

ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH

ĐẠI HỌC KHOA HỌC TỰ NHIÊN

KHOA CÔNG NGHỆ THÔNG TIN



Báo cáo đồ án

Lab: Project 1

Môn học: Toán ứng dụng và thống kê

Người thực hiện:

22127390 - Nguyễn Văn Lê Bá Thành

Giảng viên:

Cô Phan Thị Phương Uyên

Thầy Nguyễn Ngọc Toàn

Thầy Nguyễn Văn Quang Huy

Thầy Vũ Quốc Hoàng

Ngày 17 tháng 6 năm 2024

Mục lục

1	Lời cảm ơn	2
2	Giới thiệu	3
3	Ý tưởng thực hiện và mô tả các hàm	4
3.1	Thuật toán K-means	4
3.1.1	Tổng quan về K-means	4
3.1.2	Sử dụng K-means trong nén màu hình ảnh	4
3.2	Mô tả các hàm trong chương trình	5
3.2.1	Các thư viện cần dùng	5
3.2.2	Các hàm trong chương trình	5

1 Lời cảm ơn

Nếu không có sự giúp đỡ và hướng dẫn của nhiều người và từ nhiều nguồn khác nhau, đồ án này sẽ không thể hoàn thành.

Đầu tiên, em muốn cảm ơn các nhà phát triển và cộng tác viên của thư viện Numpy, Pillow và Matplotlib. Tài liệu có sẵn trên trang web của họ rất hữu ích trong việc làm quen và áp dụng để cài đặt các hàm cần thiết cho đồ án này.

Em cũng muốn cảm ơn các thầy, cô hướng dẫn vì những lời khuyên hữu ích và những bài giảng với các kiến thức cần thiết trong suốt môn học này. Sự hiểu biết của em về dự án này đã được nâng cao hơn nhiều nhờ những lời giải thích rõ ràng và sự cởi mở với các câu hỏi.

Em đặc biệt muốn cảm ơn Gemini một mô hình ngôn ngữ lớn, vì đã giúp thu thập thông tin và tóm tắt thông tin về việc thực hiện color compression trên hình ảnh và cách hoạt động của thuật toán K-means.

Em muốn cảm ơn Grammarly vì đã giúp đảm bảo rằng báo cáo được trau chuốt và chuyên nghiệp bằng cách hỗ trợ kiểm tra ngữ pháp và chính tả trong phần báo cáo này.

2 Giới thiệu

Trong xã hội hiện tại hình ảnh kỹ thuật số rất cần thiết trong cuộc sống, từ việc lưu trữ kỷ niệm, nghiên cứu, nghệ thuật. Tuy nhiên, các tệp hình ảnh có độ phân giải cao thường có kích thước lớn và điều này sẽ dẫn đến việc xử lý, truyền tải và lưu trữ kém hiệu quả về mặt thời gian và không gian lưu trữ. Đó cũng là mục tiêu của kỹ thuật nén ảnh nhằm làm giảm kích thước ảnh mà không làm giảm chất lượng hình ảnh.

Báo cáo này thảo luận về việc áp dụng phương pháp nén màu K-mean nhằm làm giảm không gian lưu trữ cần thiết cho một hình ảnh. K-mean là một phương pháp học máy không giám sát, phân cụm các điểm dữ liệu dựa trên sự tương đồng. Kỹ thuật K-mean có thể làm giảm số lượng màu hiển thị trong hình ảnh khi nén hình ảnh và phương pháp này cung cấp sự cân bằng giữa độ trung thực của hình ảnh và tỷ lệ nén.

Các phần sau được tổ chức như sau, Phần 3 với các ý tưởng và mô tả về thuật toán K-mean và các hàm tiện ích. Phần 3 đưa ra kết quả với số lượng màu sắc cụ thể và nhận xét về các kết quả này. Các kết luận sẽ được đưa ra ở Phần 4.

3 Ý tưởng thực hiện và mô tả các hàm

3.1 Thuật toán K-means

3.1.1 Tổng quan về K-means

- **K-means là gì?**

K-means là thuật toán không giám sát nổi tiếng dùng để phân vùng bộ tập dữ liệu thành k phân cụm cho trước. Mục tiêu của thuật toán này bao gồm: [1]

1. Đưa các điểm dữ liệu trong bộ dữ liệu vào các phân cụm
2. Các điểm dữ liệu trong cùng một phân cụm càng giống nhau càng tốt
3. Các điểm phân cụm khác nhau càng khác nhau càng tốt

Trong đề án này, chúng ta sẽ sử dụng thuật toán K-means để cài đặt một chương trình có khả năng nén màu của một hình ảnh thành k màu cho trước.

