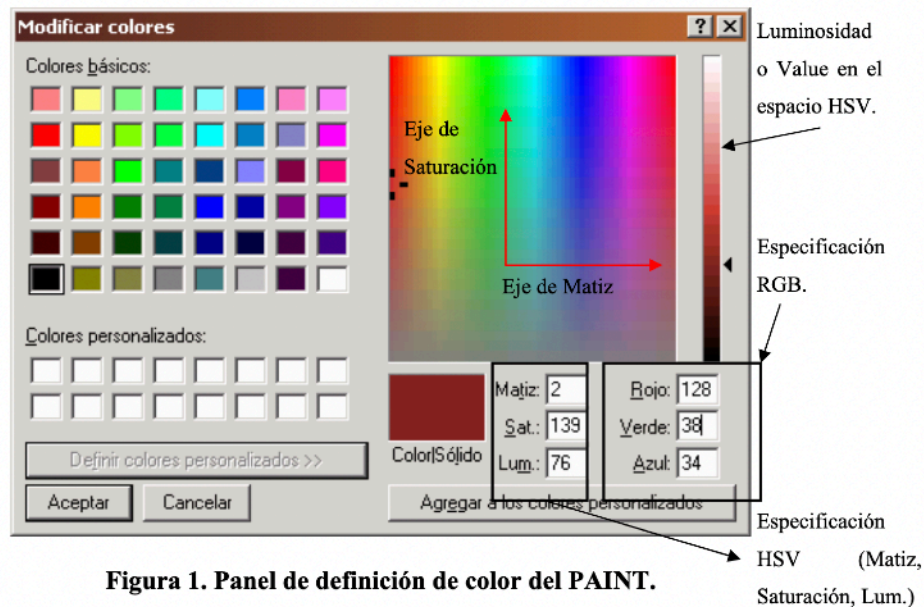


Espacio de Colores, representación y almacenamiento

RGB

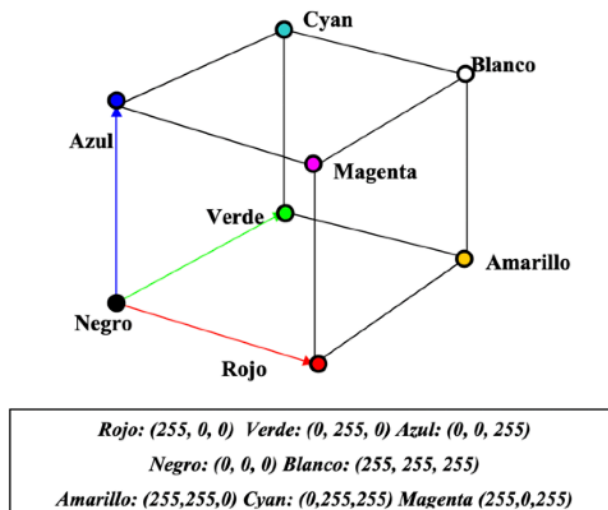


Los colores que observamos pueden construirse a partir de la mezcla de los colores denominados primarios rojo (red), verde (green) y azul (blue), cada uno con diferente intensidad. La intensidad del color primario establece una medida relativa de la cantidad de ese color producida en el dispositivo donde se visualiza la imagen. Por ejemplo, una intensidad de 0 en el canal primario rojo indica una cantidad 0 de color rojo. La intensidad máxima de color para cada canal depende del programa utilizado para generar el color y el dispositivo donde se va a representar, mientras que 255 indica la máxima cantidad de color primario que el dispositivo puede producir.

En teoría sí la intensidad del color en los tres canales rojo, verde y azul es 0, el color producido es negro, mientras que si esta intensidad es 255 para los tres canales el color es blanco. En la práctica existen pequeñas diferencias entre los posibles dispositivos.

El resto de colores se construyen usando diferentes intensidades para los canales RGB como se aprecia en la Figura 1. En el lado inferior derecho, vemos que se puede definir la cantidad de cada color Rojo, Verde o Azul para producir un nuevo color. En el mismo panel tenemos otras posibilidades para definir el color, como la definición en el espacio de colores Matiz, Saturación y Luminosidad.

