

**TRƯỜNG ĐẠI HỌC CÔNG NGHỆ THÔNG TIN
KHOA KHOA HỌC MÁY TÍNH**



UIT

BÁO CÁO CUỐI KÌ

**THE JPEG IMAGE COMPRESSION
ALGORITHM**

Giảng viên : Nguyễn Vinh Tiệp

**Sinh viên : Nguyễn Hữu Doanh - 18520606
Nguyễn Minh Châu - 18520519
Nguyễn Tấn Phúc - 18521259**

Lớp : CS114.K22.KHCL

Tháng 7/2020

MỤC LỤC

LỜI MỞ ĐẦU	3
PHẦN 1: GIỚI THIỆU	4
1. Tổng quan:	4
2. Phương pháp nén ảnh JPEG:.....	5
PHẦN 2: Ý TƯỞNG THUẬT TOÁN	6
PHẦN 3: CHI TIẾT GIẢI THUẬT	7
3.1 ENCODING:	7
• Biến đổi không gian màu:.....	7
• Down sampling:	9
• Block splitting:	10
• Discrete cosine transform:	10
• Lượng tử và giải lượng tử:.....	13
• Quét Zigzag:.....	15
• Entropy Encoding:.....	15
1. Tổng quan:.....	15
2. Mã hóa Run Length Encoding trong JPEG:	16
3. Mã hóa Huffman trong JPEG:	17
3.2 DECODING:.....	17
3.3 ƯU ĐIỂM VÀ NHƯỢC ĐIỂM CỦA JPEG:.....	18
• Ưu điểm:	18
• Nhược điểm:.....	19
PHẦN 4: CHẠY DEMO	20
PHẦN 5: NHẬN XÉT VÀ ĐÁNH GIÁ	21
PHẦN 6: TÀI LIỆU THAM KHẢO	22

LỜI MỞ ĐẦU

Xin cảm ơn khoa Khoa học máy tính và thầy Nguyễn Vinh Tiệp đã cho nhóm có cơ hội được biết và tìm hiểu thêm về kỹ thuật nén ảnh bằng thuật toán JPEG. Từ những kiến thức nền tảng được truyền tải ở mỗi buổi học và việc tìm tòi, học hỏi, nghiên cứu các kiến thức mới, nhóm đã hoàn thành báo cáo về Kỹ thuật nén ảnh bằng thuật toán JPEG. Trong quá trình thực hiện, những sai sót là không thể tránh khỏi. Chính vì vậy, nhóm mong nhận được những ý kiến góp ý từ giảng viên để báo cáo được hoàn thiện hơn.

Khi bước vào kỷ nguyên số, thế giới đã phải đối mặt với một lượng thông tin khổng lồ. Xử lý lượng thông tin khổng lồ này thường có thể dẫn đến nhiều khó khăn. Vì thế chúng ta cần có cách lưu trữ, truy xuất, phân tích và xử lý thông tin kỹ thuật số hiệu quả để được đưa vào sử dụng thực tế. Trong thập kỷ qua, nhiều khía cạnh của công nghệ kỹ thuật số đã được phát triển. Cụ thể trong các lĩnh vực thu nhận hình ảnh, lưu trữ dữ liệu và in bitmap. Một trong nhiều kỹ thuật xử lý ảnh là nén ảnh. Nén hình ảnh có nhiều ứng dụng và đóng vai trò quan trọng trong việc truyền tải và lưu trữ hình ảnh hiệu quả. Việc nén ảnh nhằm mục đích giảm sự dư thừa dữ liệu để lưu trữ hoặc truyền một lượng dữ liệu ít hơn dữ liệu gốc.

Hiện nay, JPEG được xem là 1 định dạng nén ảnh rất được phổ biến. Vậy cụ thể hơn về JPEG là như thế nào? Tại sao nó lại được biết đến rộng rãi như vậy? Làm thế nào để có thể nén ảnh được như vậy? Những thông tin trên đã được nhóm tìm hiểu và trình bày chi tiết trong bài báo cáo này.

PHẦN 1: GIỚI THIỆU

1. Tổng quan:

- Khái niệm nén dữ liệu ảnh: Biến đổi dòng thông tin ảnh nhằm giảm độ dư thừa thông tin. Các phương pháp nén có những đặc trưng khác nhau do định nghĩa các kiểu dư thừa thông tin khác nhau.
- Có ba kiểu dư thừa thông tin được phân loại như sau:
 - Dư thừa mã (Coding Redundancy): biểu diễn thông tin ở dạng mã và có những đoạn mã dư thừa. Để giải quyết vấn đề này, trong nén ảnh thường dùng Run Length Encoding (RLE), mã hóa Huffman (Huffman Encoding),...
 - Dư thừa trong điểm ảnh (Interpixel Redundancy): tương quan giữa các điểm ảnh, các điểm ảnh gần nhau có giá trị gần giống nhau (trừ các điểm ảnh gần đường biên ảnh). Biến đổi Cosine rời rạc (DCT) thường được sử dụng để giảm thiểu độ dư thừa này.
 - Dư thừa tâm lý thị giác (Psychovisual Redundancy): trên thực tế, mắt người không nhạy cảm với 1 số tần số quang phổ nhất định. Các kỹ thuật nén ảnh thường dùng phương pháp lượng tử hóa để làm giảm độ dư thừa này.
- Trọng tâm của các nghiên cứu về nén ảnh là tìm cách giảm số bit cần để biểu diễn ảnh bằng việc loại bỏ các thông tin dư thừa như trên.
- Có nhiều cách phân loại phương pháp nén ảnh khác nhau. Dựa vào nguyên lý nén ta có:
 - Nén không mất mát (Lossless Compression): bao gồm các phương pháp mà sau khi giải nén, ta thu được chính xác dữ liệu gốc.

