

# Capítulo 12

## Imágenes

---

<b>12.1. Imagen en bits.</b>	<b>4</b>
Bits por color, enfoque conceptual.	4
Tamaño puro del archivo.	6
<b>12.2 Esquema de color.</b>	<b>4</b>
<b>12.3 Formatos de imágenes.</b>	<b>8</b>
Detallando la imagen en bits.	11
Atributos de una imagen.	15
Formato BMP.	15
Formato GIF.	15
Formato PNG.	15
Formato JPG.	16

Una imagen computacionalmente hablando o imagen digital son puntos de color que técnicamente se llaman píxeles y la imagen tiene una altura y ancho definidos.

Un **pixel** es la unidad mínima de información visual en una pantalla o imagen. Es un acrónimo de las palabras *picture element*.

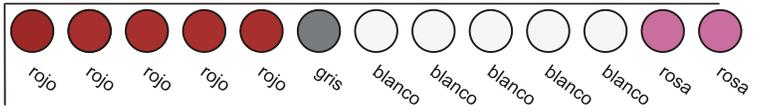
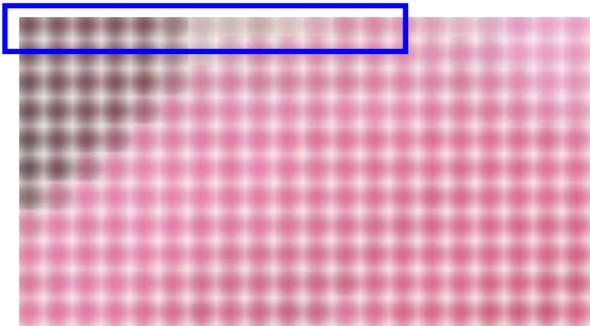
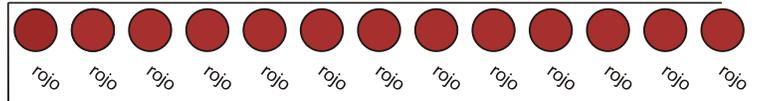
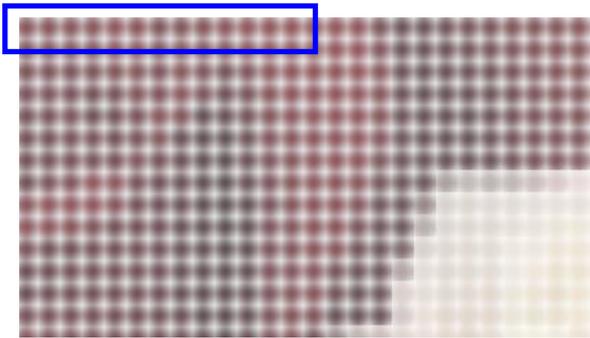
La imagen que se presenta a continuación es de 3,000 por 4,000 píxeles (enorme) y se presenta en una escala de 7% con 10cm. de alto, lo anterior para darse una idea del tamaño que tiene la imagen en píxeles con relación a centímetros.



Si se hace un zoom “píxeleando” una parte de la imagen quedaría.



Simulando los pixeles de la línea 1 y “estabilizando” a un sólo color de los usuales se tendrá.



## 12.1 Imagen en bits.

La clave para entender una imagen técnicamente es necesario convertirla a bits.

Un píxel en una imagen debe ser almacenado con todos sus atributos para poder guardarse en un archivo. Existen múltiples formas de almacenar una imagen, por esta razón existen múltiples formatos de archivos como JPG, BMP, PNG, GIF entre otros.

Uno de los atributos mas importantes a considerar en una imagen es el modelo, esquema o profundidad de color.

