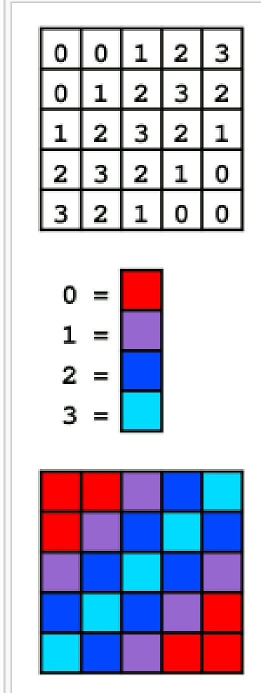
Indexed color

From Wikipedia, the free encyclopedia

In computing, **indexed color** is a technique to manage digital images' colors in a limited fashion, in order to save computer's memory and file storage, while speeding up display refresh and telecom transfers. When an image is encoded this way, the color information is not directly carried by the image pixel data, but it is stored into a separate piece of data called a **palette**: an array of color elements, in which every element, a color, is indexed by its position within the array. This way, each pixel does not contain the full specification of its color, but only its index into the *palette*. This technique is sometimes referred as **pseudocolor**[1] or **indirect color**,[2] as colors are addressed indirectly.



A 2-bit indexed color image. The color of each pixel is represented by a number; each number (the *index*) corresponds to a color in the color table (the *palette*).

Contents

The palette size

Main article: Palette (computing)

The palette in itself stores a very limited number of distinct colors, up to 4, 16 or 256 are the most common cases. These limits are often imposed by the target architecture's display adapter hardware to which the indexed color image is intended to, so it is not a surprise that those numbers are actually exact powers of two (the binary code): $2^2 = 4$, $2^4 = 16$ and $2^8 = 256$. While 256 values can be fitted into a single 8-bit byte (and then a single indexed color pixel occupies a single byte), pixel indices ranging 16 values (4-bit, a nibble) and lesser number of colors can be packed together into a single byte (two nibbles per byte if 16 colors are employed and four 2-bit pixels per byte if using 4 colors). Sometimes, a mere 1-bit per pixel (bpp), 2-color values can be used and then up to eight pixels can be packed into a single byte, but when such colors are black and white, then the image is considered a *binary image* (sometimes referred as a *bitmap* or *bilevel image*) and not an indexed color image.

In order to save even more space, some indexed color image files can store only the effectively used colors in a given image and not the whole of the palette entries available for the pixel depth employed. Thus, it is not rare to find odd palette sizes of any number between 3 and 255 entries instead of always 256 (for 8 bpp) in such files.

If simple video overlay is intended through a *transparent color*, one palette entry is specifically reserved for this purpose, and it is discounted as an actual available color. Some machines, as the MSX series, had the transparent color reserved by hardware.[3]

Indexed color images with palette sizes beyond 256 entries are rare. The practical limit is around 12-bit per pixel, 4,096 different indices. To use indexed 16 bpp or more does not provide the benefits of the indexed color images' nature, due to the color palette size in bytes may be then greater than the raw image data in itself. Also, useful direct RGB

Pixel color depth

1-bit monochrome
8-bit grayscale

8-bit color
15/16-bit color (High Color)
24-bit color (True Color)
30/36/48-bit color (Deep Color)

Related

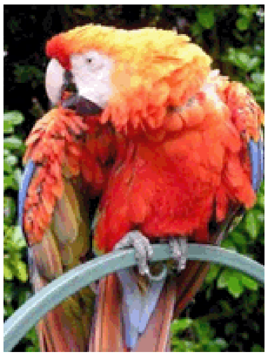
RGB color model

Highcolor modes can be set from 15 bpp and up.

Indexed color
Palette
Web-safe color

If a given real life photograph or image has many subtle color shades, it is necessary to select a limited repertoire of colors to approximate the image using color quantization. Such a palette is frequently insufficient to represent the image accurately; difficult-to-reproduce features such as gradients will appear blocky or as strips (*banding*). In those cases, it is usual to employ dithering, which mixes different-colored pixels in patterns, exploiting the tendency of human vision to blur nearby pixels together, giving a result visually closer to the original one.

