

Universidad Peruana de Ciencias Aplicadas  
Escuela de Ingeniería de Sistemas y Computación  
Carrera de Ciencias de la Computación

**CC235 Procesamiento de Imágenes**

# Compresión con pérdida: Formato JPEG

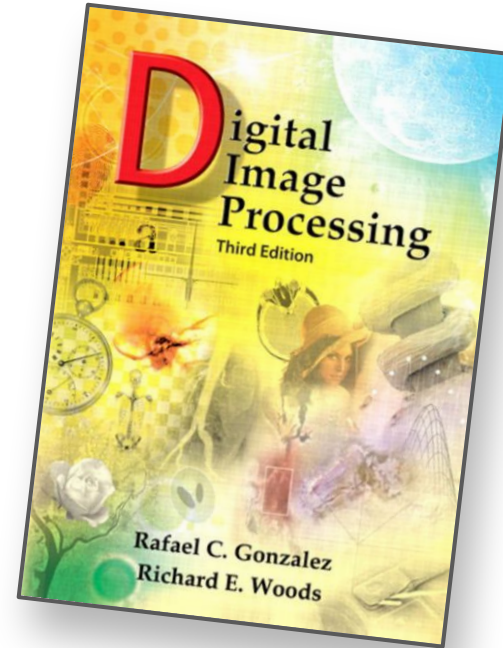
Prof. Peter Montalvo García

# Agenda

- Introducción
- Formato JPEG
- Pasos de la compresión en JPEG

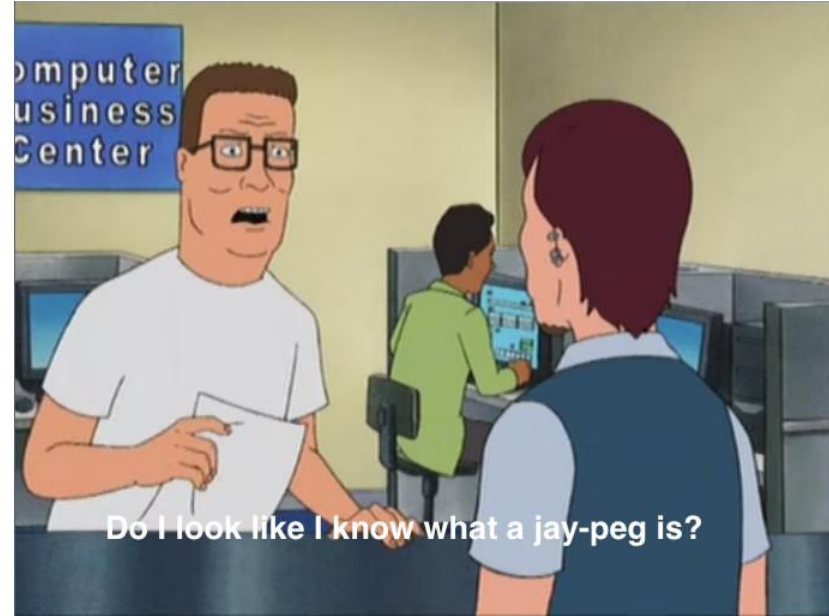
# Nota

- Esta sesión está basada en el libro “Digital Image Processing” 3ra edición de Rafael C. González y Richard E. Woods. En especial el capítulo 8



# Nota

- JPEG 101 - How does JPEG work?  
<https://arjunsreedharan.org/post/146070390717/jpeg-101-how-does-jpeg-work>



