# Making Gray-Scale Image from Color Image

This code provides functions to read a raw image file, convert it to an RGB image, and further transform it into a grayscale image. Finally, it displays the grayscale image using the matplotlib library. **I’ll break it down step by step:**



## Here, two libraries are imported:

numpy: It's a fundamental package for scientific computing in Python and provides support for arrays (including matrices) and many mathematical functions to operate on these arrays.

matplotlib.pyplot: A plotting library used for creating visualizations in Python.

## Function: raw\_to\_rgb(filename, width, height)

This function takes a raw image file and its dimensions as input and returns its corresponding RGB image.

open(filename, 'rb') : Opens the file in binary read mode.

np.frombuffer(f.read(), dtype=np.uint8) : Reads the binary data from the file and converts it into a 1D numpy array of 8-bit unsigned integers (0 to 255).

np.reshape(data, (height, width, 3)) : Reshapes the 1D array into a 3D array (an RGB image) of the specified height and width. The 3 in the dimensions indicates that there are 3 channels (Red, Green, Blue).

## Function: rgb\_to\_grayscale(img)

This function takes an RGB image as input and returns its grayscale version.

R = img[:,:,0], G = img[:,:,1], and B = img[:,:,2]: These lines split the RGB image into its three channels: Red, Green, and Blue, respectively.

Y = 0.299 \* R + 0.587 \* G + 0.114 \* B: This formula calculates the luminance (or brightness in an image) and is a commonly used method to convert RGB images to grayscale. The coefficients represent the different weights of each color channel in human perception.

Y.astype(np.uint8): Ensures the resulting grayscale values remain as 8-bit unsigned integers.

## Image Processing and Displaying

The dimensions of the raw image file are set to 256x256. You'd replace these values if you have a different image size.

raw\_to\_rgb(...): The raw image is read and converted to RGB format.

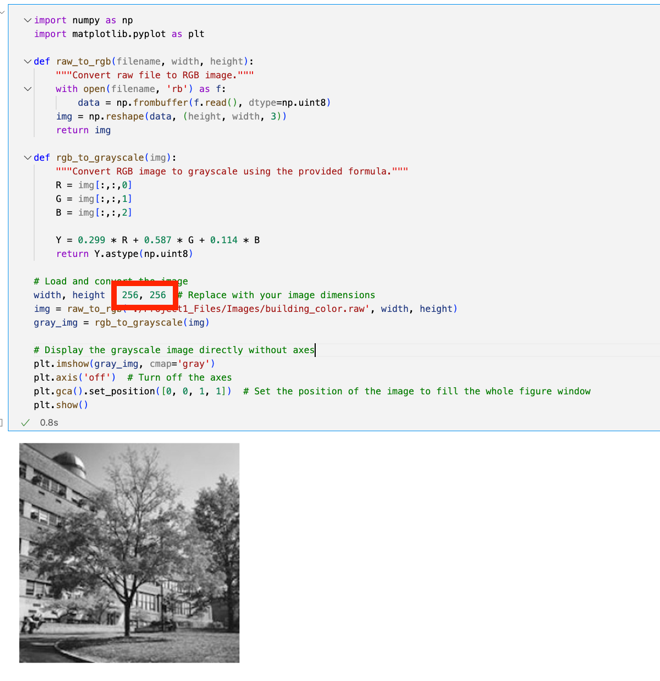
rgb\_to\_grayscale(...): The RGB image is then converted to grayscale.

plt.imshow(gray\_img, cmap='gray'): Uses matplotlib to display the grayscale image. The argument cmap='gray' ensures the image is displayed in grayscale color mapping.

plt.axis('off'): Removes the axis from the image display.

plt.gca().set\_position([0, 0, 1, 1]): Ensures the image fills the whole figure window.

plt.show(): Displays the processed grayscale image.



The first example “building\_color.raw” file:  
  
The above two are 256x256 8-bit depth, interleaved RGB images.