Para entender el espacio de colores RGB es de especial utilidad lo que se denomina el cubo de colores. En el cubo de colores, cada eje es un color primario y cada punto dentro del cubo representa un color. Este cubo se ve en la figura, mostrando diferentes tripletas (R, G, B) y el color que producen.



El origen del cubo (0,0,0) representa la total ausencia de color (negro). La esquina opuesta representa la suma de las mayores intensidades de rojo, verde y azul, dando lugar al blanco. Las esquinas Rojo, Verde y Azul son los colores primarios y el resto de esquinas son los colores secundarios (Cyan, Amrillo y Magenta). La diagonal principal del cubo, la línea que va desde el negro al blanco, representa todos los puntos con igual cantidad de rojo, verde y azul que son los grises desde el más oscuro (negro) hasta el más claro (blanco). A este eje se le denomina eje neutro, ya que en los grises no predomina ningún matiz

HSV

En la Figura 1 vemos que también es posible obtener colores rellenando unos campos denominados matiz, saturación y luminosidad o en inglés hue, saturation and value.

El matiz de un color es su localización dentro del espectro de colores: las palabras rojo o verde se refieren al matiz de dos colores. Por otra parte, la saturación representa la pureza del matiz respecto al blanco. Un color verde totalmente saturado lo podemos imaginar como una pintura verde que no ha sido mezclada con pintura blanca para rebajarla. Si se mezcla con pintura blanca el verde aparece más pastel y ya no está saturado.

Finalmente un color tiene brillo o luminosidad y representa la cantidad de luz que el color refleja. Menos luz reflejado implica que percibimos el color de forma más oscura. Una medida del brillo, aunque no es la única, es el denominado Value. En la Figura 1, la barra situada más a la izquierda indica las diferentes luminosidades del color.

A modo de curiosidad, la relación entre el brillo y el espacio de colores RGB es la siguiente. Dos colores cuya suma de contenidos $R + G + B$ es la misma tienen el mismo brillo. El color que más brillo tiene es el blanco $R=255$ $G=255$ y $B=255$, no existiendo otro color con tanto brillo. Si nos movemos por el eje neutro del cubo de color RGB hacia el negro en planos perpendiculares tenemos los diferentes brillos en relación al color blanco. A la mitad del eje, tenemos el 50% de brillo en el plano perpendicular (la suma de las componentes $R + G + B$ suman la mitad que para el blanco). El color con el menor brillo es el negro $R + G + B=0$ y no hay otro color con menos brillo. Hay varias medidas para el brillo en GIMP se usan las siguientes:

Cantidad de luz: $L = [\max(R,G,B) + \min(R,G,B)]/2$

Valor (value) $V = \max(R,G,B)$

Luminancia $Y = 0.30 * R + 0.59 * G + 0.11 * B$

La luminancia es una medida del brillo que coincide bastante bien con la percepción humana del brillo.

La saturación también está relacionada con la representación del color en el espacio de colores RGB. El eje neutro en el cubo contiene tonos de grises y estos para nosotros se caracterizan por la ausencia de color (matiz).

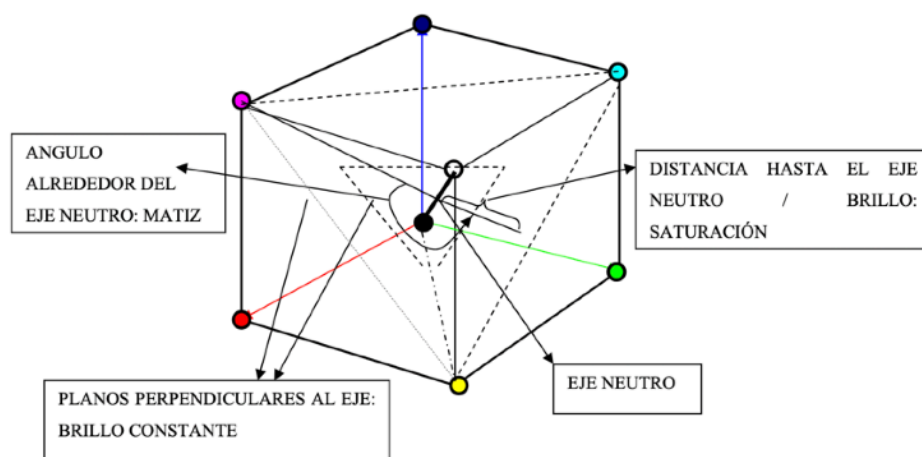
La cantidad de color de un punto en el cubo de colores RGB podemos medirla como la distancia desde el punto hasta el eje neutro. Cuanto más cercanos nos encontremos de este eje, menos cantidad de color apreciamos. La saturación es el cociente entre la cantidad de color y el brillo. De esta forma si nos movemos en un plano perpendicular al eje neutro (brillo constante), la saturación aumenta a medida que nos alejamos por el plano perpendicular de este eje.