# Introducción



42 MB  
Formato RAW



4.1 MB  
Formato JPEG

# Introducción

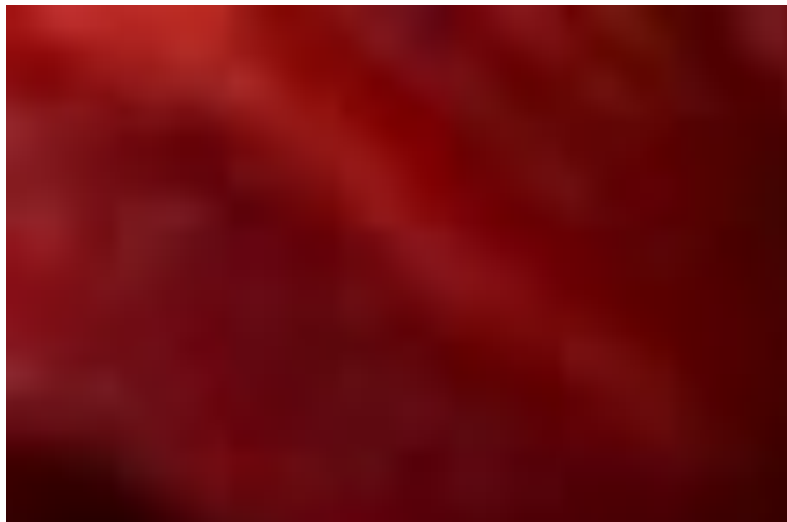


42 MB  
Formato RAW

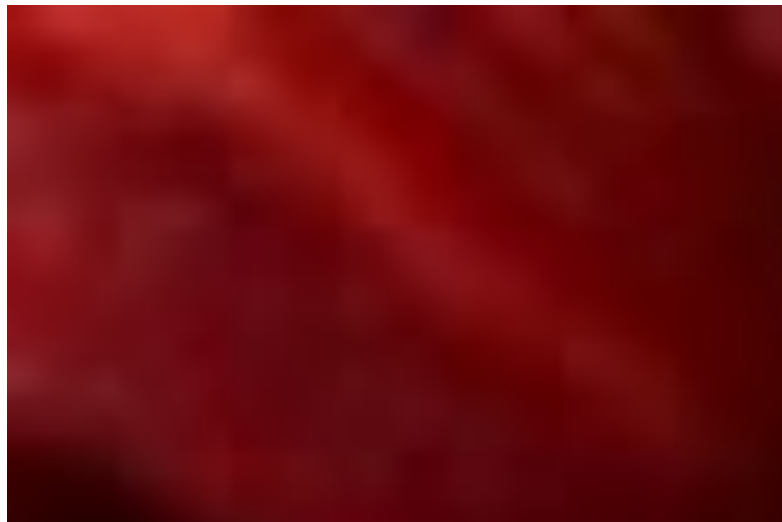


4.1 MB  
Formato JPEG

# Introducción



42 MB  
Formato RAW



4.1 MB  
Formato JPEG

# ¿Qué es JPEG?

- Joint Photographic Experts Group (JPEG) es el nombre de un comité de expertos que creó un estándar de compresión y codificación de archivos e imágenes fijas, que es actualmente uno de los formatos más utilizados para fotografías



Foto de una flor comprimida gradualmente con el formato JPEG.

[https://es.wikipedia.org/wiki/Joint\\_Photosraphic\\_Experts\\_Group](https://es.wikipedia.org/wiki/Joint_Photosraphic_Experts_Group)



# ¿Qué es JPEG?

- Joint Photographic Experts Group (JPEG) es el nombre de un comité de expertos que creó un estándar de compresión y codificación de archivos e imágenes fijas, que es actualmente uno de los formatos más utilizados para fotografías
- JPEG también es un método de compresión con pérdida aplicado a imágenes (que veremos en la siguiente sección)



Foto de una flor comprimida gradualmente con el formato JPEG.

[https://es.wikipedia.org/wiki/Joint\\_Photoscientific\\_Group](https://es.wikipedia.org/wiki/Joint_Photoscientific_Group)

# ¿Qué es JPEG?

- Joint Photographic Experts Group (JPEG) es el nombre de un comité de expertos que creó un estándar de compresión y codificación de archivos e imágenes fijas, que es actualmente uno de los formatos más utilizados para fotografías
- JPEG también es un método de compresión con pérdida aplicado a imágenes (que veremos en la siguiente sección)
- JPEG también es un formato de archivo de imagen. En Windows, su equivalente es JPG (debido a que manejan 3 caracteres como archivo)