Here is a typical indexed 256-color image and its own palette (shown as a rectangle of swatches):




Colors and palettes

See also List of palettes

How the colors are encoded within the color palette map of a given indexed color image depend also on the target platform.

Early color techniques

Many early personal and home computers had very limited proprietary color *full palettes* made up of different color spaces implemented directly in its own hardware, so the color indices were imposed by the manufacturer, as those of the Apple II and the Commodore 64. This had the advantage that the mapping between the pixels values and their correspondent colors was implicit, and then rarely (if ever) was the color map stored as a separate file or as part of an image file.

As display hardware evolved, available *full palettes* grew beyond the systems maximum color depth (the number of different values a pixel can hold), due to their otherwise still limited video memory. For that reason, the indexed color images needed to select a limited repertoire of colors from the given wider *full palette*. Thus, the programs had to load the image colors into the display's color hardware registers/CLUT (or perform other special settings) before loading the actual image pixels into the video memory, which meant that the image palette in itself (i.e. the values to set the actual screen colors) had to be stored along with the raw image data.

RGB

Hardware palettes based on composite video colors such as YPbPr or the like were generally replaced in the mid 1980s by the more flexible RGB color model, in which a given color can be obtained by mixing different amounts of the three primary colors red, green, and blue. Although the total number of different colors depends on the number of levels per primary, and on a given hardware implementation (a 9-bit RGB provides 512 combinations, a 12-bit RGB provides 4,096, and so on), in this model Digital-to-Analog Converters (DAC) can generate the colors -- simplifying the hardware design -- while the software can treat of the number per levels used abstractly and manage the RGB colors in a device-independent fashion. With colors stored in RGB format within the palettes of indexed image files, by appropriate software any image can be displayed (through appropriate transformations) on any of such systems, regardless of the color depth used in the hardware implementation.

Today, display hardware and image file formats that deal with indexed color images almost exclusively manage colors in RGB format, the de-facto standard encoding being the so-called truecolor or 24-bit RGB, with 16,777,216 different possible colors. However, indexed color images are not genuinely constrained to a 24-bit RGB color encoding; image palettes can hold any type of color encoding. For example, the PDF file format does support indexed color in other colorspaces, notably CMYK, and Adobe Distiller by default will convert images to indexed color whenever the total number of colors in an image is less or equal than 256. When using RGB, the TIFF file format can optionally store the RGB triplets with a precision of 16-bit, 65,536 levels per component, yielding a total of 48 bits per pixel. A proposed extension to the TIFF Standard allows[4] non-RGB color palettes, but this was never actually implemented in software due to technical reasons. The color map table of the BMP file format indexed color mode stores its entries in BGR order rather than RGB, and has (in the current version) an additional unused byte for padding to conform to 32-bit word alignment during processing, but it is essentially still a 24-bit RGB color encoding. (An earlier version of the BMP format used three bytes per 24-bit color map table entry, and many files in that format are probably still in circulation, so many modern programs that read BMP files support both variations.)

Pixel bits arrangements

Except for very low resolution graphic modes, early home and personal computers rarely implemented an "all-pixels-addressable" design, that is, the ability to change a single pixel to any of the available colors independently. Their limitations came from employing separate *color attribute* or *color RAM* areas, leading to attribute clash effects. Also, the pixel bits and/or the scan lines of the video memory were commonly arranged in odd ways convenient for the video generator hardware (thus saving hardware costs in a cost-competitive market), but sometimes creating difficulty for the people writing graphics programs. A pixel's bits in indexed-color, all-pixel-addressable images are not always contiguous in video memory or image files (i.e., chunky organization is not always used.) Some video hardware, such as the 16-color graphic modes of the Enhanced Graphics Adapter (EGA) and Video Graphics Array (VGA) for IBM PC compatibles[5] or the Amiga video buffer[6] are arranged as a series of bit planes (in a configuration called planar), in which the related bits of a single pixel are split among several independent bitmaps. Thus, the pixel bits are conceptually aligned along the 3D Z-axis. (The "depth" concept here is not the same as that of pixel depth, though the two concepts are related.)

