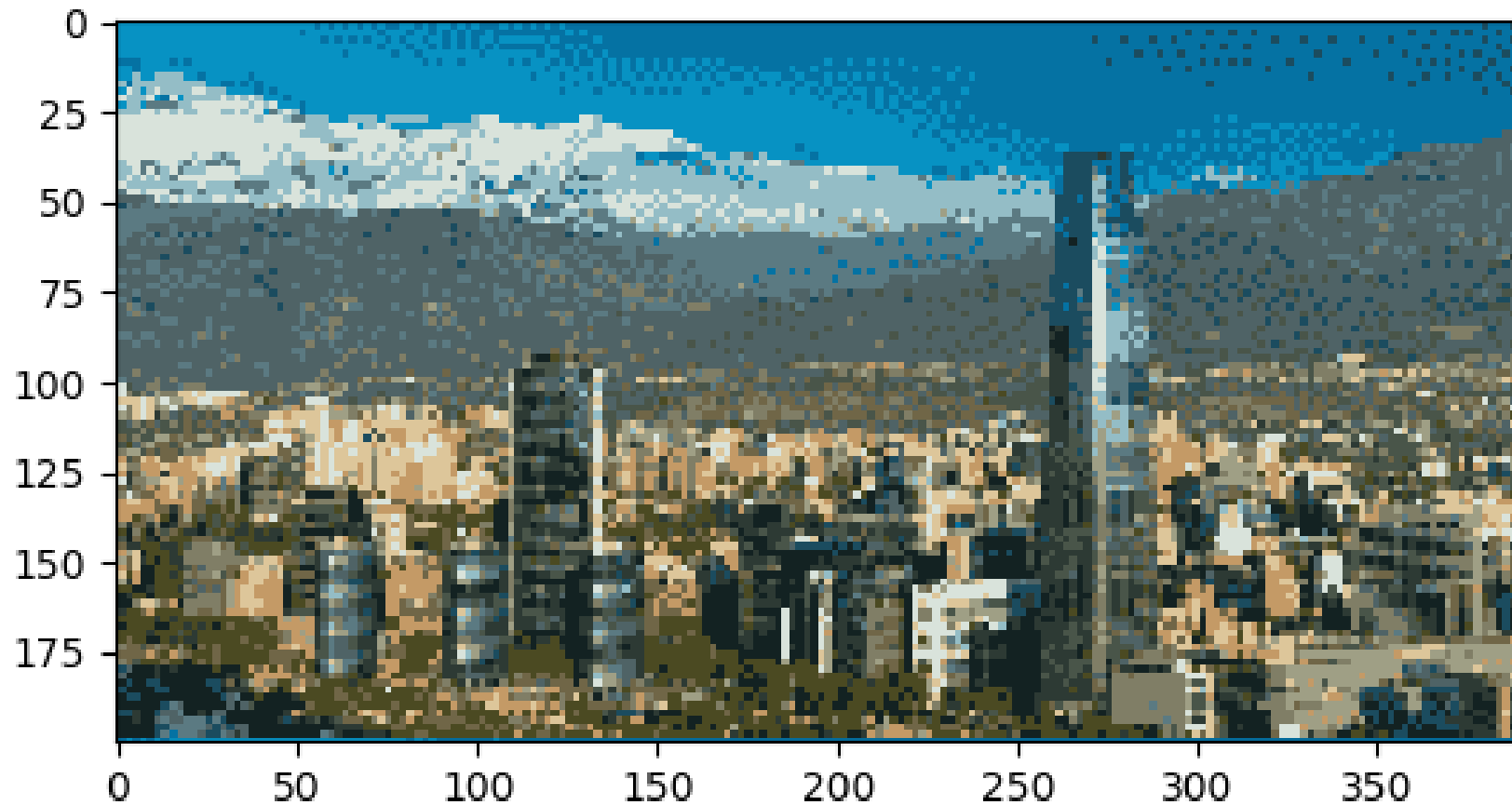


Problema de hoy – P3

- 3. Aumente el tamaño de la imagen al doble. Deje los valores de los pixeles originales cada dos pixeles. Calcule manualmente píxel por medio como el promedio de los pixeles vecinos conocidos. (MEDIO)