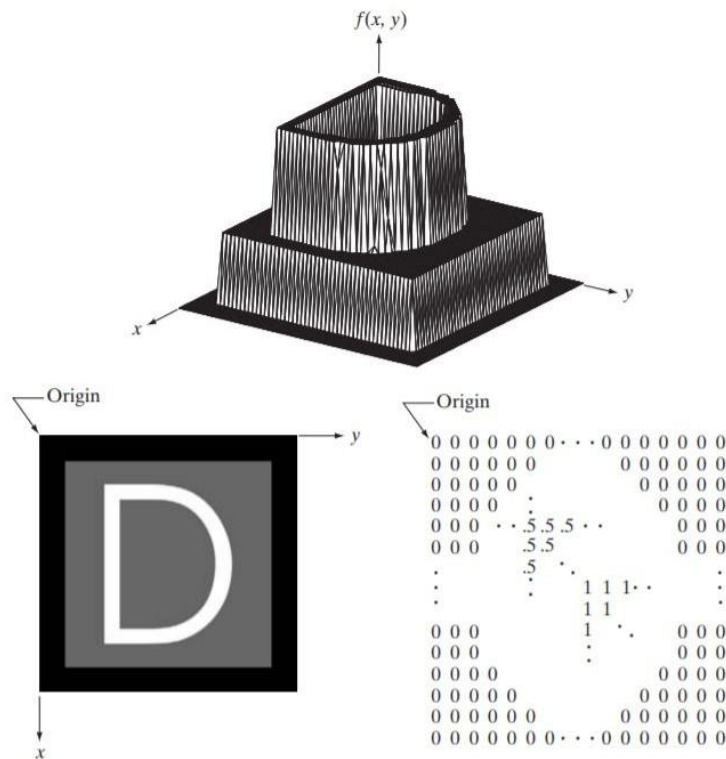
Foto de una flor comprimida gradualmente con el formato JPEG.

[https://es.wikipedia.org/wiki/Joint\\_Photoshographic\\_Experts\\_Group](https://es.wikipedia.org/wiki/Joint_Photoshographic_Experts_Group)

# Formato JPEG

- Cuando hablamos de un archivo en formato JPEG, se hace referencia a JFIF (formato de intercambio de archivos JPEG).

# Recordando cómo se obtienen las imágenes



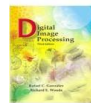
a  
b c

**FIGURE 2.18**

(a) Image plotted as a surface.

(b) Image displayed as a visual intensity array.

(c) Image shown as a 2-D numerical array (0, .5, and 1 represent black, gray, and white, respectively).