- **Điều kiện dừng và sự hội tụ** Các tâm cụm hội tụ khi các giá trị tâm cụm mới không thay đổi sau các lần lặp, do đó điều kiện dừng của thuật toán K-means diễn ra dưới các trường hợp sau đây: [1]

1. Các tâm cụm mới không thay đổi
2. Các điểm dữ liệu nằm trong cùng một phân cụm
3. Đạt đến số lần lặp tối đa

3.1.2 Sử dụng K-means trong nén màu hình ảnh

- **Cấu tạo của ảnh:**

Hình ảnh kỹ thuật số hiện nay được cấu tạo từ các điểm ảnh (pixel) có kích thước height x width với mỗi điểm ảnh có thể được biểu diễn dưới nhiều dạng khác nhau như HEX, RGB, ... [2]

Trong đề án này chúng ta sẽ sử dụng hệ màu RGB, với ba thuộc tính Red, Green, Blue là các số nguyên dương 8 bit bắt đầu từ 0 đến 255. Kết hợp ba thuộc tính ở trên ta có thể tạo ra một điểm ảnh có màu nằm trong $256 * 256 * 256 \approx 1.7 \times 10^7$ màu. [2]

Dựa vào các ý trên ta có thể phân tích một hình ảnh thành các thành phần nhỏ như sau:

1. Một điểm ảnh là một điểm dữ liệu
2. Một điểm ảnh chứa 3 thuộc tính (properties) bao gồm 3 channels Red, Green, Blue
3. Chiều dài (height) và độ rộng (width) của hình ảnh

Vậy ta có thể xác định rằng một hình ảnh là một ma trận 3 chiều height x width x channels

- **Áp dụng K-means vào thuật toán nén ảnh**

Ý tưởng của thuật toán nén được cài đặt trong đề án này bao gồm:

1. Làm phẳng ma trận điểm ảnh 3 chiều ban đầu height, width, channels trở thành ma trận 2 chiều height x width, channels.
2. Khởi tạo giá trị tâm cụm (centroids) ban đầu bằng cách chọn ngẫu nhiên hoặc chọn từ các giá trị RGB của ảnh và khởi tạo một mảng labels nguyên có kích thước height x width dùng để chỉ định pixel nào ứng với tâm cụm nào.
3. Tính toán khoảng cách giữa các điểm ảnh và các phân cụm sau đó đưa điểm ảnh vào phân cụm có khoảng cách ngắn nhất so với điểm ảnh đó bằng cách gán nhãn.
4. Tính toán lại giá trị tâm cụm (centroids) bằng cách tính trung bình các giá trị được phân loại vào phân cụm tương ứng.
5. Sử dụng các giá trị tâm cụm mới này và lặp lại bước 3 cho đến khi đạt đến số vòng lặp tối đa hoặc giá trị tâm cụm mới giống với giá trị tâm cụm cũ (hội tụ).
6. Tạo ra một ma trận 2 chiều height x width, channels bằng cách gán giá trị tâm cụm ứng với giá trị trong mảng labels và do labels có kích thước height x width, do đó ma trận 2 chiều được tạo mới sẽ có height và width giống với ảnh gốc và chứa các giá trị tâm cụm.
7. Reshape ma trận 2 chiều height x width, channels này thành ma trận 3 chiều height, width, channels.

Kết quả trả về là một hình ảnh có kích thước giống kích thước ảnh ban đầu với k màu được nén lại

3.2 Mô tả các hàm trong chương trình

3.2.1 Các thư viện cần dùng

- Các thư viện bắt buộc sử dụng bao gồm:
 1. Numpy: dùng để tính toán ma trận
 2. PIL: dùng để đọc, ghi ảnh
 3. matplotlib: dùng để hiển thị ảnh
- Các thư viện được sử dụng để test và viết báo cáo
 1. Time: dùng để tính toán thời gian thực thi
 2. BytesIO: dùng để tính kích thước của file hình ảnh

3.2.2 Các hàm trong chương trình

1. `def read-img(img-path):`

Hàm