El concepto de matiz está relacionado con lo que comúnmente entendemos por color. El matiz de un punto dentro del cubo de color RGB se define como la posición angular respecto del eje neutro. Todos los puntos que mantienen el mismo ángulo respecto del eje neutro tienen el mismo matiz.

El matiz del punto Amarillo (255,255,0) es compartido por todos los puntos situados en el plano que forma un ángulo respecto al eje neutro igual al ángulo que le corresponde el punto (255,255,0).

Como resumen y posibles reglas a recordar, podemos observar que en el cubo de color RGB, los puntos Cyan, Magenta y Amarillo son más brillantes que los puntos Rojo, Verde y Azul, ya que los primeros se encuentran en un plano perpendicular al eje neutro más cercano al color blanco que los últimos. Lo mismo cabe decir de los puntos situados en la pirámide formada por los puntos Cyan, Magenta, Amarillo y Blanco y los puntos situados en la pirámide formada por los puntos Rojo, Verde, Azul y Negro.

Además la cercanía del punto al eje neutro incrementa el aspecto pastel o acuso del color ya que disminuye la saturación.



El proceso sufrido por una imagen desde que es capturada hasta que es mostrada en un dispositivo supone un conjunto de transformaciones que finalmente resultan en que la imagen representada en el dispositivo es diferente de la imagen original. Esto ocurre independientemente del tipo de imagen y formato.

Los colores seguros del navegador

Si bien los programas de diseño o la especificación HTML nos permiten usar una gran cantidad de colores (normalmente codificados con 24 bits de profundidad), aplicaciones como los navegadores web ofrecen limitaciones. La limitación software impuesta se basa normalmente en las características de la tarjeta de video. Esto quiere decir, que los navegadores no muestran toda la variedad de colores que puedan componer nuestro diseño, sino una fracción de ellos realizando las conversiones adecuadas.

En este sentido surgen dos problemas: ¿cómo podemos estar seguros de que nuestra página web será mostrada tal y como la hemos diseñado?, y ¿será mostrada del mismo modo en todos los navegadores?.

La clave está en la denominada paleta segura. La paleta segura es un conjunto de colores establecidos y compartidos por la mayoría de los navegadores. Esta paleta consta de 256 colores. Estos colores se codifican según sus componentes RGB, representándose su valor de intensidad.

Los formatos “mapa de bits” o raster-only formats.

La resolución.

Las representaciones mapa de bits se basan en establecer una cuadrícula sobre la imagen, de forma que cada rectángulo de la cuadrícula contiene la información sobre el color predominante en esa región de la imagen. Cada rectángulo de esta retícula se denomina píxel.

El número de píxeles por unidad de área o longitud constituye lo que se denomina resolución de la imagen. Por ejemplo, en un monitor la resolución se mide en píxeles o puntos por pulgada (ppi pixels per inch). La resolución del monitor depende de las características técnicas del mismo. La mayoría de monitores nuevos tienen una resolución de 96 píxeles por pulgada. La representación de mapa de bits de una imagen es traducida directamente al monitor de forma que cada píxel en el monitor corresponde con un píxel de la imagen. Esto tiene como consecuencia que si la resolución del monitor es mayor que la resolución de la imagen original, la imagen aparecerá en un tamaño menor en el monitor (en una pulgada caben más píxeles en el monitor que en la imagen original).