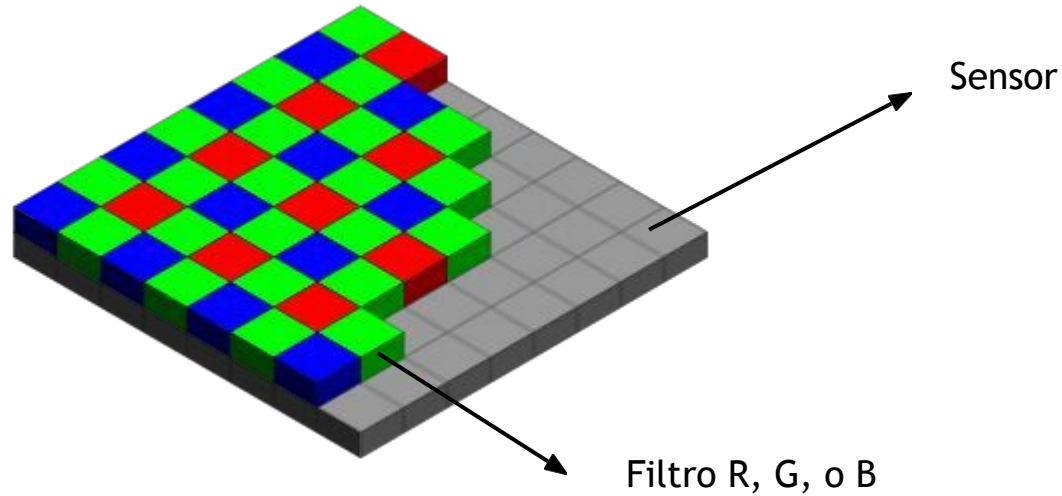


*Digital Image Processing, 3rd ed.*

Gonzalez & Woods  
www.ImageProcessingPlace.com

Chapter 2  
Digital Image Fundamentals

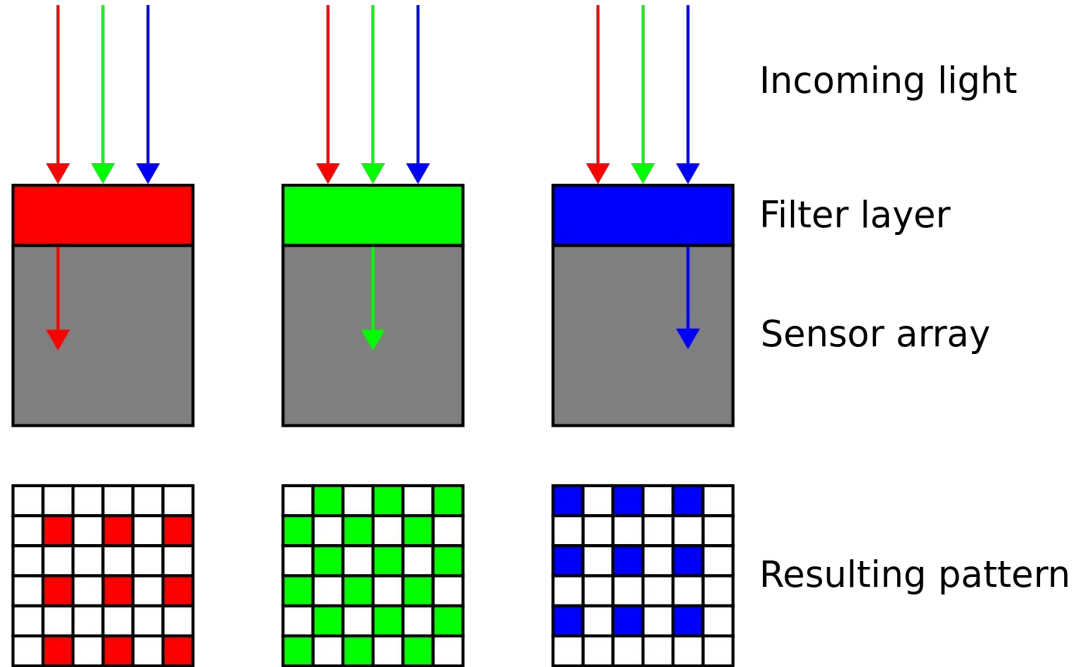
# Recordando cómo se obtienen las imágenes



By en:User:Cburnett - Own work, CC BY-SA 3.0,  
<https://commons.wikimedia.org/w/index.php?curid=1496858>

# Recordando cómo se obtienen las imágenes

- El filtro funciona al dejar pasar solo una longitud de onda (o color) a cada elemento de la grilla del sensor
- Se interpola para los valores faltantes



# Recordando cómo se obtienen las imágenes

- El filtro tiene
  - 50% verde -- *para simular la fisiología del ojo humano*
  - 25% rojo
  - 25% azul
- Patente de Bryce E. Bayer (Eastman Kodak)
- Año 1975

United States Patent [19]  
Bayer

[11] 3,971,065  
[45] July 20, 1976

[54] COLOR IMAGING ARRAY

[75] Inventor: Bryce E. Bayer, Rochester, N.Y.

[73] Assignee: Eastman Kodak Company,  
Rochester, N.Y.