- Nén có mất mát (Lossy Compression): bao gồm các phương pháp mà sau khi giải nén, ta không thu được toàn bộ dữ liệu gốc.

2. Phương pháp nén ảnh JPEG:

- JPEG là viết tắt của Joint Photographic Experts Group là tên của một nhóm nghiên cứu được thành lập vào năm 1986. Nhóm nghiên cứu đã chọn phép Biến đổi Cosine Rời rạc (Discrete Cosine Transform) là kỹ thuật nén hiệu quả nhất tính đến thời điểm đó và phát triển chuẩn nén JPEG vào cuối thập niên 1980, công bố lần đầu tiên vào năm 1992 và được công nhận là chuẩn ISO/IEC 10918-1 vào năm 1994.
- JPEG là định dạng nén ảnh có mất mát, thường được sử dụng trong lưu và truyền ảnh.
- Xử lý tốt mất mát thông tin trong quá trình nén khiến người nhìn không nhận ra sự khác biệt.
- JPEG có thể thực hiện được trên cả ảnh màu và đen trắng.
- JPEG thông thường đạt tỷ lệ nén 1:10 nhưng không ảnh hưởng nhiều đến cảm nhận của mắt người.

PHẦN 2: Ý TƯỞNG THUẬT TOÁN

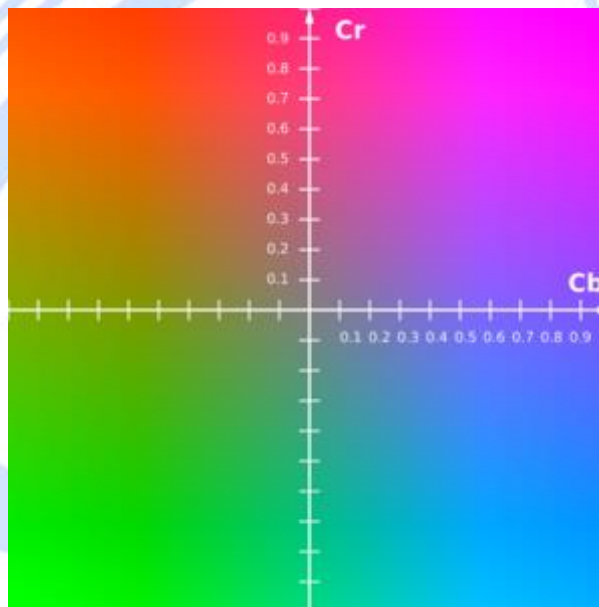
- Ảnh màu trong không gian RGB (Red, Green, Blue) được chuyển đổi qua hệ YCbCr (Luminance, Chroma Blue, Chroma Red) vì JPEG hoạt động trên hệ YCbCr.
- Cắt hình ảnh thành từng khối nhỏ (thường là khối 8x8 điểm ảnh), phân tích giá trị mỗi điểm ảnh và chuyển khối điểm ảnh về ma trận 2 chiều.
- Dùng Biến đổi Cosine Rời rạc để biến đổi tập các giá trị điểm ảnh trên miền không gian thành miền tần số.
- Lượng tử hóa để loại bỏ các tín hiệu ở tần số cao vì đây là thông tin dư thừa trong ảnh.
- Chuyển các ma trận 2 chiều thành vector bằng phép quét Zigzag để thực hiện tối ưu hóa thuật toán nén bằng Mã hóa loạt dài (RLE) và Mã hóa Huffman (Huffman Encoding).
- Khi giải nén ảnh, các bước thực thi sẽ làm ngược lại quá trình trên.

PHẦN 3: CHI TIẾT GIẢI THUẬT

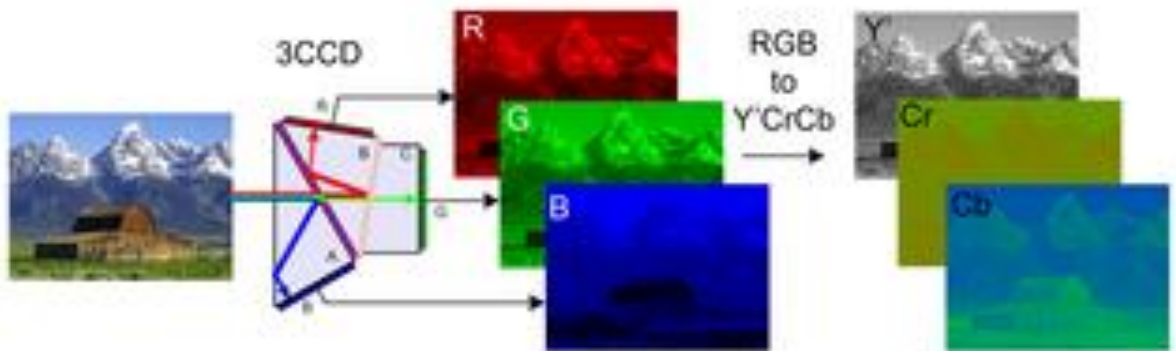
3.1 ENCODING:

- **Biến đổi không gian màu:**

Đầu tiên, hình ảnh nên được chuyển đổi từ mô hình RGB sang mô hình YcbCr. Nó có ba thành phần Y, Cb, Cr. Trong đó thành phần Y đại diện cho độ sáng của pixel, và Cb, Cr là thành phần đại diện cho chrominance (chia thành thành phần xanh và đỏ). Về cơ bản đây là không gian màu được sử dụng trong truyền hình kỹ thuật số cũng như video kỹ thuật số bao gồm các video DVD Không gian màu YcbCr cho phép nén ảnh tốt hơn mà không ảnh hưởng nhiều đến cảm nhận về chất lượng của hình ảnh.