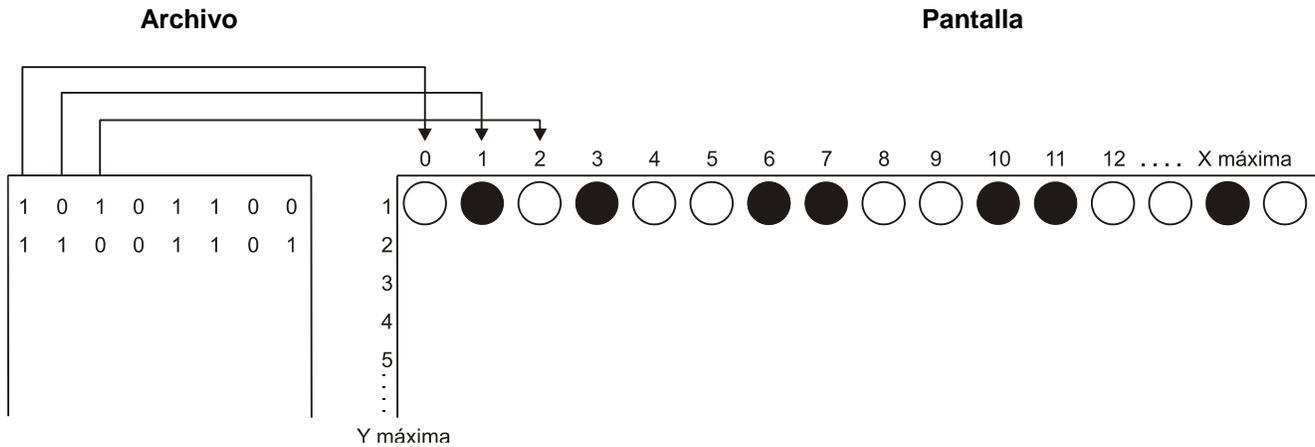
### Bits por color, enfoque conceptual.

El primer esquema de color tendría que se monocromático.

**Relación 1 bit x color.** En esta relación solamente es posible tener  $2^1 = 2$  colores, se supone una paleta de colores como.

Número	Color
0	Negro (fondo)
1	Blanco

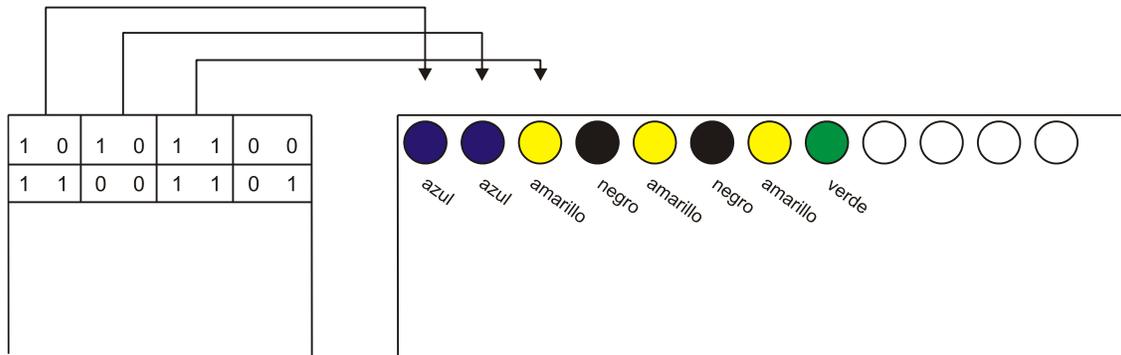
Si el archivo de la imagen contiene la siguiente información, se vería el siguiente despliegue.



**Relación 2 bits x color.** En esta relación se pueden manejar  $2^2 = 4$  colores, se supone una paleta de colores como.

