

**ĐẠI HỌC QUỐC GIA THÀNH PHỐ HỒ CHÍ MINH**  
**TRƯỜNG ĐẠI HỌC KHOA HỌC TỰ NHIÊN**  
**KHOA CÔNG NGHỆ THÔNG TIN**



**BÁO CÁO MÔN HỌC**  
**LẬP TRÌNH SONG SONG GPU**  
**ỨNG DỤNG**

| ĐỀ TÀI |

**JPEG COMPRESSION ALGORITHM**

| GIÁO VIÊN HƯỚNG DẪN |

**Thầy Phạm Trọng Nghĩa**

**Thầy Nguyễn Trần Duy Minh**

**Thầy Bùi Duy Đăng**

Thành phố Hồ Chí Minh

## I. THÀNH VIÊN

### Nhóm 9

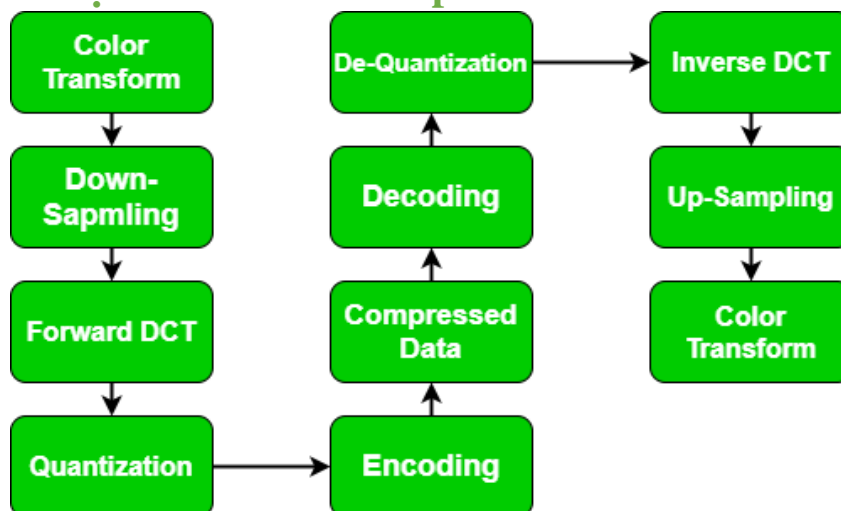
MSSV	HỌ TÊN
19127003	Nguyễn Hữu Đạt
19127472	Nguyễn Bá Minh
19127481	Trần Hoàng Nam
19127595	Nguyễn Minh Trí

## II. ĐỒ ÁN

### 1. Giới thiệu

- File JPEG (hay File Joint Photographic Experts Group) là một loại định dạng hình ảnh được lưu bằng phương pháp lossy compression (nén có mất mát dữ liệu). Trong đó, hình ảnh đầu ra là kết quả của quá trình nén, là sự cân bằng giữa kích thước lưu trữ và chất lượng hình ảnh. Hơn 70% website trên thế giới sử dụng ảnh có định dạng JPEG để lưu trữ
- Smartphone và camera sử dụng JPEG để lưu trữ
- Thuật toán nén video h.264 được sử dụng trên youtube sử dụng kỹ thuật nén tương tự JPEG
- Các thuật toán nén như JPEG giúp giảm thiểu chi phí lưu trữ, cũng như giảm thời gian khi gửi qua internet

### 2. Thuật toán JPEG Compression



Hình minh họa các bước để nén 1 file ảnh bình thường thành file có định dạng JPEG

- Thực tế pipeline gồm 6 bước, các bước còn lại là đảo ngược của các bước trước đó, các bước chính bao gồm:
  - Color transform: chuyển ảnh sang không gian màu YCbCr, ở đây, ảnh được sử dụng có kênh màu RGB, vậy nên chúng ta sẽ chuyển ảnh sang kênh màu YCbCr theo công thức:

$$\begin{aligned}
 Y &= 0.299 R + 0.587 G + 0.114 B \\
 Cb &= -0.1687 R - 0.3313 G + 0.5 B + 128 \\
 Cr &= 0.5 R - 0.4187 G - 0.0813 B + 128
 \end{aligned}$$

- Downsampling:
  - Mắt người gồm 2 loại tế bào tiếp nhận ánh sáng là rods và cones
  - Rods không nhạy cảm với màu sắc (khoảng 100 triệu tế bào trong mỗi mắt)
  - Cones thì nhạy cảm với màu sắc (khoảng 6 triệu tế bào trong mỗi mắt) Như vậy, có thể kết luận rằng mắt chúng ta nhận biết được ảnh brightness và darkness tốt hơn là ảnh màu sắc