Mặt phẳng CbCr ở hằng số luma $Y = 0,5$ (Wikipedia: <https://vi.wikipedia.org/wiki/YCbCr>)



Chuyển đổi từ RGB sang YCbCr (Wikipedia: <https://vi.wikipedia.org/wiki/YCbCr>)

Công thức biến đổi các thành phần từ mô hình RGB sang mô hình YcbCr (1) và ngược lại (2):

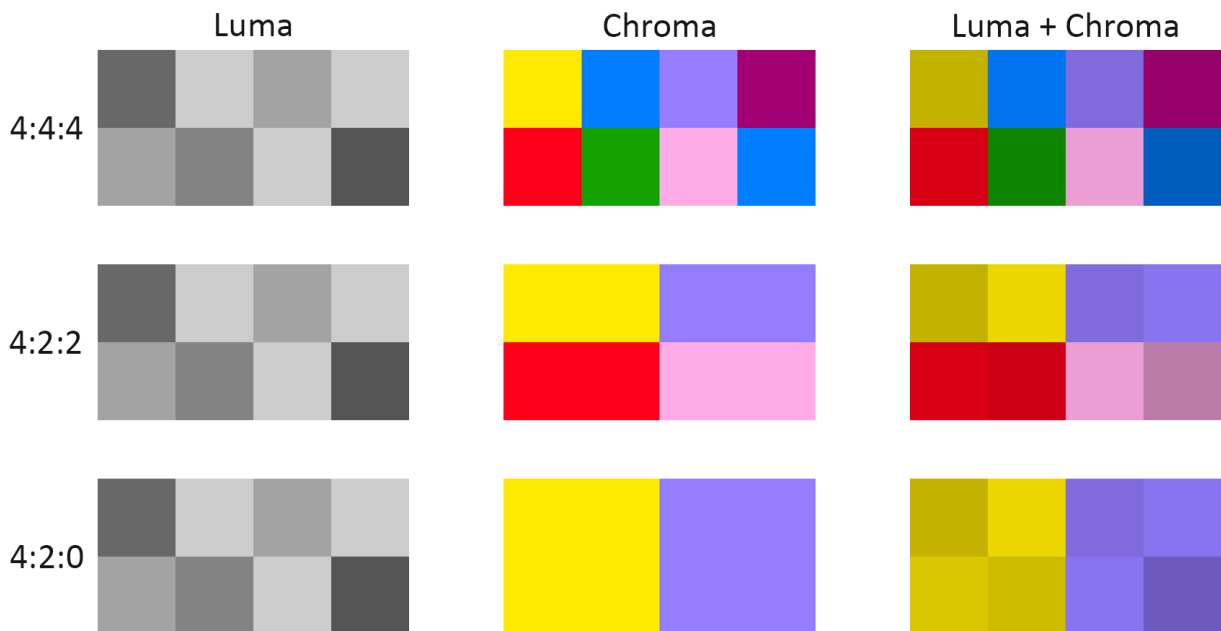
$$\begin{aligned}
 Y &= 0.299R + 0.587G + 0.114B \\
 Cb &= 0.564(B - Y) \\
 Cr &= 0.713(R - Y)
 \end{aligned}
 \tag{1}$$

$$\begin{aligned}
 R &= Y + 1.402Cr \\
 G &= Y - 0.344Cb - 0.714Cr \\
 B &= Y + 1.772Cb
 \end{aligned}
 \tag{2}$$

- **Down sampling:**

- Sau khi chuyển đổi thành hệ YcbCr, vì thị giác của con người lại rất nhạy cảm với hệ Y, ít nhạy cảm với hệ Cb, Cr nên ta sẽ giảm độ phân giải không gian của các thành phần Cb và Cr (được gọi là “Downsampling” hoặc “Subsampling”). Các tỷ lệ mà downsampling thường thực hiện cho ảnh JPEG là:

- **4 : 4 : 4** : số lượng mẫu pixel của thành phần Cb và Cr được giữ nguyên.
- **4 : 2 : 2** : giảm 1 nửa số lượng mẫu các pixel của thành phần Cb và Cr theo hướng nằm ngang, hướng dọc vẫn giữ nguyên.
- **4 : 2 : 0** : giảm 1 nửa các mẫu pixel của thành phần Cb và Cr theo hướng ngang và cả hướng dọc.



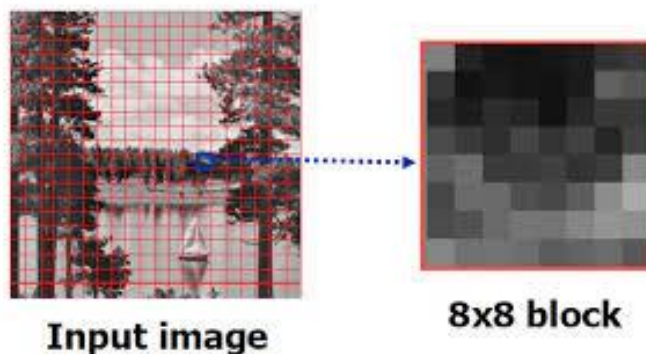
Các tỉ lệ của Chroma Subsampling (Nguồn: <https://www.rtings.com/tv/learn/chroma-subsampling>)

- Đối với những bước còn lại của quá trình nén, Y, Cb và Cr được xử lý riêng nhưng theo cùng một cách giống nhau.