Early image file formats, as PIC, stored little more than a bare memory dump of the video buffer of a given machine.

Also, some indexed-color image file formats as Graphics Interchange Format (GIF) allow the image's scan lines to be arranged in interleaved fashion (not linear order), which allowed the image to appear on screen little by little while it is still downloading over a low speed communication link (such as an analog telephone modem) so that the computer user can catch an idea of its contents during the seconds before the whole image arrives. Here is an example of a typical vertically interleaved download in four steps:

As seen here, the image has been divided into four groups of lines: group A contains every fourth line, group B contains lines immediately following ones in group A, group D likewise contains the lines immediately following those in group B, and group C contains the remaining lines, which are between group C lines (immediately above) and group A lines (immediately below). These are stored into the file in the order A, C, B, D, so that when the file is transmitted the second received group (C) of lines lie centered between the lines of the first group, yielding the most spatially uniform and recognizable possible image composed of only two of the groups of lines. The same technique can be applied with more groups (e.g. eight), in which case at each step the next group to be sent contains lines lying at or near the centers of some remaining bands that are not yet filled with image data. This method, with four or eight groups of lines, was commonly used on the early World Wide Web of the second half of the 1990s. Rather than leaving the background (black) showing as in the illustration above, the partial image was often presented on screen by duplicating each line to fill the space below it down to the next received image line. The end result was a continuous image with decreased vertical resolution that would increase to full resolution over a few seconds as the later parts of the image data arrived. In other words, take the four steps of the example image above, and for each step, copy each horizontal image line down to cover the horizontal black lines below it, making four new images; these are the four images you would see in succession over a few seconds (in the same place, replacing each other like an animation) in a late '90s web browser such as Netscape 3.

Advantages

Indexed color saves a lot of memory/storage space and/or transmission time: using truecolor, each pixel needs 24 bits, or 3 bytes. A typical 640×480 VGA resolution, truecolor uncompressed image needs 640×480×3 = 921,600 bytes (900 KiB). Limiting the image colors to 256, every pixel needs only 8 bits, or 1 byte each, so the example image now need only 640×480×1 = 307,200 bytes (300 KiB), plus 256×3 = 768 additional bytes to store the palette map in itself (assuming RGB), approx. one third of the original size. Smaller palettes (4-bit 16 colors, 2-bit 4 colors) can pack the pixels even more (to 1/6 or 1/12), obviously at cost of color accuracy.

Indexed color has been widely used in early personal computers and display adapters' hardware to reduce costs (mainly, fewer then-expensive RAM chips) but also for convenient image management with limited-power CPUs (of the order of 4 to 8 MHz) and file storage (cassette tapes and low density floppy disks). Notable computer graphics systems extensively (or even exclusively) using pseudocolor palettes in the 1980s include CGA, EGA, and VGA (for the IBM PC compatibles), the Atari ST, and Amiga's OCS and AGA.

Image files exchanged over the Compuserve net in the early 1990s were encapsulated in the GIF format. Later, the Internet HTML web pages still used the GIF along with other indexed color-supporting file formats such as PNG, to exchange limited-color images quickly and store them in limited storage space.

Most image file formats that support indexed color images also commonly support some compression scheme, enhancing their ability to store the images in smaller files.

Interesting colorized and artistic effects can be easily achieved by playing with the color palette of the indexed color images, for example to produce colorized sepia tone images. Due to the separate nature of the associated palette element of the indexed color images, they are ideal to remap grayscale images into false color ones through the use of false color palettes.