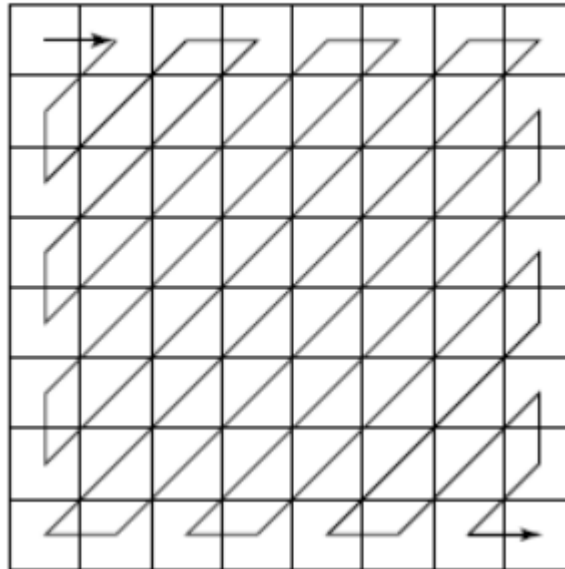
- Chúng ta sẽ gộp 4 pixel thành 1 block có kích thước 2x2, rồi tính giá trị trung bình của block đó đối với 2 kênh màu Chrominance.
- Discrete cosine transform (DCT): Thực hiện chia ma trận Y, Cb, Cr thành các block 8\*8 (đây là kích thước block tối ưu), thực hiện DCT trên các block này dựa theo công thức:

$$DCT(i, j) = \frac{1}{\sqrt{2N}} C(i) C(j) \sum_{x=0}^{N-1} \sum_{y=0}^{N-1} \text{pixel}(x, y) \cos \left[ \frac{(2x+1)i\pi}{2N} \right] \cos \left[ \frac{(2y+1)j\pi}{2N} \right]$$

$$C(x) = \frac{1}{\sqrt{2}} \text{ if } x \text{ is } 0, \text{ else } 1 \text{ if } x > 0$$

- Quantization:
  - Bước quantization trong JPEG compression có ý nghĩa là giảm độ chi tiết của dữ liệu hình ảnh bằng cách làm tròn các giá trị màu xuống theo một bước nhất định và lưu trữ chúng bằng số bit ít hơn. Quá trình này giúp giảm kích thước của dữ liệu hình ảnh, giảm dung lượng lưu trữ và tăng tốc độ truyền dữ liệu.
  - Quá trình quantization trong JPEG compression có thể được điều chỉnh thông qua bảng quantization. Bảng quantization xác định các giá trị quantization cho từng phần tử trong khối 8x8, ảnh hưởng đến mức độ nén và chất lượng của ảnh sau quá trình quantization.
  - Ở Bước này, ta thực hiện lấy từng block rồi chia cho quantization\_table (quantization table là bản được cho trước)

- Encoding: Gồm 2 bước nhỏ
  - Run length encoding: chúng ta sẽ chuyển ma trận quantized thành một array một chiều đi theo thứ tự zigzag, rồi thực hiện run length encoding



- Run length encoding (RLE) là một thuật toán nén dữ liệu đơn giản được sử dụng để giảm kích thước của dữ liệu. Nó hoạt động bằng cách đếm số lượng các ký tự liên tiếp trong một chuỗi và thay thế chúng bằng cặp giá trị: số lượng và ký tự tương ứng. Ví dụ bạn có chuỗi AAAABBBCCCCC thì sẽ được viết gọn lại thành 4A3B5C
- Huffman coding: được áp dụng cho các hệ số DCT (Discrete Cosine Transform) của các khối 8x8 trong ảnh. Các hệ số DCT được chia thành các nhóm và mã hóa bằng cách sử dụng bảng mã huffman được tạo trước. Bảng mã huffman định nghĩa các mã bit đại diện cho các nhóm hệ số DCT khác nhau.
- Sau khi hoàn thành giai đoạn huffman coding, dữ liệu được chuyển đổi thành các mã bit huffman, và các mã bit này được ghi lại vào tệp tin nén JPEG.

### III. PHÂN CÔNG

NGƯỜI THỰC HIỆN	CÔNG VIỆC	ĐÁNH GIÁ
Nguyễn Hữu Đạt	<ul style="list-style-type: none"> <li>- Color-transform</li> <li>- Downsampling</li> </ul>	Tốt
Nguyễn Bá Minh	<ul style="list-style-type: none"> <li>- DCT</li> <li>- Quantization</li> </ul>	Tốt
Trần Hoàng Nam	<ul style="list-style-type: none"> <li>- RLE + Thuyết trình + decode jpeg tuần tự</li> </ul>	Tốt
Nguyễn Minh Trí	<ul style="list-style-type: none"> <li>- Zigzag scanning + decode jpeg tuần tự</li> </ul>	Tốt
Cả nhóm	Tổng hợp và viết báo cáo	

#### Mục tiêu:

- Hiểu được cách hoạt động của thuật toán JPEG compression tuần tự : 50%
- củng cố và nâng cao kỹ năng tối ưu thuật toán trên GPU: 40%
  - o Color transform: 10%
  - o Downsampling: 10%
  - o Quantization: 10%
  - o Run length encoding: 5%
  - o Huffman encoding: 5%
- Báo cáo: 10%

### IV. NGUỒN THAM KHẢO

- Slide bài giảng
- <https://cs.stanford.edu/people/eroberts/courses/soco/projects/data-compression/lossy/jpeg/dct.htm>
- <https://www.youtube.com/watch?v=Kv1Hiv3ox8I&t=277s>