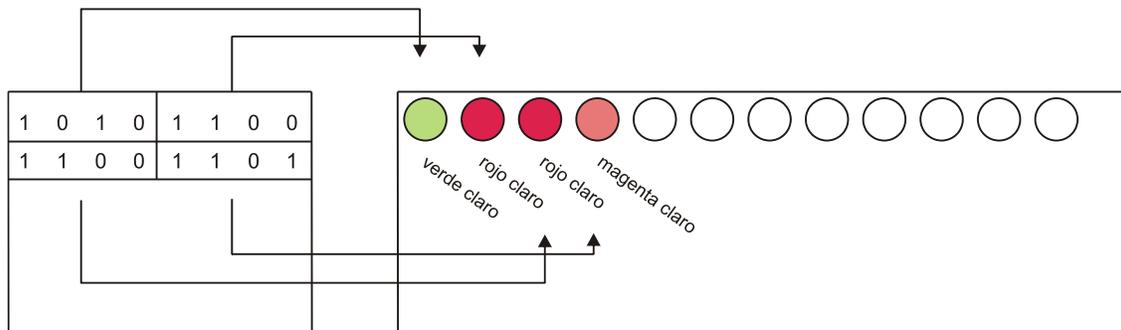
Número	Color
0	Negro (fondo)
1	Verde
2	Azul
3	Amarillo

Si la imagen contiene los mismos valores en bis del ejemplo anterior a hora la imagen tendría los siguientes puntos de color.

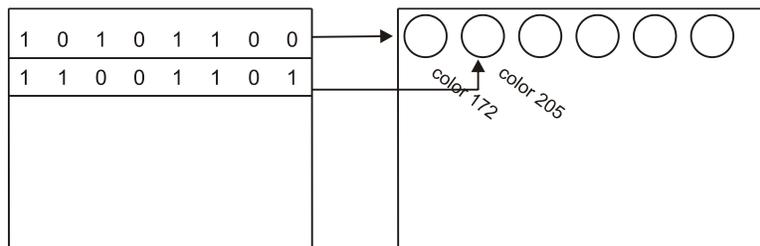


**Relación 4 bits x color.** En esta relación se pueden manejar  $2^4 = 16$  colores, se supone una paleta de colores como.

Número	Color	Número	Color	Número	Color	Número	Color
0	negro	4	rojo	8	gris oscuro	12	rojo claro
1	azul	5	magenta	9	azul claro	13	magenta claro
2	verde	6	marrón	10	verde claro	14	amarillo
3	violeta	7	gris claro	11	violeta claro	15	blanco



**Relación 8 bits x color.** En esta relación se pueden manejar  $2^8 = 256$  colores.



No tiene caso definir una paleta de colores, llega un momento que el manejo de la paleta de colores se complica porque son muchos colores, en la actualidad solamente el formato GIF trabaja con el esquema de 256 colores.

**15 bits, miles de colores.**

Posterior a los 256 colores, el color deja de ser un número y se convierte en una mezcla de tres tonos, uno para el color rojo, otro para el verde y otro para el color azul, este modelo de color se conoce como RGB por sus siglas en inglés Red, Green y Blue.

El siguiente esquema de color fue el de 15 bits,  $2^{15} = 32,768$  colores, 5 bits para el rojo, 5 para el verde y 5 para el azul, sobrando un bit.

**16 bits, miles de colores.**

El siguiente esquema de color fue el de 16 bits,  $2^{16} = 65,536$  colores, 5 bits para el rojo, 6 para el verde y 5 para el azul.

**24 bits, millones de colores.**

El siguiente esquema de color fue el de 24 bits,  $2^{24} = 16,777,216$  colores, 8 bits para el rojo, 8 para el verde y 8 para el azul.

Este esquema de color también es conocido como true color o color real y es el más utilizado actualmente.

Con este esquema de color se llega al límite humano, hay estudios que estiman que nuestra especie capta 1 millón de colores, otros 3 millones, otros estudios indican que 10 millones y como no es mi área de conocimiento mejor me quedo hasta aquí.

**32 bits, transparencia.**

El siguiente esquema de color fue el de 32 bits, el cual agrega un byte no para el color sino para agregar a la imagen pixeles que representen transparencia, es decir pixeles que no formen parte del "marco" de la imagen, lo que ha permitido que las imágenes actuales sean una obra de arte.