- **Block splitting:**

Trước khi đưa vào biến đổi DCT, ảnh được phân chia thành các khối vuông, mỗi khối này thường có kích thước 8×8 pixel, do:

- Kích thước block lớn hơn 8 làm tăng độ phức tạp thuật toán.
- Kích thước block nhỏ hơn 8 sẽ không đủ thông tin để nén, nén sẽ cho ra kết quả thấp.



- **Discrete cosine transform:**

Biến đổi DCT là một công đoạn chính trong các phương pháp nén JPEG. DCT cho phép biến đổi dữ liệu dưới dạng biên độ thành dữ liệu dưới dạng tần số. Mục đích để loại bỏ sự dư thừa dữ liệu trong không gian. Hai công thức ở đây minh họa cho 2 phép biến đổi DCT thuận nghịch đối với mỗi khối ảnh có kích thước 8×8 . Ở đây chúng ta sẽ tìm hiểu về công thức DCT thuận dùng để nén ảnh.

Đầu tiên, ta cần trừ các giá trị pixel với 128 để chuyển giá trị trung bình về 0 do pixel trong khoảng $(0, 255)$ và DCT hoạt động dựa trên hàm cosin và hàm cosin hoạt động trên khoảng -1 đến 1 .

$$F(u, v) = \frac{1}{4} C(u)C(v) \left[\sum_{x=0}^7 \sum_{y=0}^7 f(x, y) * \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16} \right] \quad (1)$$

$$f(x, y) = \frac{1}{4} \left[\sum_{u=0}^7 \sum_{v=0}^7 C(u)C(v)F(u, v) * \cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16} \right] \quad (2)$$

where: $C(u), C(v) = 1/\sqrt{2}$ for $u, v = 0$;
 $C(u), C(v) = 1$ otherwise.

Muzhir Al-Ani, Fouad H Awad. The Jpeg image impression algorithm .IJAET, tháng 5/2013

Ta gán:

$$\cos \frac{(2x+1)u\pi}{16} \cos \frac{(2y+1)v\pi}{16} \rightarrow C(u, v, x, y)$$

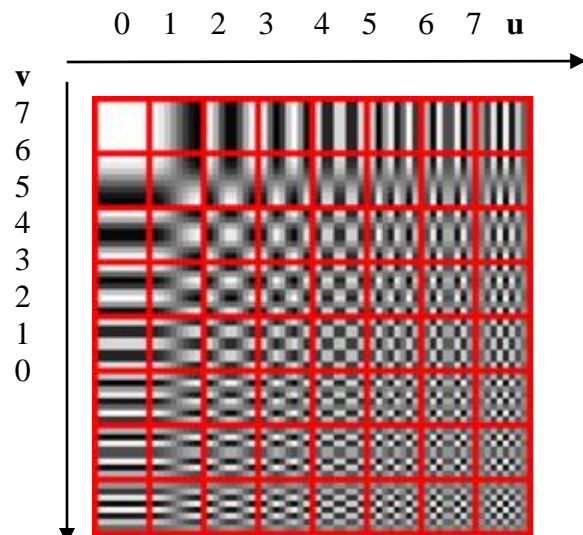
$$C(u)C(v)F(u, v) \rightarrow B(u, v)$$

Trong đó:

- x, y : tọa độ ảnh. ($0 \leq x, y \leq 7$)
- u : tần số theo chiều ngang của hàm cosin. ($0 \leq u \leq 7$)
- v : tần số theo chiều dọc của hàm cosin. ($0 \leq v \leq 7$)
- $f(x, y)$: biểu diễn các mức xám trong miền không gian.
- $F(u, v)$: là các hệ số sau biến đổi DCT trong miền tần số.
- $B(u, v)$: là hệ số DCT.
- $C(u, v, x, y)$: đại diện cho các hàm cosin 2 chiều cơ bản tương ứng với tần số u, v .

Đây là bảng biểu diễn các hàm cosin cơ bản với trục hoành là chỉ số u là tần số theo chiều ngang và trục tung là chỉ số v là tần số theo chiều dọc của hàm cosin.

Nguồn [Wikipedia](https://en.wikipedia.org/wiki/Discrete_Cosine_Transform)



Sau khi gán C vào công thức, ta có:

$$f(x, y) = \sum_{u=0}^{N-1} \sum_{v=0}^{N-1} B(u, v) C(u, v, x, y)$$

Với $N = 8$.

Đây là công thức tạo ra ảnh giống với ảnh nguyên gốc bằng cách tính tổng có trọng số giữa các hàm cosin 2 chiều với tần số u theo chiều ngang và v theo chiều dọc. Ở đây trọng số chính là các hệ số mà các hàm cosin đã đóng góp để tạo nên ảnh gần giống với ảnh gốc nhất.

Mỗi khối 64 điểm ảnh sau biến đổi DCT thuận sẽ nhận được 64 hệ số thực DCT (bảng 1,2). Mỗi hệ số này có chứa một trong 64 thành phần tần số không gian hai chiều. Hệ số với tần số bằng không theo cả hai hướng (tương ứng với x và y bằng 0) được gọi là hệ số DC, hệ số này chính là giá trị trung bình của 64 điểm ảnh trong khối. 63 hệ số còn lại gọi là các hệ số AC. Hệ số một chiều DC tập trung phần lớn năng lượng của ảnh.

