**Student:** Le Minh Bang 黎明朋

**ID:** 313540015

**Video Compression Homework 1**

**Color Space Conversion**

1. **Overview**

In this assignment, we explore the fascinating world of color transformations using the "lena.png" image. Specifically, we’ll look at how this image changes when we convert it between the RGB, YUV, and YCbCr color spaces. What’s exciting is that we did all of this in Python without relying on any pre-built libraries for color conversions. The result? Eight grayscale images, each representing a different color channel: R, G, B, Y, U, V, Cb, and Cr.

This task isn't just about creating pretty pictures; it’s about deepening our understanding of how different color spaces represent image data.

Color spaces play a crucial role in digital imagery, affecting how we store, process, and display visuals. While RGB is the go-to model for screens, it’s not always the best choice for tasks like compression or transmission. This is where models like YUV and YCbCr come in handy—they separate brightness from color data, making compression more efficient. In this report, we’ll take you through the manual steps of converting the image from RGB to YUV and YCbCr, along with showing the grayscale representations of each channel.

1. **Insight into Color Spaces**
   1. **RGB**

* **Definition**: This model uses Red, Green, and Blue to represent colors.
* **Range**: Each color is between 0 and 255 in 8-bit images.
* **Use Case**: Commonly employed for display purposes, though it could be more suitable for applications needing the separation of brightness and color.
  1. **YUV**
* **Components**:
  + **Y (Luminance)**: Describes brightness levels.
  + **U and V (Chrominance)**: Convey color information.
* **Advantages**: This model enhances compression by decoupling brightness from color, as the human eye is more sensitive to brightness than color differences.
  1. **YCbCr**
* **Components**:
  + **Y (Luminance)**: Equivalent to Y in the YUV model.
  + **Cb and Cr (Chrominance)**: Represent the color difference components for blue and red, respectively.
* **Application**: Extensively utilized in video compression formats like JPEG and MPEG.
  1. **RGB to YUV and YCbCr Transformations**

The conversions between RGB, YUV, and YCbCr were conducted using specific mathematical formulas tailored for each color model.

Convert RGB to YUV:

Convert RGB to YCbCr:

1. **Python Implementation**
   1. **Libraries**

* **NumPy**: Employed for numerical operations.
* **OpenCV (cv2)**: Utilized for handling image processing tasks, including reading and writing images.
  1. **Steps**

1. **Loading the Image**: The image "lena.png" was loaded using OpenCV’s imread function, converting it into a NumPy array for further processing.
2. **Channel Separation**: The red (R), green (G), and blue (B) channels were extracted from the image array for individual processing.
3. **Conversion Formulas**: The conversion from RGB to YUV and YCbCr color spaces was performed using predefined mathematical expressions applied to the respective channels.
4. **Normalization and Clipping**: The U and V values were adjusted by adding 128, and all channel values were clipped within the range of 0–255.
5. **Data Conversion**: The channel arrays were converted into 8-bit unsigned integers (uint8) to ensure proper representation of the image data.
6. **Saving the Images**: Grayscale images were saved for each channel using OpenCV’s imwrite function, with files stored in the specified images directory.
7. **Output**

A total of eight grayscale images were produced, each representing one of the RGB, YUV, and YCbCr components:



**Figure 1:** R.png - Red channel



**Figure 2:** G.png - Green channel.



**Figure 3**: B.png - Blue channel.



**Figure 4**: Y.png - Luminance component.



**Figure 5**: U.png - Chrominance U from YUV.



**Figure 6**: V.png - Chrominance V from YUV.



**Figure 7**: Cb.png - Chrominance Cb from YCbCr.



**Figure 8:** Cr.png - Chrominance Cr from YCbCr.

1. **Analysis**
   1. **Luminance (Y) Channel:**

The Y channel resembles a standard grayscale image, encapsulating brightness, which plays a significant role in human vision.

* 1. **Chrominance Channels**
* **U and V (YUV)**: These channels represent the color variations, showing color differences with reduced intensity since humans are less sensitive to changes in chrominance.
* **Cb and Cr (YCbCr)**: These channels operate similarly to U and V, though they use different coefficients. They are vital in compression, where chrominance can be reduced.
  1. **RGB Channels**

Each channel visualizes the contribution of red, green, and blue to the overall image, showcasing how the primary colors form the final image.

1. **Conclusion**

This exercise of manually transforming "lena.png" into different color spaces and generating individual grayscale images provided valuable insights into image representation. Separating luminance from chrominance is particularly beneficial for tasks like compression, where the bandwidth can be optimized by leveraging human visual system characteristics. Understanding these color spaces is crucial for anyone working in digital imaging and related areas, as the choice of color space significantly impacts the efficiency and quality of image processing operations.