El formato PNG utiliza este esquema, razón que ha permitido que este formato sea muy popular.

**Tamaño puro del archivo.**

El precio a pagar por tener color y resolución (número de pixeles) en una imagen es la cantidad de memoria.

Supóngase una imagen de 4,000 x 3,000 o 12 Megapixeles el cual es de los más comunes por ahora.

Tabla.Equivalencias de memoria para una misma imagen.

Esquema	Tamaño (bytes).	Tamaño en kilobytes y megabytes
1 bit por pixel, monocromático.	1,500,000.  12,000,000 / 8 bits por unidad de memoria.	1,500 kB. 1.5 mB.
2 bits, 4 colores.	3,000,000.  12,000,000 / 4	3,000 KB. 3 mB.
4 bits, 16 colores.	6,000,000.  12,000,000 / 2	6,000 kB. 6 mB.

8 bits, 256 colores.	12,000,000. 12,000,000 / 1	12 mB.
16 bits, miles de colores.	24,000,000. 12,000,000 x 2 bytes.	24 mB.
24 bits, millones de colores.	36,000,000. 12,000,000 x 3 bytes.	36 mB.
32 bits, transparencias.	48,000,000. 12,000,000 x 4 bytes.	48 mB.

La misma imagen de 3000x4000px en un modesto esquema monocromático ocuparía 1.5 mB comparado con 36 mB para un esquema de millones de colores.

Nota. B mayúscula es para Bytes, b minúscula es para bits.

Nota. Se indica en el título tamaño puro del archivo porque los formatos de imagen realizan una compresión, lo que hace que dependiendo del formato de almacenamiento de imagen el archivo sea más grande o más pequeño.

## 12.2 Esquema de color.

El color deja de ser un número y se convierte en una mezcla de colores primarios, dependiendo del esquema de color es como se almacenará la imagen en un archivo.

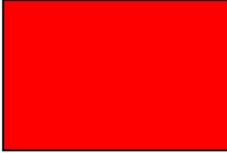
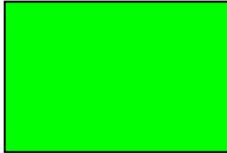
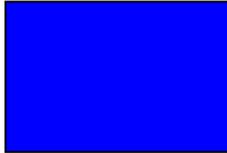
### Esquema RGB.

Es el esquema de color más común por ser el de las pantallas de computadora.

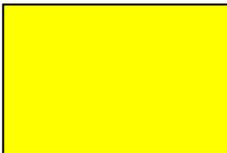
**Los colores primarios** son como su nombre lo indica, R = red = rojo, G = green = verde, B = blue = azul.

Entonces para generar un color se necesitan tres datos, dependiendo del esquema de bit por cada color primario se forma un nuevo color, es algo como mágico. Primero siempre será rojo, segundo verde y tercero azul.

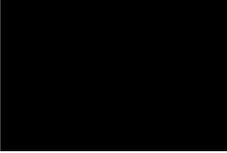
Los colores primarios y el tono que se obtiene para un esquema de 8 bits x color primario, que es el más común actualmente, quedan.

Rojo. (255, 0, 0)	Verde. (0, 255, 0)	Azul. (0, 0, 255)
		
Este es el rojo "más rojo" que hay en este esquema.	Este es el verde "más verde" que hay en este esquema.	Este es el azul "más azul" que hay en este esquema.

**Colores secundarios.** Se obtienen a partir de la combinación de dos colores primarios.

Amarillo. (255, 255, 0)	Magenta (anglicismo) o morado. (255, 0, 255)	Cyan (Anglicismo) o celeste. (0, 255, 255)
		
Este es el amarillo "más amarillo" que hay en este esquema.		

**Colores de espectro.** El color mínimo es el negro o también conocido como ausencia de color y el blanco es la unión de todos los colores.