52	55	61	66	70	61	64	73
63	59	55	90	109	85	69	72
62	59	68	113	144	104	66	73
63	58	71	122	154	106	70	69
67	61	68	104	126	88	68	70
79	65	60	70	77	68	58	75
85	71	64	59	55	61	65	83
87	79	69	68	65	76	78	94

Bảng 1. Các giá trị mức xám của một khối 64 điểm ảnh

-415.38	-30.19	-61.20	27.24	56.12	-20.10	-2.39	0.46
4.47	-21.86	-60.76	10.25	13.15	-7.09	-8.54	4.88
-46.83	7.37	77.13	-24.56	-28.91	9.93	5.42	-5.65
-48.53	12.07	34.10	-14.76	-10.24	6.30	1.83	1.95
12.12	-6.55	-13.20	-3.95	-1.87	1.75	-2.79	3.14
-7.73	2.91	2.38	-5.94	-2.38	0.94	4.30	1.85
-1.03	0.18	0.42	-2.42	-0.88	-3.02	4.12	-0.66
-0.17	0.14	-1.07	-4.19	-1.17	-0.10	0.50	1.68

Bảng 2. Các hệ số sau khi biến đổi DCT thuận

Chú ý rằng bản thân biến đổi DCT không làm mất thông tin vì DCT là một biến đổi tuyến tính chuyển các giá trị của điểm ảnh từ miền không gian thành các hệ số trong miền tần số. Nếu biến đổi DCT thuận và nghịch được tính toán với độ chính xác tuyệt đối và nếu các hệ số DCT không phải qua bước lượng tử và mã hoá thì ảnh thu được sau biến đổi DCT ngược sẽ giống hệt ảnh gốc.

- **Lượng tử và giải lượng tử:**

Sau khi thực hiện biến đổi DCT, những thành phần có giá trị tần số cao sẽ bị loại bỏ qua quá trình lượng tử hóa dựa trên một bảng lượng tử gồm 64 phần tử $Q(x,y)$ với $0 \leq x, y \leq 7$. Dưới đây là 2 bảng lượng tử hóa chuẩn cho Y và Cb, Cr:

16	11	10	16	24	40	51	61
12	12	14	19	26	58	60	55
14	13	16	24	40	57	69	56
14	17	22	29	51	87	80	62
18	22	37	56	68	109	103	77
24	35	55	64	81	104	113	92
49	64	78	87	103	121	120	101
72	92	95	98	112	100	103	99

Bảng lượng tử hóa cho thành phần Y

17	18	24	47	99	99	99	99
18	21	26	66	99	99	99	99
24	26	56	99	99	99	99	99
47	66	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99
99	99	99	99	99	99	99	99

Bảng lượng tử hóa cho thành phần Cb và Cr

Các phần tử trong bảng lượng tử có giá trị từ 1 đến 255 được gọi là các bước nhảy cho các hệ số DCT. Các phần tử đầu đều có giá trị nhỏ và dần tăng lên khi về phía cuối bảng lượng tử, điều này giúp việc loại bỏ các thành phần có tần số cao về giá trị là 0 và những thành phần có tần số nhỏ sẽ chỉ bị giảm đi đôi chút. Quá trình lượng tử sẽ chia các hệ số DCT của mỗi block cho bước nhảy lượng tử tương ứng, kết quả này sau đó sẽ được làm tròn xuống số nguyên gần nhất. Công thức (3) thể hiện việc lượng tử hóa F với bảng lượng tử Q:

$$F'(x, y) = \frac{F(x, y)}{Q(x, y)} \quad (3)$$

Trong đó:

- $F(x, y)$: phần tử hệ số DCT theo vị trí (x, y) trong block.
- $Q(x, y)$: phần tử ứng với vị trí (x, y) trong bảng lượng tử.
- $F'(x, y)$ là hệ số sau khi lượng tử hóa ứng với vị trí (x, y) trong block.

-26	-3	-6	2	2	-1	0	0
0	-3	4	1	1	0	0	0
-3	1	5	-1	-1	0	0	0
-4	1	2	-1	0	0	0	0
1	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0