[22] Filed: Mar. 5, 1975

[21] Appl. No.: 555,477

[52] U.S. Cl. .... 358/41; 350/162 SF;  
350/317; 358/44

[51] Int. Cl.<sup>2</sup> ..... H04N 9/24

[58] Field of Search ..... 358/44, 45, 46, 47,  
358/48; 350/317, 162 SF; 315/169 TV

[56] References Cited

UNITED STATES PATENTS

2,446,791	8/1948	Schroeder	358/44
2,508,267	5/1950	Kasperowicz	358/44
2,884,483	4/1959	Ehrenhaft et al.	358/44
3,725,572	4/1973	Kurokawa et al.	358/46

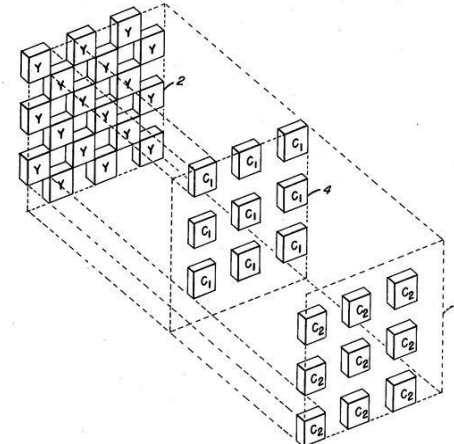
Primary Examiner—George H. Libman  
Attorney, Agent, or Firm—George E. Grosser

[57] ABSTRACT

A sensing array for color imaging includes individual luminance- and chrominance-sensitive elements that are so intermixed that each type of element (i.e., according to sensitivity characteristics) occurs in a repeated pattern with luminance elements dominating the array. Preferably, luminance elements occur at every other element position to provide a relatively high-frequency sampling pattern which is uniform in two perpendicular directions (e.g., horizontal and vertical). The chrominance patterns are interlaid therewith and fill the remaining element positions to provide relatively lower frequencies of sampling.

In a presently preferred implementation, a mosaic of selectively transmissive filters is superposed in registration with a solid state imaging array having a broad range of light sensitivity, the distribution of filter types in the mosaic being in accordance with the above-described patterns.

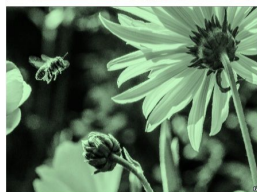
11 Claims, 10 Drawing Figures



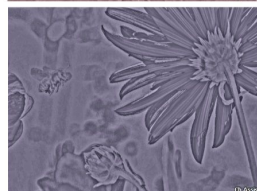
# Luminancia y Crominancia



RGB



YCrCb





# Luminancia

- Se define como la densidad angular, rectangular y superficial de flujo luminoso que incide, atraviesa o emerge de una superficie siguiendo una dirección determinada. Alternativamente, también se puede definir como la densidad superficial de intensidad luminosa en una dirección dada.



<https://en.wikipedia.org/wiki/Luminance#/media/File:TealightLuminanceImage.jpg>

# Crominancia

- La crominancia es la señal que en los sistemas de vídeo transporta la información de color de la imagen, separadamente de la señal luma o señal Y.



# Formato JPEG

- El archivo de imagen sin procesar así obtenido es un mapa de bits (matriz 2D). Es de tamaño muy grande.
- Entonces, se debe comprimir el archivo de imagen. Una de las técnicas más útiles es JPEG.
- JPEG es una técnica de compresión con pérdida, lo que significa que utiliza aproximaciones y descartes parciales de datos para comprimir el contenido. Por lo tanto es irreversible.

# Formato JPEG

- La compresión JPEG se basa en las siguientes 2 observaciones:
- Observación n.º 1 : los ojos humanos no ven el color (crominancia) tan bien como nosotros vemos el brillo (luminancia).
- Observación #2 : Los ojos humanos no pueden distinguir cambios de alta frecuencia en la intensidad de la imagen.

# Pasos para la compresión en JPEG

1. Conversión del espacio de color RGB a YCbCr
2. Reducción de muestreo
3. Aplicación de la Transformada de Coseno Discreta (DCT)
4. Cuantización
5. Codificación
6. Agregar el encabezado

# Conversión del espacio de color RGB a YCbCr

Cada píxel de su imagen se almacena como una combinación aditiva de valores de rojo, azul y verde. Cada uno de estos valores puede estar en el rango de:

A) 0 - 255

B) 0 - 256

C) 0 - 125

# Conversión del espacio de color RGB a YCbCr

Cada píxel de su imagen se almacena como una combinación aditiva de valores de rojo, azul y verde. Cada uno de estos valores puede estar en el rango de:

A) 0 - 255

B) 0 - 256

C) 0 - 125

# Conversión del espacio de color RGB a YCbCr

Cada píxel de su imagen se almacena como una combinación aditiva de valores de rojo, azul y verde. Cada uno de estos valores puede estar en el rango de 0 a 255.