Negro. (0, 0, 0)	Blanco. (255, 255, 255)	Gris. (100, 100, 100)
		

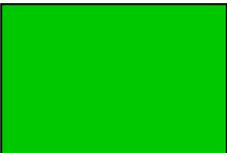
Para generar gris los tonos deben ser los mismos valores para rojo, verde y azul.

Técnicamente hablando el color negro y el blanco son grises, el negro es el gris más intenso y el blanco es el gris menos intenso o tenue.

**Colores nombre común versus combinación RGB.**

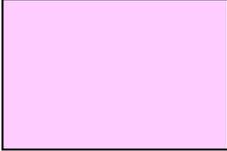
Naranja. (255, 153, 50)	Café (153, 51, 0)	Rosa. (255, 204, 255)
<div style="text-align: center;"></div> <p data-bbox="147 1108 570 1171">El naranja se obtiene con mucho rojo, la mitad de verde y poco azul.</p>		

**Algunos verdes.**

Verde. (0, 200, 0)	Verde (50, 100, 0)	Verde. (50, 150, 50)
		

### Notación hexadecimal.

La notación hexadecimal es más estable para el manejo de color debido a que solo en dos dígitos se pueden representar todas las combinaciones.

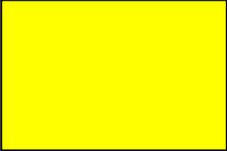
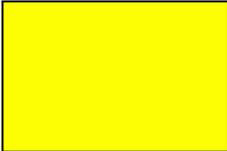
<b>Naranja. (255, 153, 50)</b> <b>Naranja. (#FF, #99, #32)</b>	<b>Café (153, 51, 0)</b> <b>Café (#99, #33, #00)</b>	<b>Rosa. (255, 204, 255)</b> <b>Rosa. (#FF, #CC, #FF)</b>
		

### True color, Color real o millones de colores.

Mediante este esquema de 1 byte por tono se pueden obtener técnicamente 16.5 millones de colores.

$$255 \text{ (rojos)} \times 255 \text{ (verdes)} \times 255 \text{ (azules)} = 16,581,375 \text{ colores.}$$

Por qué se dice técnicamente, bueno porque en realidad aquí se presentan 3 colores técnicamente diferentes por la combinación de los tonos pero en la práctica los tres son amarillos con diferencias imperceptibles para el ojo humano.

<b>Amarillo. (255, 255, 0)</b>	<b>Amarillo. (255, 255, 10)</b>	<b>Amarillo. (253, 254, 4)</b>
		

Se concluye entonces que una imagen son pixeles de los cuales cada uno de ellos para almacenarse necesita de 3 bytes. Lo anterior nos da la pauta para determinar el tamaño del archivo.

## 12.3 Formatos de imágenes.

A continuación se presenta una misma imagen almacenada en diferentes formatos de archivos.

Las cuatro imágenes tienen una resolución de 4000x3000 píxeles y el esquema de color a millones de colores RGB, la imagen fue adquirida a través de una cámara de teléfono celular en formato JPG y partir de ésta se convirtió a formatos BMP, GIF, PNG.

La imagen fue reducida para su visualización a un tamaño en ancho de 7cm.



Formato BMP.  
Tamaño del archivo: 36,578,360 bytes.



Formato JPG.  
Tamaño del archivo: 1,918,842 bytes.



Formato PNG.  
Tamaño del archivo: 7,976,871 bytes.



Formato GIF.  
Tamaño del archivo: 1,498,066 bytes.  
Nota. El esquema de color es de 256 por eso la imagen tiene un aspecto diferente, el cambio no obedece al formato.

