

BÁO CÁO PROJECT 1

1. Thông tin cá nhân

- Họ và tên: Phan Trí Tài
- MSSV: 20127318

2. Ý tưởng thực hiện, mô tả các hàm

a) Ý tưởng

- Tìm hiểu về thuật toán K-Means.
- Tìm hiểu các thư viện NumPy (tính toán ma trận), PIL (đọc, ghi ảnh), matplotlib (hiển thị ảnh).
- Tìm nguồn trên mạng hoặc trong sách.
- Nếu còn thắc mắc thì nên hỏi GVBM để hiểu rõ hơn về đề án.

b) Mô tả hàm

- Hàm đọc ảnh và hiển thị ảnh ra màn hình

```
## Hàm đọc ảnh và in ảnh ra ngoài màn hình
img_before = Image.open('view.png')
plt.imshow(img_before)
img = np.array(img_before)
img_height, img_width = img.shape[0], img.shape[1]
img = img.reshape( img_height*img_width, img.shape[2])
```

- Hàm K-Means

- Điều kiện if else đầu tiên dùng để init trên ảnh hoặc random.
- Vòng lặp for dùng để in ra color centroids.

```
def kmeans(img_1d, k_clusters, max_iter, init_centroids='random'):  
  
    if init_centroids == 'in_pixels':  
        centroids = img_1d[np.random.choice(img_1d.shape[0] , size = k_clusters , replace = False)]  
  
    elif init_centroids == 'random':  
        centroids = np.random.randint(0,255,size=(k_clusters,img_1d.shape[1]))  
  
    for i in range(max_iter):  
        # Gan cho centroids  
        d = np.linalg.norm(img_1d - centroids[:, np.newaxis] , axis = 2)  
        labels = np.argmin(d , axis = 0)  
  
        # Cap nhat mean  
        means = []  
        for k in range(k_clusters):  
            means.append(img_1d[labels == k].mean(axis = 0))  
        means = np.array(means)  
        for i in range(k_clusters):  
            if len(means[i]) != 0:  
                centroids[i] = means[i]  
  
        d = np.linalg.norm(img_1d - centroids[:, np.newaxis] , axis = 2)  
        labels = np.argmin(d , axis = 0)  
    return centroids, labels
```

