

BÀI TẬP: HỌC KHÔNG GIÁM SÁT

1. Lập trình lại thuật toán K-means, so sánh tốc độ và kết quả với thuật toán K-means trong thư viện Scikit-learn
 - a. Lưu đồ giải thuật của thuật toán K-means như sau:

```
% Initialize centroids
centroids = kMeansInitCentroids(X, K);
for iter = 1:iterations
    % Cluster assignment step: Assign each data point to the
    % closest centroid. idx(i) corresponds to c^(i), the index
    % of the centroid assigned to example i
    idx = findClosestCentroids(X, centroids);

    % Move centroid step: Compute means based on centroid
    % assignments
    centroids = computeMeans(X, idx, K);
end
```

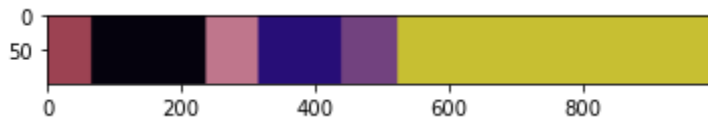
Sử dụng lưu đồ giải thuật này để tự lập trình lại thuật toán K-means, dùng các thư viện Python chuẩn và Numpy, Matplotlib để vẽ minh họa dữ liệu

- b. Tự sinh ra dữ liệu để kiểm thử tốc độ và tọa độ các tâm tính được của thuật toán lập trình được với thư viện chuẩn của Scikit-learn. Thử nghiệm với các bộ dữ liệu từ nhỏ (vài nghìn điểm) tới lớn (>1 triệu điểm, có thể tự sinh ra dữ liệu), vẽ minh họa kết quả giữa hai phương pháp
 - c.
2. Từ kết quả của bài 1, với mỗi điểm đưa vào (không nằm trong dữ liệu), hãy tìm tâm gần nhất.
 3. Từ kết quả của bài 1 và 2, xây dựng hệ thống phát hiện khuôn mặt giống nhau:
 - a. Thu thập ít nhất 100 ảnh khuôn mặt
 - b. Với mỗi ảnh, xây dựng histogram của ảnh tạo thành vector đặc trưng (1xn chiều)
 - c. Sử dụng thuật toán phân cụm, phân thành n nhóm khác nhau, vẽ minh họa kết quả
 - d. Với một ảnh khuôn mặt bất kỳ, tìm ra ảnh này sẽ thuộc cụm nào, vẽ lên hình ảnh đại diện của cụm (điểm gần nhất so với tâm cụm) và ảnh đầu vào để so sánh kết quả
 - e. Đánh giá chất lượng của thuật toán
 4. Tìm màu đại diện cho hình ảnh, vẽ minh họa:

Cho một hình ảnh bất kỳ, đọc ảnh và lưu tất cả các giá trị pixel màu thành một mảng các điểm (n điểm x 3 màu), ví dụ một hình ảnh đầu vào

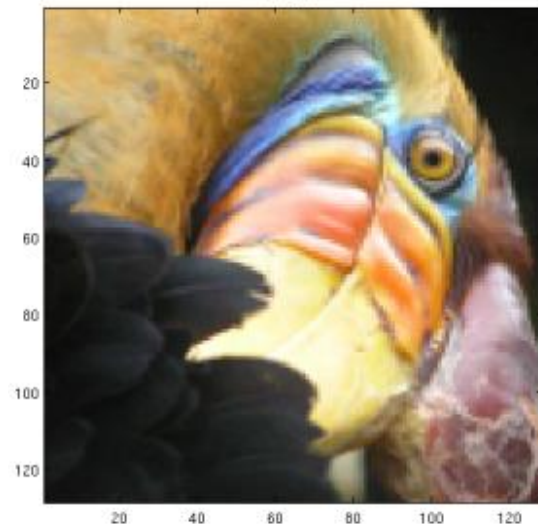


- a. Phân cụm để tìm ra 6 màu chính xuất hiện trên ảnh
- b. Xác định tần suất xuất hiện của các điểm trọng cụm, vẽ minh họa.
Ví dụ: với ảnh đầu vào như trên, các màu cơ bản sẽ như hình sau, trong đó độ rộng của màu thể hiện tần suất xuất hiện của từng màu (tỉ lệ số điểm thuộc cùng 1 cụm/ tổng số điểm)



- c. Thay đổi số lượng màu cần hiển thị, đánh giá kết quả
 - d. Resize ảnh, đánh giá kết quả và tốc độ chạy chương trình
 - e. Mở rộng bài toán, tìm n màu cơ bản trong 1 đoạn video clip, vẽ minh họa
5. Nén ảnh:
- Thông thường ảnh màu được thể hiện bởi 24 bit màu, mỗi pixel được thể hiện bởi 3 giá trị 8 bit số nguyên không dấu (từ 0-255) thể hiện kênh màu Red, Green, Blue. Tổng số thể hiện màu là $256^3 = \sim 16$ triệu màu.
- a. Sử dụng giải thuật phân cụm, tìm ra 16 màu cơ bản nhất
 - b. Biểu diễn lại ảnh sử dụng 16 màu đó
 - c. Đánh giá tốc độ, kết quả khi giảm kích thước ảnh trước khi phân cụm
Ví dụ: ảnh bên trái là ảnh gốc với đầy đủ màu, ảnh bên phải là ảnh mã hóa chỉ sử dụng 16 màu

Original



Compressed, with 16 colors.