Un ejemplo: si se quiere mostrar una imagen de 1 pulgada por 1 pulgada con una resolución de 144 ppi sobre un monitor de 72 puntos por pulgada (72-dpi), aparecerá sobre el monitor con unas dimensiones de 2 pulgadas por 2 pulgadas, ya que para reunir 144 píxeles donde mostrar la imagen hacen falta 2 pulgadas por 2 pulgadas del monitor.

Con las impresoras pasa lo mismo. La mayoría de las impresoras láser de sobremesa tienen una resolución de 600 dpi (600 puntos por pulgada). Por el contrario las impresoras de chorro de tinta tienen una resolución de 300 a 600 dpi. Este tipo de información hay que considerarla a la hora de deducir el tamaño de la imagen y la calidad resultante cuando se quiere representar una imagen.

La representación de la imagen en un mapa de bits tendrá mayor calidad mientras mayor sea la resolución (para una misma área un mayor número de píxeles implica un tamaño menor del píxel, con lo que se representará con un mejor nivel de detalle la imagen).

En la Figura 2 podemos apreciar la diferencia en el uso de dos resoluciones diferentes sobre una imagen. En la figura superior se puede apreciar el efecto de cuadrícula típico que surge cuando la resolución es demasiado baja.

El precio a pagar por una mayor resolución es la necesidad de una mayor capacidad de almacenamiento. Para tomar una decisión sobre la resolución con la que una imagen debe representarse hay que tener en muy en cuenta entre otros factores la capacidad del dispositivo donde se va a mostrar la imagen.

Por ejemplo si la resolución y tamaño de nuestro monitor es tal que dispone de 600 píxeles de alto por 800 de ancho, y queremos representar al completo una imagen en él no podremos usar una imagen con 900 píxeles de alto por 1300 de ancho ya que no cabrá en la pantalla a menos que redimensionemos la imagen para adaptarla al dispositivo.

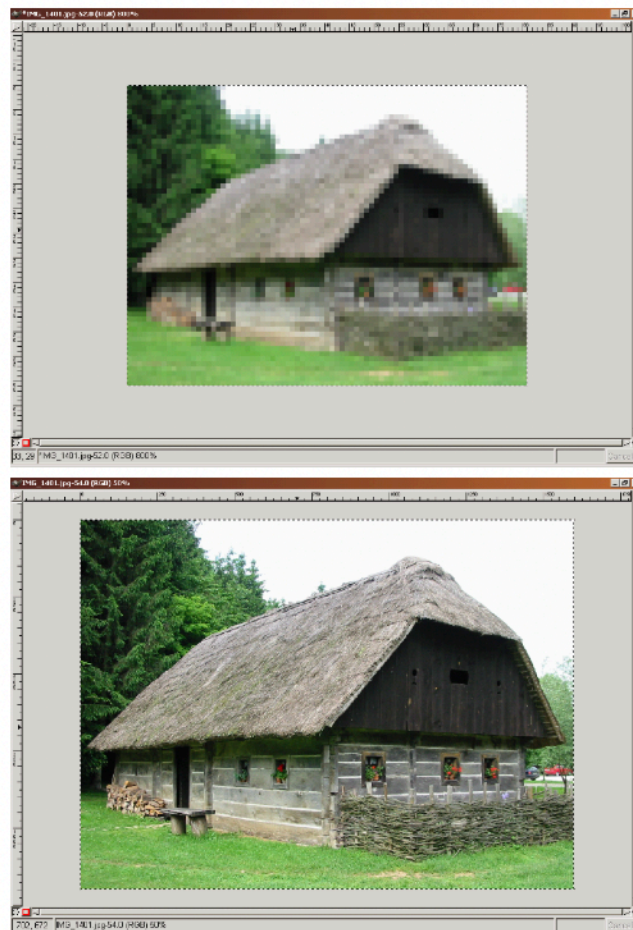


Figura 2. Dos imágenes con diferente resolución. La superior tiene mucha menos resolución y ha sido ampliada a un tamaño similar a la original: obsérvese la pérdida de detalle y la peor calidad.