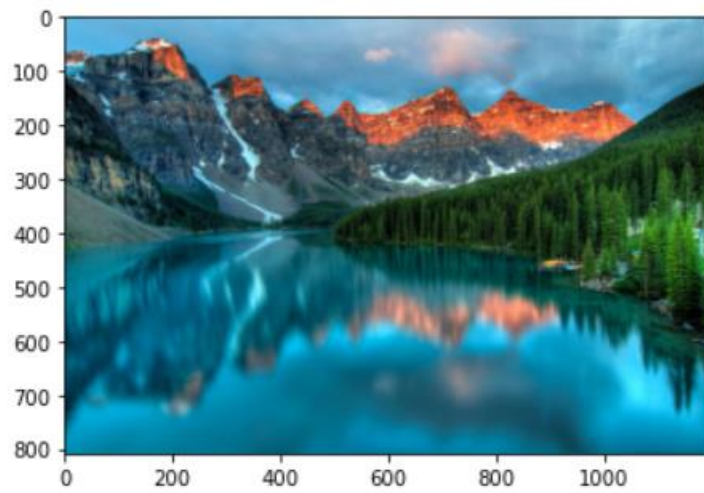
- Hàm test case

- Cho các k = 3,5,7 thì cũng như thế chỉ khác ở dòng `k_cluster = 3`

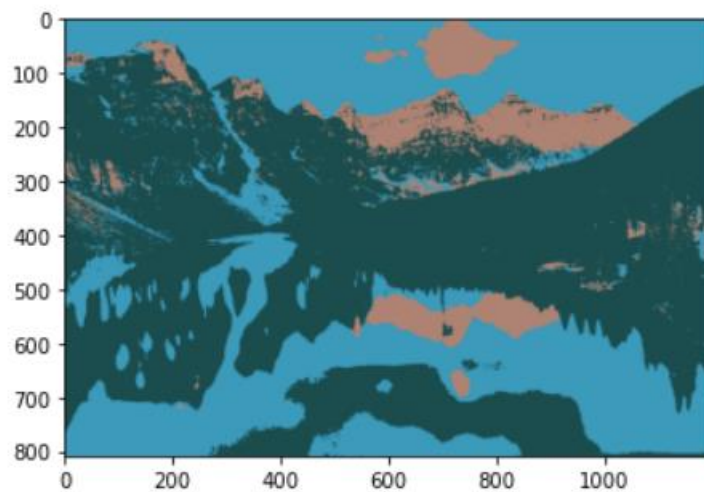
```
k_cluster = 3  
img_after = img.copy()  
centroids, labels = kmeans(img_after,k_cluster,10,'random')  
  
## Gan Label Lại cho ảnh  
for k in range(centroids.shape[0]):  
    img_after[labels == k] = centroids[k]  
  
img_after = img_after.astype("uint8")  
img_after = img_after.reshape(img_height,img_width,3)  
plt.imshow(img_after)
```

3. Hình ảnh kết quả

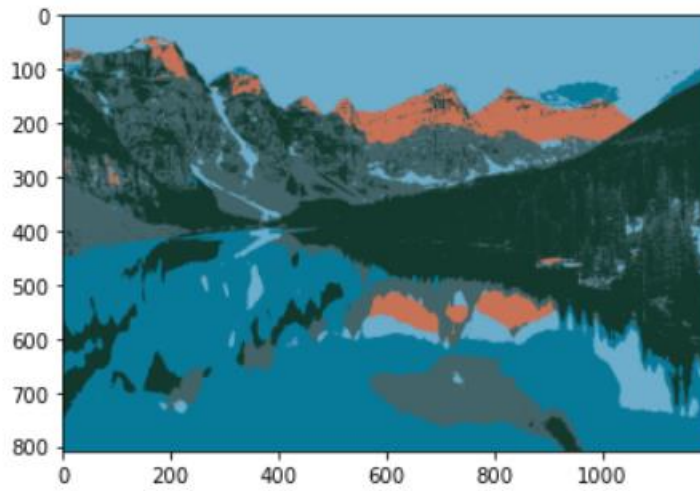
- Ảnh gốc



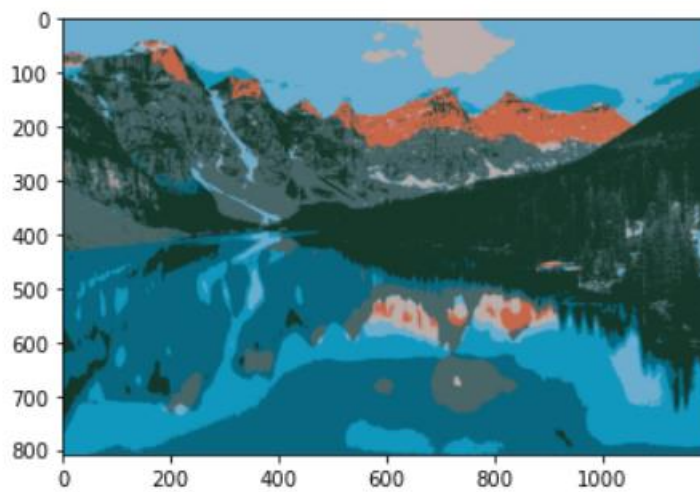
- $K = 3$



- $K = 5$



- $K = 7$



4. Nhận xét

- Bài làm chỉ hoàn thành được 50-60% vì còn nhiều thiếu sót và vẫn chưa hoàn toàn hiểu được hết thuật toán K-Means.

5. Tài liệu tham khảo

- <https://nguyenvanhieu.vn/thuat-toan-phan-cum-k-means/#gioi-thieu-ve-k-means>
- <https://github.com/rdsquare/Image-Compression-SVD-Python/blob/master/compression.ipynb/>
- <https://towardsdatascience.com/dimensionality-reduction-of-a-color-photo-splitting-into-rgb-channels-using-pca-algorithm-in-python-ba01580a1118>
- <https://www.geeksforgeeks.org/image-compression-using-k-means-clustering/>
- <https://www.youtube.com/watch?v=H7qMMudo3e8>