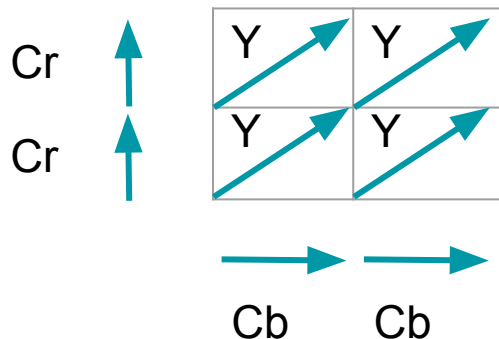
La luminancia es más importante que el color para la eventual calidad de percepción de la imagen. Así que convertimos del espacio de color RGB a uno donde la luminancia se limita a un solo canal. Este espacio de color se llama YCbCr.



# Reducción de muestreo

Dado que la crominancia no es muy importante, podemos reducir la muestra y reducir la cantidad de color (componentes CbCr).

Generalmente, el color se reduce por un factor de 2 en ambas direcciones (vertical y horizontal), es decir, se muestrea Y en cada píxel, mientras que Cb y Cr se muestrean en cada bloque de 2x2 píxeles.



# Reducción de muestreo

Es decir, por cada 4 píxeles Y, existirá solo 1 píxel CbCr.

No se notará mucho cambio en la imagen, pero se reduce una buena cantidad del tamaño del archivo.

En el software de edición de imágenes, generalmente se le pregunta qué calidad desea que se guarde la imagen. De hecho, este es el software que le pregunta cuánta reducción de resolución desea que haga en la imagen.



# Aplicación de la transformada de coseno discreta (DCT)

Cada uno de los tres componentes YCbCr se comprime y codifica por separado utilizando el método DCT.

# Aplicación de la transformada de coseno discreta (DCT)

Cada uno de los tres componentes YCbCr se comprime y codifica por separado utilizando el método DCT.

DCT es un método que expresa una secuencia finita de puntos de datos en términos de una suma de funciones coseno que oscilan a diferentes frecuencias.

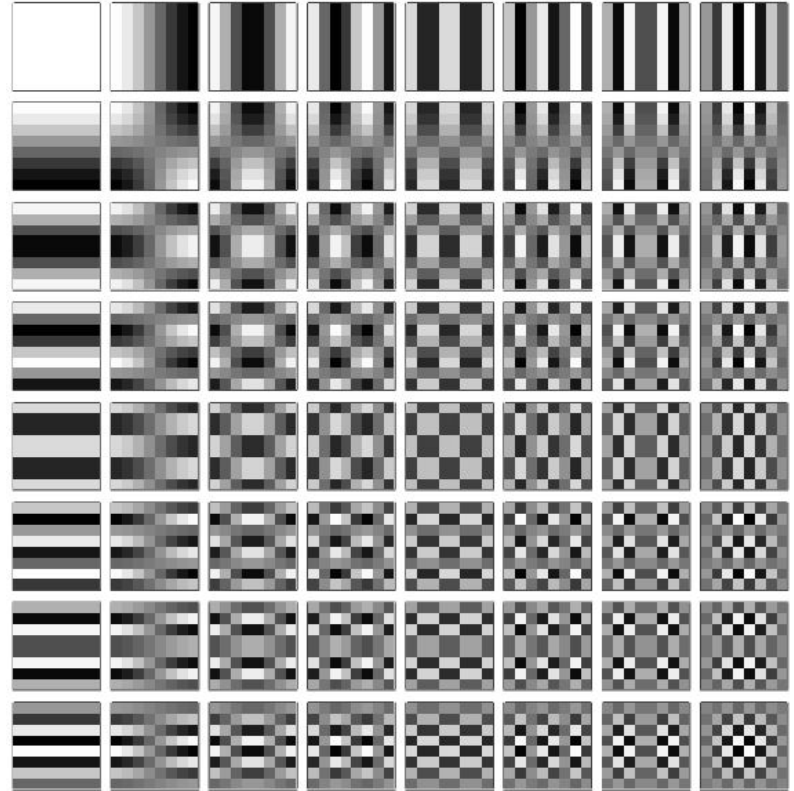
# Aplicación de la transformada de coseno discreta (DCT)