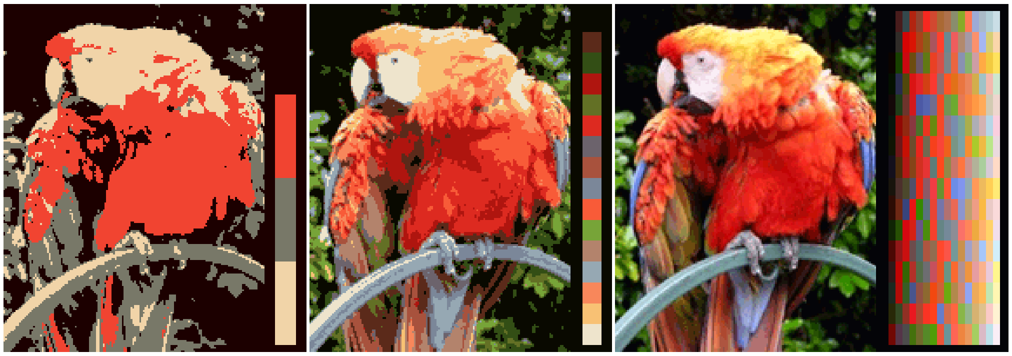
Simple video overlay can be achieved easily through the transparent color technique.

By manipulating the color hardware registers (Color look-up table or CLUT) of the display adapter in the indexed color graphic modes, interesting full-screen color-animation effects can be achieved without the need of entirely redrawing the image, that is, at low CPU time cost; a single change of the register values affects the whole screen at once. Color-map animation is extensively used in the demoscene. The Microsoft Windows boot logo screen in Windows 95, 98, ME, and 2000 Professional (which uses VGA 320x200x256 color display mode because it is the greatest common denominator on all PCs) employs this technique for the scrolling gradient bar across the bottom of the screen; the picture is a static image with no pixels rewritten after it is initially displayed. Custom boot screen images could tap the cycled colors to animate other parts of the images. For example, in the general computing lab sites at Villanova University, they once had a Windows boot screen with a Wildcats logo in which the glowing eyes of the cat would dim and brighten in a pulsing pattern as the progress wave-bar moved.

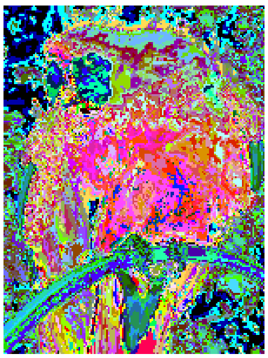
Disadvantages

The main disadvantage of using indexed color is the limited set of simultaneous colors per image. Small 4- or 16-color palettes are still acceptable for little images (icons) or very simple graphics, but to reproduce real life images it becomes nearly useless. Some clever tricks, as color quantization, anti-aliasing and dithering combined together can approximate indexed 256-color images to the original one up to an acceptable level.

For comparison, here are the same image rendered with a 4-, 16-, and 256-color size with adaptive palettes (the best picked selected colors) without dithering, (full truecolor version at top):

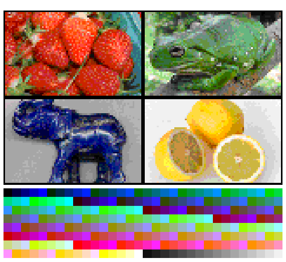


Indexed color images are heavily dependent of their own color palettes. Except for a few and well known common fixed-color palettes (as that of the Color Graphics Adapter—CGA), raw image data and/or color map tables cannot be reliably exchanged between different image files without some kind of intermediate mapping. Also, if the original color palette for a given indexed image is lost, it can be near impossible to restore it. Here is an example of what happens when an indexed color image (the previous parrot) has associated an incorrect color palette:



Indexed color graphic modes for display adapters have the 16- or 256-color limit imposed by hardware. Indexed color images with rich but incompatible palettes can only be accurately displayed one at a time, as in a slideshow. When it is necessary to show multiple images together, as in a mosaic of thumbnails, usually a common or *master palette* is used, which encompasses as many different hues as possible into a single set, thus limiting even more the overall accurate color availability.

Here is a mosaic of four different indexed color images rendered with a single shared *master palette* of 6-8-5 levels RGB plus 16 additional grays. Note the limited range of colors used for every image, and how many palette entries left unused.



Many indexed color display devices do not reach the 24-bit limit for the full RGB palette. The VGA for IBM PC compatibles, for example, only provides a 18-bit RGB with 262,144 different possible colors in both 16- and 256- indexed color graphic modes.