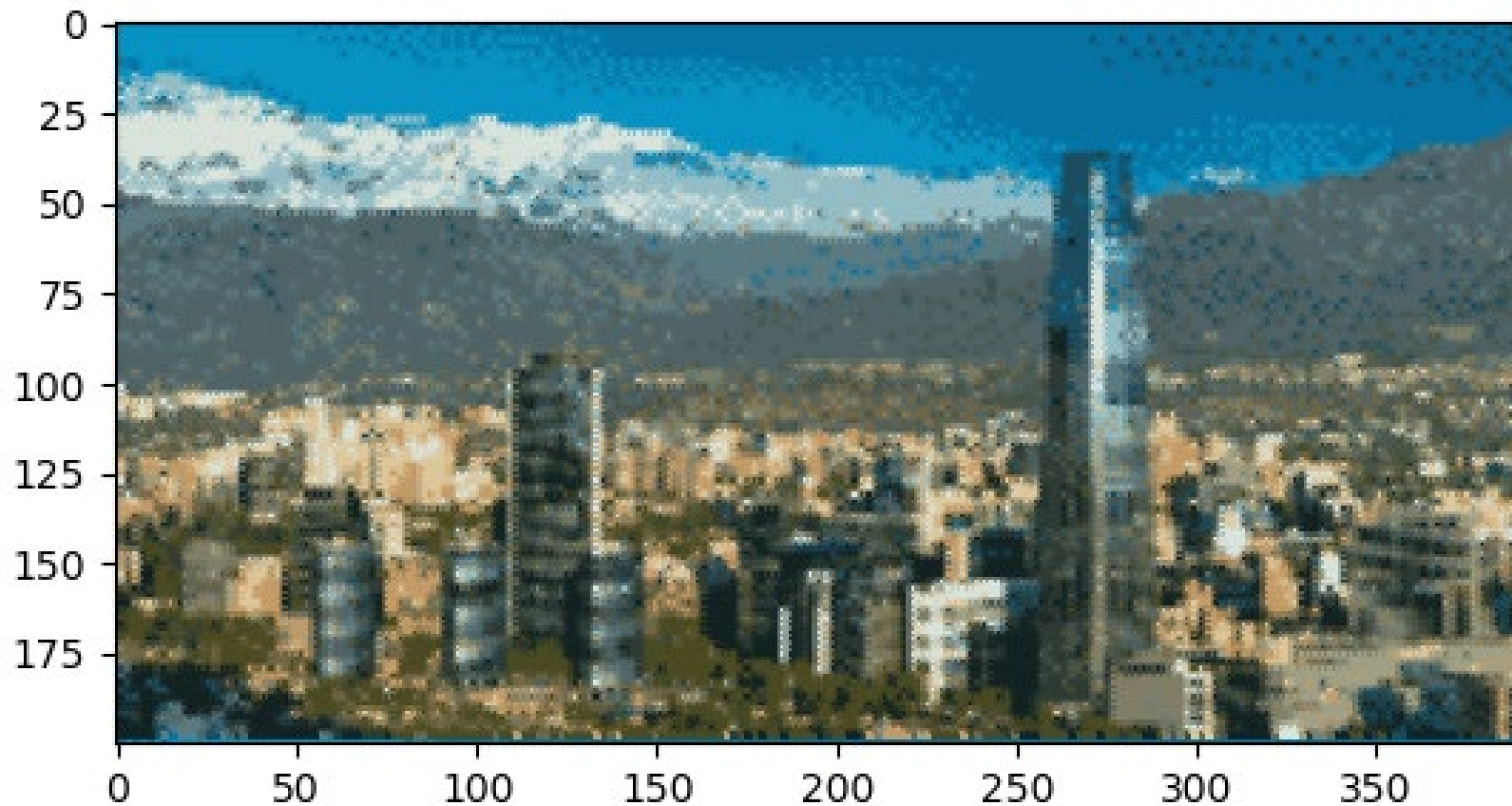
Problema de hoy – P3

- 3. Normal, sin promediar:



Problema de hoy – P3

- 3. Promediado:



Actividad sugerida

- Por una décima de bonificación, realizar los puntos 4 al 6 de la actividad y enviar el código.
- Defina una función que dado un parámetro t interpole entre dos colores $c1$ y $c2$ calculando el color final como: $c1 * (1 - t) + c2 * t$. Basándose en el archivo `ex_color_palette_anim.py`, utilice la función anterior para generar una animación donde la imagen cambie entre dos paletas de colores distintas. Utilizando la estrategia anterior genere una animación que convierta la imagen en escala de grises, tono sepia, visión nocturna (negro a verde claro) y visión térmica (entre rojo, amarillo y azul).

Muchas gracias por su atención

¿Preguntas?