La representación del color

Como se ha mencionada más arriba cada píxel de la imagen debe almacenar información acerca del color. Tal como se discutió, el color se puede representar por un conjunto de parámetros en diferentes espacios de colores. Por ejemplo en el espacio de color RGB necesitaremos tres parámetros para indicar las intensidades del rojo, el verde y el azul. Estos tres parámetros deben ser almacenados en cada píxel.

En este punto debemos profundizar un poco en la manera en la que los números son almacenados en un ordenador. El elemento básico es el bit. Un bit puede representar una elección entre dos opciones, por ejemplo blanco o negro. Si la imagen que tenemos que almacenar fuese en blanco y negro cada píxel podría contener un solo bit para indicar si el color asociado al píxel es blanco o negro. De esta forma, una imagen de 10 por 10 pulgadas con una resolución de 96 ppi tiene 9600 píxeles. Como cada píxel para una imagen en blanco y negro requiere 1 bit, harían falta en general 9600 bits.

Sin embargo hacen falta más de dos colores para representar con cierta calidad una imagen. Para ello usaremos más bits. Por ejemplo, con dos bits podemos representar cuatro. En general n bits permiten 2^n opciones que para nosotros son posibles colores diferentes.

El término profundidad de color hace referencia precisamente a esto. Una imagen representada con una profundidad de color (color depth) de 24 bits quiere decir que cada píxel representa un color de entre 2^{24} (posibles). Este sería por ejemplo el caso de usar el espacio de colores RGB y guardar cada posible intensidad del rojo, el verde y el azul en 8 bits.

Evidentemente el uso de más colores amplía los requerimientos de memoria. Para representar la misma imagen del ejemplo descrito para dos colores hacen falta 9600 píxeles \times 24 bits = 28.125 Kbs, es decir 24 veces más, como es lógico. Sin embargo como veremos más abajo la cantidad de memoria necesaria para almacenar la imagen depende del formato en que ésta se almacene, por lo que estas medidas sólo son válidas si la imagen se almacena píxel a píxel con su representación de color asociada.

Los formatos de imagen del tipo mapa de bits. A nadie se le escapa de lo dicho anteriormente que pueden existir mil formas de representar y ordenar esta información asociada a los píxeles. Atendiendo a las necesidades técnicas y a los recursos disponibles han ido apareciendo a lo largo de los últimos años una gran cantidad de estrategias para almacenar esta información. La lista es extensa y los detalles de cada uno de los formatos abundantes.

Windows Bitmap (BMP)

Cada fichero contiene cuatro partes: un encabezado de fichero bitmap, un encabezado de información bitmap, una tabla de color y un conjunto de bytes que establece los bits incluidos en el bitmap.

El encabezado de fichero contiene información sobre las características del propio fichero, en concreto tipo, tamaño y distribución.

El encabezado de información bitmap contiene información sobre la codificación de la imagen en el archivo: dimensiones de la imagen, tipo de algoritmo de compresión y formato de color del bitmap.

La tabla de color es una característica muy importante dentro de este formato. Podemos imaginarnos esta tabla como un casillero donde cada casilla contiene 24 bits para representar un color. Con este casillero en lugar de utilizar 24 bits para hacer referencia al color asociado a un píxel podemos referenciar el número de casilla dentro del casillero. Supongamos que tenemos un bitmap con sólo un bit por píxel. En este caso sólo podemos hacer referencia a dos casillas, pero cada casilla puede tener un color de 24 bits. Esto supone un ahorro en muchos casos.

Por ejemplo: en una imagen de 10 por 10 pulgadas con resolución 96 ppi, y un solo bit por píxel podemos codificar haciendo uso de la tabla de color dos colores de 24 bits en cada píxel con: 9600 bits en el mapa de bits y 48 bits más para los dos colores en la tabla de color es decir 9648 bits. Si codificáramos asociando 24 bits a cada píxel necesitaríamos 230400 bits.

El concepto de tabla de color que también se denomina paleta de colores se utiliza en muchos formatos y programas de dibujo. Otra gran ventaja es que podemos cambiar de una vez todos los colores de la imagen sin más que modificar los colores en la tabla de color.