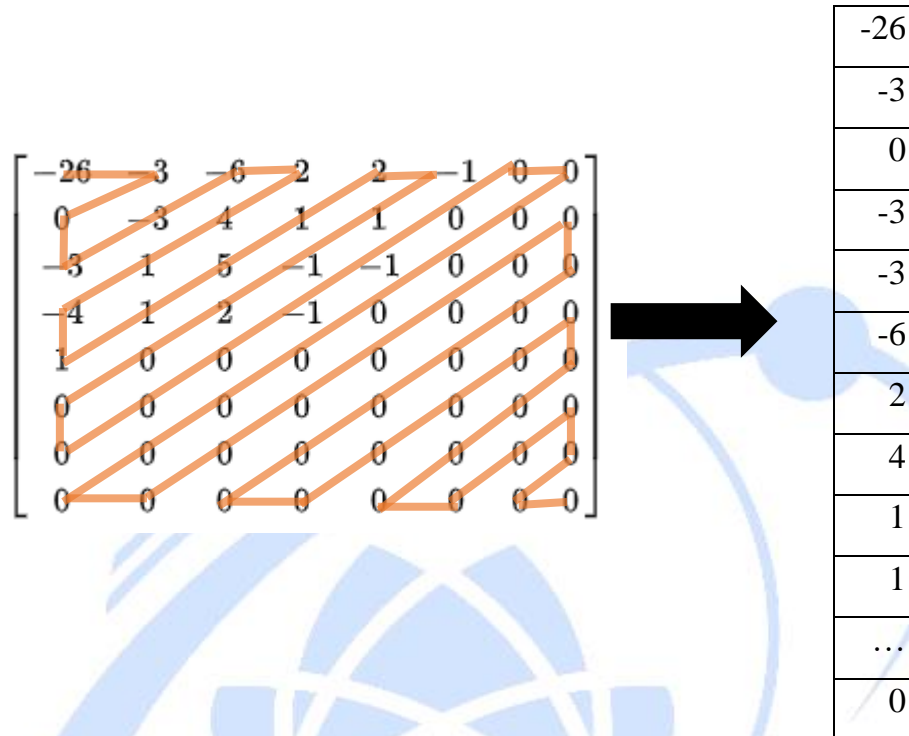
Bảng 3. Các hệ số sau lượng tử hóa

Mục đích của việc lượng tử hoá là giảm số lượng bit cần để lưu trữ các hệ số biến đổi bằng việc giảm độ chính xác của các hệ số này cho nên lượng tử là quá trình xử lý có mất thông tin.

Quá trình giải lượng tử ở phía bộ giải mã được thực hiện ngược lại. Các hệ số sau bộ giải mã entropy sẽ nhân với các bước nhảy trong bảng lượng tử (bảng lượng tử được đặt trong phần header của ảnh JPEG). Kết quả này sau đó sẽ được đưa vào biến đổi DCT ngược.

- **Quét Zigzag:**

Do quá trình mã hóa cần đầu vào là 1 vector nên cần phải quét zigzag để chuyển các block từ ma trận 2 chiều thành vector:



- **Entropy Encoding:**

1. Tổng quan:

Chuẩn nén ảnh JPEG hiện nay dùng phương pháp mã hoá Run Length Coding và Huffman, đây là những phép mã hoá không làm mất thông tin.

RLE hoạt động bằng cách tìm loạt dữ liệu liên nhau lặp lại trong chuỗi dữ liệu thành một dữ liệu khác, mục đích là để giảm kích thước dữ liệu gốc.

VD: Ta có 1 chuỗi như sau:

Sau khi sử dụng RLE ta nhận được kết quả:

12W1B12W3B24W1B14W

Ta đã giảm kích thước của chuỗi kí tự từ 67 kí tự về còn 18 kí tự.

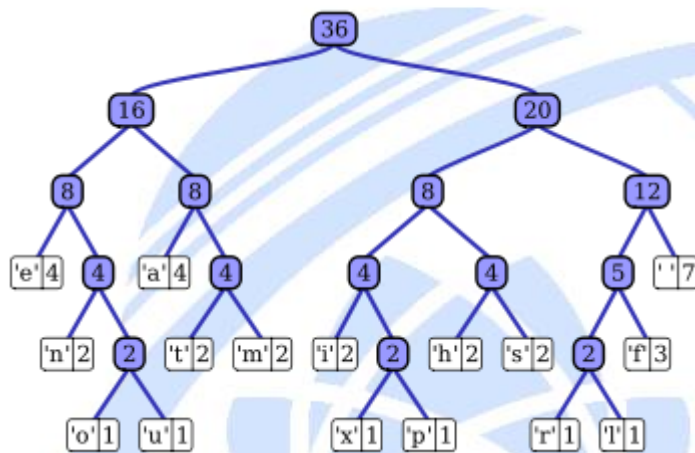
(Nguồn: [Wikipedia](#))

Phương pháp mã hóa Huffman dựa trên bảng tần suất xuất hiện các ký tự cần mã hóa để xây dựng một bộ mã nhị phân cho các ký tự đó sao cho dung lượng (số bit) sau khi mã hóa là nhỏ nhất.

Ta có 1 chuỗi ký tự:

"this is an example of a huffman tree"

Sau khi sử dụng Huffman, ta có cây Huffman:



(Nguồn: [Wikipedia](https://en.wikipedia.org/wiki/Huffman_coding))

2. Mã hóa Run Length Encoding trong JPEG:

Sau bước Zigzag sẽ tạo ra 1 vector với 1 dãy nhiều số 0 ở đằng sau, nên ta sử dụng Run Length Encoding để rút gọn vector này. Vector sẽ được mã hóa theo mẫu (**skip**, **value**), trong đó **skip** là số lượng các số 0 ở phía trước **value** và **value** là phân tử khác 0. Sau đó thêm phân tử (0, 0) và cuối để thể hiện sự kết thúc cho 1 block (End – of – block).

Vd: -26, 3, ..., -1, -1, 0, 0, ..., 0

Qua bước RLC sẽ cho ra kết quả:

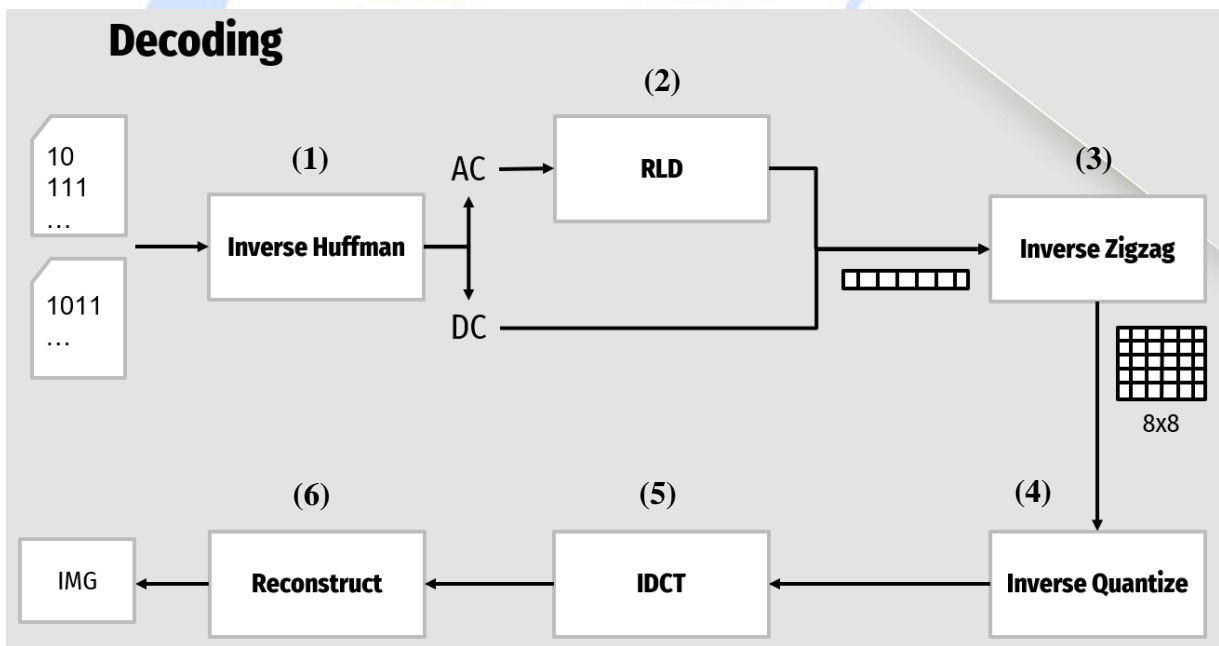
(0, -26), (0, 3), ..., (0, -1), (0, -1), (38, 0), (0, 0).

3. Mã hóa Huffman trong JPEG:

Phương pháp mã hóa Huffman tính tần suất xuất hiện các hệ số. Việc tính tần suất được thực hiện bằng cách duyệt tuần tự từ đầu khối đến cuối khối, sau đó, những hệ số có tần suất cao được gán cho một từ mã ngắn, các hệ số có tần suất thấp được gán một từ mã dài. Với cách thức này chiều dài trung bình của từ mã đã giảm xuống.

3.2 DECODING:

Để coi được ảnh encode theo thuật toán JPEG, chúng ta cần phải trải qua quá trình decode để có thể xuất ảnh ra, vì đây là thuật toán nén ảnh có mất mát nên việc xuất ảnh ra sẽ không đạt chất lượng như ban đầu chưa nén. Dưới đây là hình ảnh mô phỏng quá trình decode ảnh đã nén JPEG:



Ảnh mô phỏng quá trình giải nén ảnh JPEG

- **Bước 1:** Ảnh sau khi mã hóa sẽ là 1 file chứa các bit do Huffman mã hóa. Để đọc được ta phải có 1 bảng mã Huffman và 1 chuỗi đã được mã hóa Huffman, 2 file này đi qua bước Inverse Huffman sẽ trả về vector DC và vector AC
- **Bước 2:** Riêng vector AC do đã đi qua bước Run Length Encoding nên quá trình Decoding cần đi qua 1 bước Run Length Decoding để trả về mảng AC 2 chiều hoàn chỉnh với kích thước(, 63).
- **Bước 3:** Sau đó mảng AC và vector DC được gộp lại theo chiều ngang sẽ trở thành mảng 2 chiều có kích thước(, 64). Sau đó, ta thực hiện bước Inverse Zigzag để chuyển vector về lại block 8x8.
- **Bước 4:** Sau khi đã có các block 8x8, ta qua bước Inverse Quantize trong bước này ta nhân từng phần tử DCT với từng phần tử trong bảng lượng tử.
- **Bước 5:** Bước IDCT (Inverse DCT) là bước nghịch đảo của DCT sẽ đổi từ miền tần số trở lại thành miền không gian.
- **Bước 6:** Là bước cuối cùng để decode 1 ảnh nén theo thuật toán JPEG. Ảnh đã về lại miền không gian sẽ được biến đổi từ hệ YCbCr về lại hệ RGB. Sau đó xuất ảnh ra.

3.3 ƯU ĐIỂM VÀ NHƯỢC ĐIỂM CỦA JPEG:

- **Ưu điểm:**
 - Chuẩn nén JPEG cho phép tỉ số nén lên đến 80:1 tức sẽ tiết kiệm được 80% lượng dữ liệu dùng để lưu tệp này trong giới hạn chất lượng ảnh vẫn còn tốt. Khả năng nén hiệu quả giúp JPEG được ưa chuộng hơn trong việc lưu trữ, truyền tải thông tin qua internet khi so sánh với các định dạng ảnh khác như BMP và PNG.
 - Lợi ích đáng chú ý của chuẩn JPEG là nó hỗ trợ hiển thị các hình ảnh với các màu 24-bit chính xác (true-colour) có thể lên đến 16 triệu màu. Điều đó

cho phép nó được sử dụng tốt nhất cho các ảnh chụp và ảnh minh họa với lượng màu lớn và có độ phân giải cao.

- Ảnh JPEG tương thích với nhiều ứng dụng và thiết bị phần cứng. Điều này giúp chuẩn nén JPEG càng phổ biến hơn trong việc truyền tải dữ liệu trên mạng cũng như kết nối với các thiết bị điện tử khác.
- Một đặc trưng khác của JPEG là người dùng có thể điều chỉnh mức độ nén trên một bức ảnh kỹ thuật số sao cho hợp lý. Cụ thể hơn là trong việc đánh đổi chất lượng ảnh và kích thước tệp ảnh.