Para la compresión, se utilizan funciones de coseno en lugar de funciones de seno debido a una diferencia particular en el comportamiento de sus límites.

Una función como el brillo de una imagen no necesita tomar valores cero en el límite como lo hace el seno. Entonces, es difícil aproximar tal señal por una combinación lineal de senos.

# Aplicación de la transformada de coseno discreta (DCT)

- Estas son 64 imágenes base, que se construyen a partir de funciones coseno a diferentes frecuencias en los ejes X e Y.
- Esto nos ayuda a obtener las imágenes base sobre las que se obtendrán las subimágenes.



# Aplicación de la transformada de coseno discreta (DCT)

La imagen completa que queremos comprimir se divide en subimágenes, cada una de las cuales consta de 8x8 píxeles. Llamemos a cada uno de ellos como una subimagen.

# Aplicación de la transformada de coseno discreta (DCT)

La imagen completa que queremos comprimir se divide en subimágenes, cada una de las cuales consta de 8x8 píxeles. Llamemos a cada uno de ellos como una subimagen.

Original Sub-image

64	60	57	56	48	47	47	43
61	58	53	52	48	49	52	53
67	60	53	53	49	47	48	54
68	61	63	63	62	65	65	64
71	61	70	63	69	74	88	88
83	94	102	105	107	111	110	115
95	108	108	124	122	130	128	128
107	118	125	134	137	142	141	137

<https://arjunsreedharan.org/post/146070390717/jpeg-101-how-does-jpeg-work>



# Aplicación de la transformada de coseno discreta (DCT)

Como vamos a usar DCT, y las ondas de coseno van de 1 a -1, vamos a centrar nuestros valores alrededor de cero. Esto significa que cambiamos el rango de  $[0..255]$  a  $[-128..128]$ . Entonces restamos 128 de cada valor.

Shifted sub-image

-64	-68	-71	-72	-80	-81	-81	-85
-67	-70	-75	-76	-80	-79	-76	-75
-61	-68	-75	-75	-79	-81	-80	-74
-60	-67	-65	-65	-66	-63	-63	-64
-57	-67	-58	-65	-59	-54	-40	-40
-45	-36	-26	-23	-21	-17	-18	-13
-33	-20	-20	-4	-6	2	0	0
-21	-10	-3	6	9	14	13	9

<https://arjunsreedharan.org/post/146070390717/jpeg-101-how-does-jpeg-work>

# Aplicación de la transformada de coseno discreta (DCT)

Ahora, tenemos 2 cosas en nuestra mano:

1. La subimagen de 8x8 que se comprimirá
2. 64 imágenes base

# Aplicación de la transformada de coseno discreta (DCT)

Se debe transformar la subimagen en una combinación lineal de estas 64 imágenes base.

La subimagen se puede convertir a esta representación en el dominio de la frecuencia utilizando una transformada de coseno discreta (DCT) de tipo II normalizada y bidimensional.

Podemos pensar en la subimagen como compuesta por un conjunto ponderado de estas 64 imágenes base fusionadas una encima de la otra.

Por lo tanto,  $\text{subimagen} = C_1f_1 + C_2f_2 + C_3f_3 + \dots + C_{64}f_{64}$   
donde  $C_i$  es una constante y  $f_i$  son las imágenes base.

# Aplicación de la transformada de coseno discreta (DCT)

Se debe transformar la subimagen en una combinación lineal de estas 64 imágenes base.

La subimagen se puede convertir a esta representación en el dominio de la frecuencia utilizando una transformada de coseno discreta (DCT) de tipo II normalizada y bidimensional.