Dentro de la cabecera de información sobre el bitmap se indica el número de bits dedicados a cada píxel y el número máximo de colores en el bitmap. Hay diferentes posibilidades: 1 bit por píxel o dos colores (monocromo), 4 bits por píxel o 16 colores, 8 bits por píxel o 256 colores o 24 bits por píxel. En este último caso no se hace uso de la tabla de color sino que se codifica directamente en formato RGB el color asociada a cada píxel.

Otra característica importante del formato Windows bitmap es que utiliza un tipo de codificación que se puede encuadrar en los formatos de compresión sin pérdidas. La compresión es un concepto que se aplica en informática en el sentido de evitar el uso de información redundante o poco importante para la aplicación con la finalidad de ahorrar espacio en memoria y/o tiempos de comunicación.

Veamos un sencillo ejemplo.

Supongamos la siguiente secuencia de números:

1 3 3 3 3 5 5 5 5 5 2 2 2 2 2 2 2 2 2 2 3 3 3 3 4 4 4 4 4 4 4 4 4 4 4 4 4

A la hora de almacenar esta secuencia podemos optar por almacenar cada número por otra estrategia que reduzca las necesidades de almacenamiento. Podemos poner para cada número su valor y el número de veces que aparece.

En el ejemplo propuesto, la secuencia se codifica como:

(1,1) (3,5) (5,5) (2,11) (3,4) (4,16).

Se aprecia que el número de elementos de la secuencia codificada es menor. Esta es la versión más primitiva de la técnica conocida en inglés como Run Length Encoding. En el caso de las imágenes esta técnica es beneficiosa cuando aparecen grandes zonas con el mismo color: en lugar de almacenar repetitivamente para cada píxel el mismo color, se cuentan las repeticiones sucesivas del color y se almacena este número junto a la codificación del color. La técnica se denomina compresión sin pérdidas porque se elimina información redundante pero se puede decodificar la secuencia de datos original exactamente a partir de su versión codificada. El formato Windows bitmap, codifica de esta forma aquellas imágenes con 4 bits por píxel y 8 bits por píxel (16 y 256 colores).

Los resultados suelen ser mejores cuando la imagen proviene de dibujos para ordenador en lugar de imágenes reales, ya que en el caso de los dibujos la variedad de cambios en los colores es menor y suelen aparecer grandes superficies con el mismo color. Para las imágenes reales este tipo de codificación no es el mejor, por lo que las fotografías codificadas en Windows bitmap suelen ocupar una gran cantidad de memoria.

Graphics Interchange Format (GIF)

El formato de intercambio de gráficos o GIF está especialmente diseñado para la transmisión en línea de datos gráficos tipo mapa de bits entre ordenadores, siendo uno de los formatos gráficos más usados en internet. De hecho es un estándar MIME nativo para los navegadores. Fue desarrollado por la empresa CompuServe a finales de los años 80. El formato GIF utiliza el algoritmo de compresión LZW (Lempel-Ziv y Welch) de la empresa UNISYS.

Un conjunto de datos GIF, denominado GIF data stream puede contener uno o más imágenes con diferentes parámetros. Este hecho es aprovechado en el diseño de páginas web para incluir animaciones simples o secuencias de gráficos.

El decodificador del archivo GIF es un programa o parte de un programa que se encarga de leer paso a paso el GIF data stream e ir interpretándolo para producir los gráficos en el dispositivo de destino. A medida que va leyendo el GIF data stream puede ir encontrando diferentes gráficos e ir mostrándolos uno detrás de otro.

El formato GIF también se basa en el uso de tablas de color al igual que el Windows bitmap. De hecho el máximo número de colores que se puede emplear en un GIF es 256, por lo que el uso de color indexado es importante. Los archivos GIF pueden tener varias tablas de color locales (una para cada gráfico incluido en el archivo) y varias tablas de color globales usadas para aquellos gráficos que no dispongan de una tabla de color local. Las tablas de color también van usándose secuencialmente por parte del decodificador. Por ejemplo, el decodificador encuentra una tabla de color global y a continuación aparece un gráfico con una tabla de color local. El decodificador salva la tabla global y usa la tabla local para producir el gráfico. Continúa la decodificación y encuentra un gráfico sin una tabla de color local. Entonces, restaura la tabla global y con ella representa el gráfico. La lectura continúa y encuentra una tabla de color global nueva, que sustituye a la existente.