• **Nhược điểm:**

- Nhược điểm lớn nhất của chuẩn JPEG là chất lượng ảnh sau khi nén bị mất đi một phần do hệ quả của việc nén mất mát dữ liệu (do DCT, lượng tử hóa). Tỷ số nén càng cao thì sự mất mát thông tin trên ảnh càng lớn và cứ mỗi lần lưu ảnh thì chất lượng ảnh càng bị giảm đi và không thể khôi phục.
- Chuẩn JPEG chỉ hiệu quả trên các ảnh kỹ thuật số với những vùng chuyển màu nhẹ. Thực hiện nén trên các ảnh có văn bản hay có nhiều góc cạnh sẽ làm các đường bao giữa các khối màu có thể xuất hiện nhiều điểm mờ, và các vùng sẽ mất đi sự rõ nét (do DCT, lượng tử hóa).
- Chuẩn JPEG trở nên không hiệu quả trên những vùng mờ hoặc trong suốt (các loại khuôn mẫu, logo, nút,...). Trong trường hợp này định dạng PNG hoặc GIF sẽ trở nên hiệu quả hơn.

PHẦN 4: CHẠY DEMO

Ở phần chạy Demo nhóm đã sử dụng 2 bức ảnh không mờ và 2 bức ảnh mờ để so sánh về thời gian nén và giải nén của từng loại và đồng thời là dung lượng ảnh sau khi nén sẽ khác nhau như thế nào. Dưới đây là các ảnh nhóm đã chọn cùng với bảng kết quả và ảnh sau khi giải nén:



Ảnh đầu vào

Bảng kết quả sau khi sử dụng thuật toán JPEG

No.	Kích thước ảnh	Dung lượng ảnh ban đầu	Dung lượng ảnh sau khi nén	Tỉ lệ nén r	Thời gian nén	Thời gian giải nén
1	750x750	2.15 MB	115.82 KB	0.229	1.153 s	1.991 s
2	1920x1080	7.91 MB	334.19 KB	0.170	4.097 s	6.715 s
3	1920x1080	7.91 MB	107.11 KB	0.056	3.403 s	3.896 s
4	1280x800	2.93 MB	59.94 KB	0.070	1.681 s	1.958 s



Ảnh sau khi decode

Nhận xét:

- Dung lượng ảnh giảm đi nhiều so với ban đầu.
- Thuật toán chạy với ảnh có tần số ánh sáng cao hoặc có kích thước lớn sẽ chiếm nhiều thời gian hơn ảnh có tần số ánh sáng thấp hoặc kích thước nhỏ.
- Ảnh có tần số ánh sáng thấp sẽ cho hệ số nén tốt hơn ảnh có tần số ánh sáng cao.
- Sau lần nén đầu tiên độ mờ của ảnh không đáng kể.

PHẦN 5: NHẬN XÉT VÀ ĐÁNH GIÁ

Nén ảnh là một phần cực kỳ quan trọng của điện toán hiện đại. Việc nén làm kích thước ảnh nhỏ hơn so với kích thước gốc giúp tiết kiệm đáng kể không gian bộ nhớ nhất là trong thời đại hiện nay, khi mọi người phải tiếp xúc với một lượng rất lớn các ảnh hằng ngày thông qua công việc hay đơn giản hơn thông qua sự phát triển của mạng xã hội. Thêm vào đó, việc truyền ảnh trên không gian mạng trở nên dễ dàng hơn cũng là một trong những lý do mà nén ảnh đóng vai trò quan trọng trong sự phát triển của internet. Mặc dù việc triển khai thuật toán JPEG khó khăn hơn so với các định dạng ảnh khác và quá trình nén ảnh trên thực tế rất phức tạp về mặt tính toán thì tỷ lệ nén cao đạt được sau khi áp dụng JPEG bù lại thỏa đáng lượng thời gian dành cho việc triển khai thuật toán vì kích thước tệp nén giảm đi đáng kể so với kích thước gốc trong khi chất lượng ảnh vẫn được đảm bảo.

PHẦN 6: TÀI LIỆU THAM KHẢO

- [1] Pao-Yen Lin , " Basic Image Compression Algorithm and Introduction to JPEG Standard" , National Taiwan University, Taipei, Taiwan, ROC 2009.
- [2] G.M.Padmaja and P.Nirupama , " Analysis of Various Image Compression Techniques", ARPN Journal of Science and Technology 2011- 2012. All rights reserved.
- [3] R.C. Gonzalez, and R. E. Woods, “Digital Image Processing,” vol. 2, Prentice Hall, 2002.
- [4] A. K. Jain, “Fundamentals of Digital Image Processing,” vol. 3, Englewood Cliffs, Prentice Hall, 1989.
- [5] L. M. Bá, N. T. Thủy, “Nhập môn xử lý ảnh,” Trường Đại học Bách Khoa Hà Nội, 2008.
- [6] R.C. Gonzalez, R. E. Woods, “Digital Image Processing using Matlab”.
- [7] Bhabatosh Chanda, Dwijesh Dutta Maumder, “Digital Image Processing and Analysis”, Prentice Hall of India, 2001.
- [8] William K. Pratt, “Digital Image Processing: PIKS inside”, Third Edition 2001, John Wiley & Sons Inc.
- [9] JPEG, "Joint Photographic Experts Group, JPEG Homepage", retrieved 2009-11-08.
- [10] JPEG Wikipedia. <https://en.wikipedia.org/wiki/JPEG>