

# Proposal: Identifying Makeup Color from Images

**Student: Constanza F. Schibber**

**tl;dr:**

- **Goal:** Identify the color of makeup on the CIELab color space in order to compare shades on a standardized scale rather than by creative names each brand decides.
- **Data:** Table of makeup products along with an image of each product collected from makeup retailers by scraping their website or using their API.
- **Method:** (1) color segmentation (e.g., k-means) (2) Multimodal Large Language Model (e.g. Chat GPT-4)
- **Tech stack:** Python, Google Colab, Streamlit

What is the problem you want to solve? Why is it an interesting problem?

I aim to identify the color of makeup from images to organize makeup by shade. Makeup products often have fanciful and inconsistent names, making it difficult for consumers to find the desired shade.

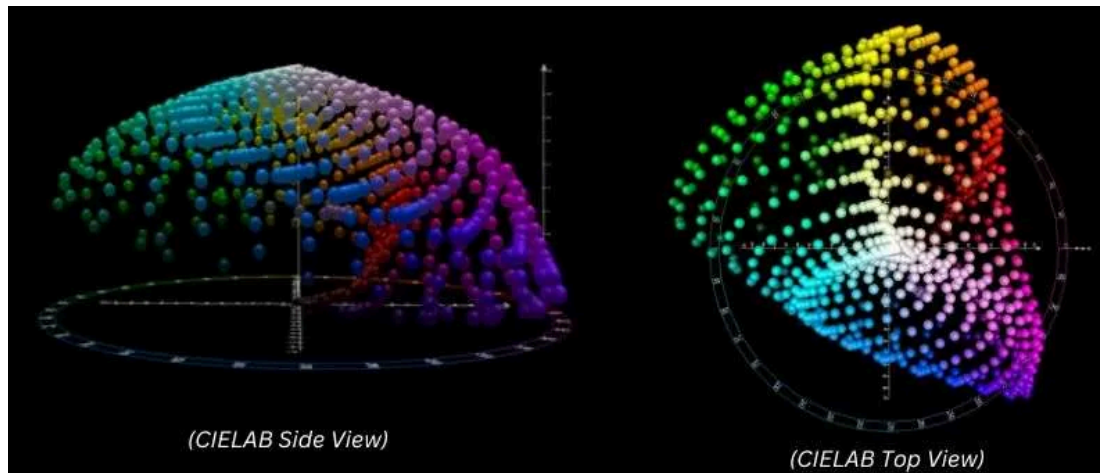
Additionally, these names can complicate organizing makeup by color. For example, the following makeup shades are all labeled as "mauve," even though they are clearly different shades. Some are not even mauve (as defined by Pantone) but brown.



By leveraging the [CIELAB color space](#), which provides a standardized and perceptually uniform representation of color, I aim to identify makeup colors from images, and organize them and categorize them by shade.

CIELAB is a color space defined by the International Commission on Illumination (CIE) where colors are represented in three dimensions:

- L (lightness)
- a (green to red)
- b (blue to yellow)



Source: Wikipedia

CIELAB is designed to be perceptually uniform, meaning that the numerical differences between colors align with the perceived differences by the human eye. This makes it a more accurate representation of color perception compared to models like RGB, which do not account for how we naturally perceive color differences, rendering them less reliable for tasks that require precise color matching.

Moreover, CIELAB offers a standardized method for describing colors, enabling accurate color comparison and matching across various brands and products. This standardization eliminates the confusion caused by subjective or creative color names, making it easier for consumers to find their desired shades. This is especially important for consumers seeking a specific color or looking for a similar shade from a more affordable brand.

In sum, this approach would enable the creation of a more reliable and user-friendly system for consumers to search and compare makeup by shade.

What **data** are you going to use to solve this problem?

I have a table of +2000 makeup products from different brands. The data was collected from makeup sellers like Ulta, Sephora, etc. either by web scraping or the seller API. The features I am interested in are:

- **Product:** The description of the product like lipstick, blush, eye shadow. I need to simplify the product description into clear categories because at the moment, it is the description provided by the brand. For instance, “lasting color lip gloss stain” should be “lip gloss”.

- **Brand:** The brand of the product. There are around 110 unique brands.
- **Link to image:** The link to the make up image (like the ones above). I need to download the images with Python and save them in a folder.
- **ID:** I will create a unique ID for each row in the table that will also link to the images of the product.

The exploratory data analysis won't be extensive because I am working with images.

## Outline your approach to solving this problem.

I will follow two approaches to identify the color:

- **Approach 1:** Color-based segmentation. I will use a clustering algorithm to identify the colors of clusters on the image.

Example:

- **Approach 2:** I will use a Multimodal model like Chat GPT-4 or Claude to identify the color of the make-up. I did a small test with ChatGPT-4 and Claude, and was able to identify the color of the makeup correctly in 2 images. Although using a Multimodal model is more expensive than approach 1, in practice it could be used for cases in which approach 1 does not work well.

Example:  [Multimodal LLM] CIELAB makeup with Claude

I believe the biggest challenge will be to identify the color of makeup when the package is included in the picture.

For each approach, I will split the data into training, validation, and test sets for model evaluation. In the case of 'approach 2', the training portion will include considering different prompts and then evaluating the prompt selected.

I will also compare these approaches to each other:

- Did they retrieve the same color? Coded 0 and 1.

- How different are the colors they retrieved? Distance between both colors within the CIELab color space. The measure is called Delta E:  
<https://zschuessler.github.io/DeltaE/learn/>

I will also select a (small) random sample and manually check if the identified color is correct or not. In a real work scenario, we would have human evaluators to compare the color of the original image and the color retrieved, but because hiring human evaluators would be expensive, I will do that myself.

## What will be your final deliverable?

Simple web-user interface in which users can filter makeup by color (data will be saved in a table)

## What are the computational resources needed?

They will not be significant. I will use python, Google Colab, and Streamlit.