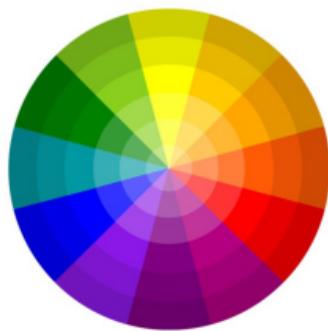


ĐẠI HỌC QUỐC GIA TP.HCM  
TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN

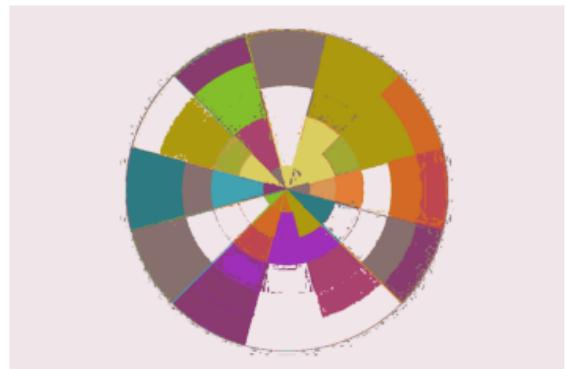


**BÀI BÁO CÁO  
COLOR COMPRESSION**

Original Image



Reduced Image



**Sinh viên thực hiện:**  
*Trịnh Đức Thiên - 20127330*



## Mục lục

I.	ĐÁNH GIÁ .....	3
II.	Ý TƯỞNG THỰC HIỆN.....	3
III.	KẾT QUẢ .....	3
IV.	NHẬN XÉT KẾT QUẢ .....	4
V.	TÀI LIỆU THAM KHẢO.....	4

## I. ĐÁNH GIÁ

NỘI DUNG	MỨC ĐỘ HOÀN THÀNH
1. Sử dụng Numpy, PIL, matplotlib để thực hiện yêu cầu	100%
2. Chương trình cho kết quả đúng yêu cầu	100%
3. Báo cáo đầy đủ nội dung	100%

## II. Ý TƯỞNG THỰC HIỆN

### 1. Ý tưởng

- K-means là thuật toán phân cụm pixel vào các cụm màu khác nhau và thay thế màu của mỗi pixel bằng màu của centroid tương ứng để giảm số lượng màu của một ảnh

### 2. Mô tả hàm

```
def kmeans(img_1d, k_clusters, max_iter, init_centroids='random'):
```

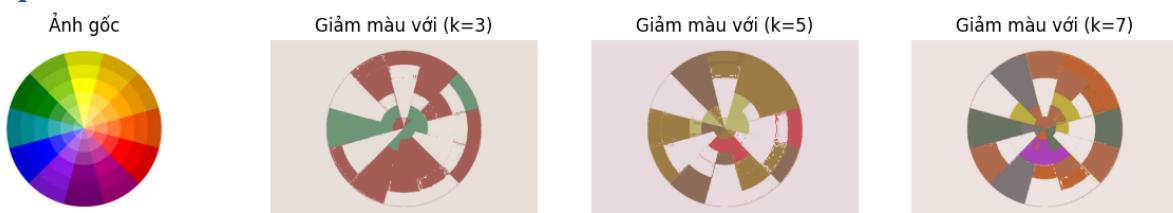
#### a. Input:

- img\_1d: Mảng numpy 1D chứa tất cả các pixel của ảnh.
- k\_clusters: Số lượng cụm mà chúng ta muốn tạo.
- max\_iter: Số lượng lặp tối đa cho thuật toán.
- init\_centroids: Phương pháp khởi tạo centroids, có hai lựa chọn: 'random' hoặc 'in\_pixels'.

#### b. Output:

- centroids: Mảng numpy chứa các centroid màu.
- labels: Mảng numpy chứa nhãn cho từng pixel (chỉ số của cụm mà pixel thuộc về).

## III. Kết quả



#### IV. NHẬN XÉT KẾT QUẢ

- Khi giảm số lượng màu của ảnh, mức độ chi tiết và sắc nét của ảnh giảm đi. Các vùng màu tương tự được gom lại thành các cụm màu đại diện, dẫn đến sự mất mát thông tin màu sắc.
- Giá trị  $k$  ảnh hưởng đến mức độ giảm số lượng màu và chất lượng kết quả. Với  $k$  càng lớn, số lượng màu càng nhiều và ảnh giữ được nhiều chi tiết màu sắc hơn. Tuy nhiên, khi  $k$  quá lớn, có thể xảy ra hiện tượng over-segmentation, khiến ảnh trở nên hỗn loạn và không tự nhiên.

#### V. Tài liệu tham khảo

- [1] [K-means Clustering](#)
- [2] [K-means Clustering: Simple Applications](#)