El formato GIF tiene además otras particularidades interesantes, sobre todo desde el punto de vista de la decodificación. Se pueden introducir campos denominados Delay Times (tiempos de retraso) que indican que el decodificador debe esperar un tiempo antes de continuar decodificación. Esto se usa para controlar el ritmo de una animación.

Otra característica interesante es el uso del denominado Transparency Index. Este índice indica píxeles que el decodificador no debe modificar en el dispositivo de presentación del gráfico. De esta manera se consigue un efecto de transparencia entre el fichero procesado y el fondo sobre el que se está mostrando el gráfico. El transparency index actúa realmente como un cuarto canal adicional a los tres típicos R, G y B.

Es muy importante recordar que el formato GIF usa una variante del algoritmo de compresión sin pérdidas (ver explicación en Windows BITMAP) denominado LZW, mucho más eficiente que el mencionado RLE para el Windows Bitmap. En cualquier caso se trata

también de un algoritmo sin pérdidas con lo que la imagen originalmente codificada se obtiene exactamente igual tras el proceso de decodificación.

Otro aspecto reseñable es la posibilidad de usar GIF entrelazado. Los GIF entrelazados los reconocemos cuando descargamos una página web, ya que la imagen aparece inicialmente con poca calidad y posteriormente se va perfilando. El entrelazado no afecta a la velocidad o al tamaño del GIF.

JPEG

JPEG es un algoritmo de compresión con pérdida de calidad, es decir parte de la información de la imagen es eliminada. JPEG se especializa en eliminar justo aquella información que es menos relevante en la percepción de la imagen con cierta calidad.

Esto es una gran diferencia con el formato GIF donde no hay pérdida de información.

Otra característica es que se puede ajustar el nivel de compresión en JPEG, evidentemente a costa de aceptar una pérdida de calidad. Esta es una de las razones de su popularidad en la web: los diseñadores pueden diseñar optimizando para calidad en los gráficos o reducción del ancho de banda consumido.

JPEG trabaja siempre con 24 bits por píxel (8 bits por canal), por lo que todo lo relacionado con paleta de colores e indexación no se utiliza. Esta característica proviene del hecho de que fue diseñado para imágenes fotográficas.

PNG.

Este formato es apoyado por el consorcio W3 debido a sus características especialmente adecuadas para la web. PNG utiliza compresión sin pérdidas.

A diferencia de GIF o JPEG, PNG puede almacenarse a muchas profundidades de bit diferentes utilizando diversos medios de almacenamiento. Por ejemplo, GIF sólo se puede almacenar a profundidades de bit de 8 bits o menores y JPEG de 24 bits. PNG puede almacenarse a 8, 24 o 32 bits.

Una característica importante es que PNG se creó de manera que se puede especificar en el formato las características de la plataforma donde fue creado, de manera que al menos en teoría, la imagen pueda ser recreada de forma fidedigna. Por ejemplo, se puede tratar de incorporar la misma corrección gamma. Sin embargo, en la práctica esto es dificultoso pues normalmente no se conoce el valor de gamma. Esta característica no ha sido desarrollada por parte de los fabricantes de navegadores.

El formato PNG dispone de un auténtico canal de transparencia alfa donde se pueden establecer 256 niveles de transparencia parcial. En PNG se utiliza por lo tanto un esquema de 4 canales RGBA (RGB + alfa).

TIFF.

TIFF es el acrónimo de Tagged Image File Format. Se trata de un formato muy popular para el intercambio de imágenes que no requieren la capacidad de ser secuenciadas (streamability). Esta capacidad significa que los elementos de la imagen pueden ser decodificados progresivamente, mostrándose sin necesidad de llegar a decodificar todo el fichero. Esta capacidad si bien no es necesaria en muchas aplicaciones, es importante en otros casos como por ejemplo la web. Por este motivo TIFF no se utiliza normalmente en páginas web.