Podemos pensar en la subimagen como compuesta por un conjunto ponderado de estas 64 imágenes base fusionadas una encima de la otra.

Por lo tanto,  $\text{subimagen} = C_1f_1 + C_2f_2 + C_3f_3 + \dots + C_{64}f_{64}$   
donde  $C_i$  es una constante y  $f_i$  son las imágenes base.

# Aplicación de la transformada de coseno discreta (DCT)

Shifted sub-image

-64	-68	-71	-72	-80	-81	-81	-85
-67	-70	-75	-76	-80	-79	-76	-75
-61	-68	-75	-75	-79	-81	-80	-74
-60	-67	-65	-65	-66	-63	-63	-64
-57	-67	-58	-65	-59	-54	-40	-40
-45	-36	-26	-23	-21	-17	-18	-13
-33	-20	-20	-4	-6	2	0	0
-21	-10	-3	6	9	14	13	9



After DCT

-376	-23	1	-2.5	-0.3	4	0.2	-2.6
-224	53	20	3.4	5	3	0.6	2.3
68	3.3	-14	-0.3	-2.8	-1.9	-4.7	-6.2
2.3	-8.9	-1.5	-3.8	-2.5	1.2	1.4	1.9
-8.4	1.2	1.9	3.3	-2.1	5	1.8	5.3
4.5	7.3	-7.4	1.9	1.3	-0.7	-1.5	-6
6.4	6.8	-3.2	-2.6	1.3	-2.1	1.7	1
-16	0.1	9	0.8	1.8	1.7	-1	1

# Cuantización

Se debe cuantificar la tabla de coeficientes que obtuvimos usando DCT. Esta es la verdadera parte con pérdidas del proceso.

En la tabla de coeficientes que obtuvimos a través de DCT, las celdas de la parte superior izquierda se refieren a la parte de baja frecuencia y las celdas de la parte inferior derecha se refieren a la parte de alta frecuencia.

Sabemos que la parte de alta frecuencia se puede eliminar sin mucha pérdida en el aspecto de la imagen.

# Cuantización

Así que ahora preparamos una tabla de cuantificación de 8x8. Esta tabla tendrá valores muy pequeños en la parte superior izquierda y valores muy altos en la parte inferior derecha.

Cada valor en la tabla de coeficientes se divide por el valor correspondiente en la tabla de cuantificación y se redondea al entero más cercano.

# Cuantización

Ahora, debido al divisor alto en la parte inferior derecha, los valores divididos aquí se vuelven cero, eliminando así los datos de alta frecuencia.

Esta tabla de cuantificación depende del codificador y, por lo tanto, la tabla se mantiene en el encabezado de la imagen para que la imagen pueda decodificarse más tarde.

Quantization Table

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



# Cuantización

Observe que en la tabla de salida cuantificada, todos los valores, excepto el bloque 3x3 de la parte superior izquierda, son todos ceros. Estos son los datos de alta frecuencia que eliminamos. El reclamo a la fama de JPEG es que con solo estos 9 valores podemos recuperar casi la misma imagen.

### Quantization Table

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



### After Quantization

[illegible]

# Codificación

Ahora tenemos la salida comprimida como una matriz 2D. También sabemos que muchos de ellos son ceros. Por lo tanto, encontraremos una mejor manera de almacenar la subimagen que almacenarla como una matriz 2D.

Guardaremos los valores en orden de zigzag.

Entonces los datos serán:

-24, -23, 19, 5, 4, 0, 0, 1, 0, 0, 0, 0, 1 seguido de 53 ceros.

### After Quantization

[illegible]



# Agregar el encabezado

Aunque los archivos JPEG/JFIF no poseen un encabezado definido formalmente, generalmente contienen lo siguiente:

- \* Marcador JPEG de inicio de imagen (SOI) ( 0xFFD8)
- \* Marcadores de aplicación
- \* Ancho en píxeles
- \* Alto en píxeles
- \* Número de componentes (p. ej., 3 para RGB)

# Ejemplo

Podemos ver el siguiente ejemplo de cada paso:

<https://cgjennings.ca/articles/jpeg-compression/>