Some image editing software allow to apply a gamma correction over the colors of the palette for indexed color image files. In general for files, to apply a gamma correction directly to the color map is a bad practice, due to the original RGB color values being lost. Always it is better to apply the gamma correction through the display hardware (most modern display adapters support this feature), or as an active intermediate step of the rendering software through some kind of color management, which preserves the original color values. Only when the indexed color images are intended to systems that lacks any kind of color calibration abilities and they are not intended to be cross-platform interchanged, the gamma correction may be applied over the color table in itself.

Image file formats supporting indexed color

These are some of the most representative image file formats that support indexed color modes. Some of these support other modes (e.g. truecolor), but only the indexed color modes are listed here.

NOTE: most of the formats will also support a color table with fewer colors than the maximum that a given bit depth can offer.

Acronym	Full name	Creator	DOS extension	1–bit (2)	2–bit (4)	3–bit (8)	4–bit (16)	5–bit (32)	6–bit (64)	7–bit (128)	8–bit (256)	Compression
PCX	PC Paintbrush Image File	ZSoft Corporation	.pcx	Yes	Yes	No	Yes	No	No	No	Yes	RLE
ILBM	InterLeaved BitMap	Electronic Arts	.lbm, .iff	Yes	Yes	Yes	Yes	Yes	Yes* (EHB mode, 64-color)	Yes*	Yes*	Uncompressed, RLE
GIF	Graphics Interchange Format	Compuserve	.gif	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	LZW
TGA	TARGA File format	Truevision	.tga, .vda, .icb, .vst	No	No	No	No	No	No	No	Yes	RLE
TIFF	Tagged Image File Format	Aldus	.tif	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Uncompressed, PackBits, LZW (**)
BMP	Device-independent Bitmap	Microsoft	.bmp, .dib, .rle	Yes	No	No	Yes	No	No	No	Yes	Uncompressed, RLE (***)
PSD	Photoshop Document	Adobe Systems	.psd	No	No	No	No	No	No	No	Yes	PackBits
PNG	Portable Network Graphics	PNG Development Group	.png	Yes	Yes	No	Yes	No	No	No	Yes	DEFLATE

** 64- (true, not EHB), 128- and 256-color modes only available for the AGA Amiga chipset.*
*** Native support for proprietary compression schemes.*
**** RLE with optional proprietary Delta-leaps.*

Notes

1. **^** Charles A. Poynton (2003). *Digital Video and HDTV: Algorithms and Interfaces* (http://books.google.com/books?id=ra1lcAwgvq4C&pg=RA1-PA38&dq=pseudocolor&lr=&as_brr=0&ei=RrXER6SjLIecsgPI-NGwCA&sig=IE3Yi_P2hkm5mz_kKA-VJ4tWIzw) . Morgan Kaufmann.
2. **^** http://www.cs.binghamton.edu/~reckert/class5b_01.PDF Computer Graphics, Prof. R. Eckert, Lect. #5, February 2001, Binghamton U., N.Y.
3. **^** http://emu-docs.org/VDP%20TMS9918/Datasheets/TMS9918.pdf Online datasheet for the Texas Instruments TMS9918 Video Chip used in the MSX.
4. **^** The TIFF image file format specification (http://partners.adobe.com/public/developer/en/tiff/TIFFPM6.pdf)
5. **^** Richard Wilton, Programmer's Guide to PC & PS/2 VIDEO SYSTEMS, 1987, Microsoft Press. ISBN 1-55615-103-9
6. **^** Inc. Commodore-Amiga, Amiga Hardware Reference Manual, 1991, Addison-Wesley. ISBN 0201567768

References

- Julio Sanchez and Maria P. Canton (2003). *The PC Graphics Handbook.* CRC Press. ISBN 0849316782.

See also

- Palette (computing)
- Color depth
- Color Look-Up Table
- List of palettes
- Image file formats

- Computer display
- List of home computers by video hardware

External links

- Introduction to indexed color images (http://www.scantips.com/palettes.html)

Retrieved from "http://en.wikipedia.org/wiki/Indexed_color"
Categories: Color | Computer graphics data structures | Computing output devices