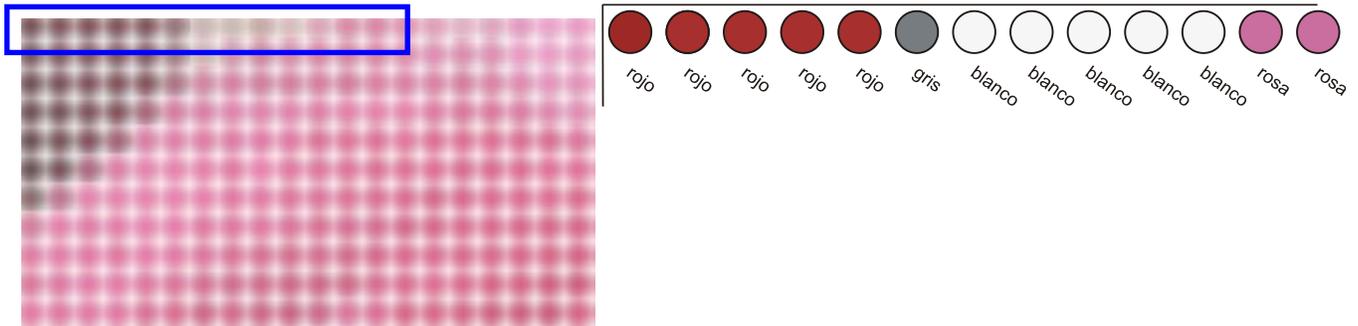
Para la vista todas las imágenes son lo mismo, el formato no da la calidad de la imagen, la calidad de la imagen la da la resolución, es decir el alto y el ancho y el esquema de color, la imagen es la imagen, entonces porqué hay tantos formatos de imagen.

El formato mínimo para almacenar una imagen en un archivo tiene que ser guardar el ancho, el alto y los tonos de colores para cada pixel, un byte para cada tono RGB en un esquema de color real o millones de colores que es el más usual.

Imagen JPG de tamaño real. Se aclara que no es suficiente el espacio para presentarla por completo.



## Detallando la imagen en bits.



Retomando el segmento de imagen descrito anteriormente y considerando los primeros 13 píxeles se tendría en bits.

Imagen en número decimales.

11, 20, 157,41,39, 157,41,39, 157,41,39, 157,41,39, 157,41,39, 120,120,120, 245,245,245, 245,245,245, 245,245,245, 245,245,245, 212,116,180, 212,116,180, ...

Imagen en números con alto, ancho y lista de colores en RGB.

11 (alto), 20 (ancho), 157 (tono **R**ed del primer pixel), 41 (tono **G**reen del primer pixel), 39 (tono **B**lue del primer pixel), el cual es azul, [ 157,41,39 (segundo pixel, rojo) ], [ 157,41,39 (tercer pixel, rojo) ], [ 157,41,39 (cuarto pixel, rojo) ], [ 157,41,39 (quinto pixel, rojo) ], [ 120,120,120 (sexto pixel, gris) ], [ 245,245,245 (séptimo pixel, blanco) ], [ 245,245,245 (octavo pixel, blanco) ], [ 245,245,245 (noveno pixel, blanco) ], [ 245,245,245 (décimo pixel, blanco) ], [ 245,245,245 (décimo primero pixel, blanco) ], [ 212,116,180 (décimo segundo pixel, rosa) ], [ 212,116,180 (décimo tercero pixel, rosa) ],

Imagen en bits, formato crudo o extensión raw.

00001011 (alto), 00010100 (ancho), 10011101 00101001 00100111 (rojo) 10011101 00101001 00100111 (rojo) 10011101 00101001 00100111 (rojo) 10011101 00101001 00100111 (rojo) 01111000 01111000 01111000 (gris) 11110101 11110101 11110101 (blanco) 11110101 11110101 11110101 (blanco) 11110101 11110101 11110101 (blanco) 11110101 11110101 11110101 (blanco) 11010100 01110100 10110100 (rosa) 11010100 01110100 10110100 (rosa)

Los primeros dos bytes para valores de tipo *unsigned int* que permitirían guardar imágenes de hasta 65,536 x 65,536 píxeles de ancho y alto, si nos quedamos con un byte solo se podrían tener imágenes de hasta 256 x 256 píxeles.

Al formato anterior se le conoce como formato crudo o raw en inglés, el tamaño de un archivo con este esquema es de  $11 \times 20 \times 3 + 1 + 1 = 222$  bytes, cada pixel de la imagen tiene que multiplicarse por 3 para cada uno de los tonos de color RGB, por eso se multiplica 11 píxeles de alto x 20 píxeles de ancho x 3 bytes uno por cada tono RGB y se suma 1 byte para el alto y un byte para el ancho.

## Atributos de una imagen.

Debido a que no es suficiente con guardar solo los pixeles de la imagen es necesario definir un protocolo para guardar toda la información relativa a la imagen como.

- Alto.
- Ancho.
- Esquema de color en bits. 1 (monocromático), 2, 4, 8 (256 colores), 15, 16 (miles de colores), 24 (millones de colores), 32 (transparencia).
- Método de compresión. RLE (run length encoding), familia LZ (Lempel-Ziv, apellidos de sus creadores) y Huffman (apellido de su creador).
- Número de planos.
- Resolución horizontal.
- Resolución vertical.
- Paleta de colores.

De la forma en como se implemente esta información surgen los formatos de imágenes, los más populares son BMP, JPG, PNG y GIF.

### Formato BMP.

Bit Mapped Picture es un formato fue inventado por Microsoft y es de los más sencillo técnicamente hablando, en la mayoría de las veces las imágenes BMP no tienen compresión, esto hace que el tamaño de los archivos BMP sean muy grandes, este formato es popular debido a que viene por “default” con los productos Microsoft.

El formato BMP permite compresión RLE, sin embargo por lo regular las imágenes no están comprimidas.

### Formato GIF.

El formato fue creado por la compañía CompuServe en 1987 su algoritmo de compresión es LZW, más eficiente que el RLE, la popularidad de este formato es que permite trabajar animación, soporta varias imágenes y las presenta como una animación, la “desventaja” es que solamente maneja un modelo de color de 8 bits, es decir solamente maneja 256 colores.

### Formato PNG.

PNG (Portable Network Graphics) es un formato muy popular debido a que permite el manejo de transparencias por eso su modelo de color es de 32 bits o 4 bytes, tres para los tonos RGB y el cuarto para el manejo de la transparencia, además aprovecha los 8 bits de la transparencia para el manejo de semitransparencias u opacidades, utiliza un método de compresión llamado deflación que es una mezcla del algoritmo LZ y Huffman que es bastante eficiente, comprime mejor que el LZW pero no como el del formato JPG.

El manejo de transparencias permite superponer imágenes, recursos muy útil en la creación de imágenes compuestas ampliamente utilizadas en carteles publicitarios.

## Formato JPG.

Joint Photographic Experts Group fue creado por un comité de expertos con el objetivo de procesar imágenes digitales para fotografía, su popularidad se debe a que su método de compresión es muy bueno llega a tasas de compresión del 90% lo cual lo hace ideal para compartir en medios digitales y fotografía amateur.

Su método de compresión está basado en una serie de pasos, el primero es aprovechar dos características de percepción del ojo humano la luminancia y la crominancia, es decir percepción de luz y color. Para lograrlo el algoritmo de compresión convierte el modelo RGB a YUV o YCbCr, Y es el factor de luminancia y Cb y Cr es la crominancia, posteriormente se aplica un submuestreo donde se aplica un coeficiente que hace que la conversión no tenga o tenga un factor de pérdida, es decir que los datos originales no se puedan reconstruir una vez hecha esta conversión, posteriormente se aplica el método de compresión llamado codificación entrópica y finalmente se aplica el método de compresión de Huffman.

Todos los formatos tratados hasta aquí son basados en bits existen otros tipos de imágenes llamadas vectoriales las cuales su principio es que la imagen se compone de figuras geométricas básicas, estos formatos tienen ventajas sobresalientes que permiten una edición minuciosa de la imagen lo cual es muy útil para los diseñadores profesionales. Los formatos vectoriales no serán tratados en este material.