A Short History of Device-Independent Color

Robert Buckley
Xerox Webster Research Center, Webster, New York

This paper describes the origin of device-independent color, some of the events that led up to it and methods currently used to implement it. Any one of these topics would merit its own paper; describing them together will necessarily make this history short and selective.

Introduction

Device-independent color is a new application of an old idea: using tristimulus values to describe color in an imaging system. The term has gained currency and the idea new significance within the last ten years with the growing use of networked systems, linking scanners, monitors and printers. With these systems, a user at a personal computer can scan or create a color image, interactively edit it while viewing the result on a monitor, and then print it on any printer available over the network. Device-independent color cleanly separates the editing and printing operations and links them with color descriptions based on tristimulus values.

Tristimulus values have a long history in the development of color imaging systems. Shortly after their standardization in 1931, tristimulus values were applied to the analysis of color printing and the design of color television. But it was the coming together of printing and television displays, aided by digital technology, that gave birth to device-independent color.

Computer Graphics

The notion of device independence originated in computer graphics in the late 1960's; see Ref. [1] for example. The idea was to use one application program to generate equivalent images on a wide variety of display devices, independent of device characteristics such as resolution, as opposed to having a different application program for each device. One way to achieve this independence was to use reference coordinates to decouple the way an application describes the image from the way a device renders it[2]. In Figure 1, the application *App* describes the desired output in terms of user coordinates *U*, which the transformation S converts to reference coordinates *R*. The transformation T converts reference coordinates *R* to output device coordinates *D* for rendering on device *Dev*. Neither the application nor the device knows what coordinates the other uses, but both understand reference coordinates, which is what they use to communicate. The transformation T is different for each different device, according to its imaging characteristics, so that all devices will produce results that look the same, given the same reference coordinate values.

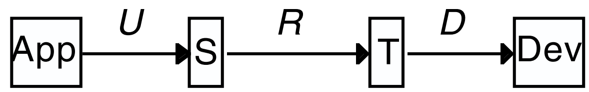

Figure 1. Coordinate Transformations

This approach gives us an operational definition of device independence: the description of image appearance using the coordinate system of a reference or ideal device. Making the CIE Standard Observer the ideal device and using XYZ tristimulus values as reference (and often user) coordinate gives device-independent color. By comparison, device-dependent coordinates are those a device actually uses to render an image.

Although computer graphics invented device independence, it didn't discover device-independent color until relatively late. Because most color output devices were monitors whose color characteristics didn't differ significantly one from another, there was no need for device-independent color. The usual way to get hard copy was to photograph the monitor screen. It wasn't until the tristimulus values used to drive the monitor were calibrated and converted to calibrated printer values did computer graphics meet device-independent color. One of the first encounters is described in Ref. [3].

Page Description Languages

Page description languages such as Interpress and PostScript® use the model of Figure 1 for device-independent graphics. A creator describes the page content with whatever spatial coordinates *U* are convenient, specifies the transformation S to reference coordinates *R*, and then sends *U* and S (instead of *R*) to a printer. The printer supplies the transformation T and combines it with the transformation S to convert the page description from user coordinates *U* to device coordinates *D* for imaging.

The first page description language to adopt device-independent color was Interpress, using the Xerox Color Encoding Standard[4], early versions of which were described in 1986 and 1987[5,6]. This standard was designed for the interchange of color data among applications such as creation, editing, printing, storage, scanning and mail. It offered a choice of three color reference coordinates: a cal-ibrated RGB model and two black-and-white compatible models—a luminance-chrominance model and the CIELAB uniform color space. All were defined or calibrated in terms of XYZ tristimulus values. A source application, such as a document editor, would convert the color coordinates U it used internally to reference coordinates R, using one of these three models, and then transmit reference coordinates to a destination device, such as a printer, which would convert them to internal device coordinates D.

Tristimulus values were the obvious choice as reference coordinates for device-independent color. They provided a standard method of measuring and specifying color. Colors with equal tristimulus values match or look the same to an average human observer viewing them under the same conditions, independent of how they were generated or the device used to reproduce them.

Color Printing

While it has been known since the early 18th century that color mezzotints made with three plates using red, blue and yellow could represent all colors, the analytic science to support this craft did not appear until the 20th century.

One form this science took was the Neugebauer equations, which used a simple model of halftoned color reproduction to express tristimulus values in terms of ink percentages. Hardy and Wurzburg[8] described an electronic scanner that scanned the tristimulus values of the original, converted them to ink values by solving the Neugebauer equations, and then recorded the ink values on a photographic plate as halftoned separations. The printed result was essentially a colorimetric match of the original, with the same tristimulus values.

Although RCA developed a commercial color scanner based on this approach, it didn't see widespread use for several reasons. The Neugebauer equations, as originally formulated, were not an accurate model of the halftone process; Neugebauer[9] himself "didn't trust them enough to apply them for basing color computers on these equations." Another reason was that a colorimetric match was often not possible because of gamut differences between the original and the reproduction, and usually not desirable for aesthetic or editorial reasons. While it was possible to modify the color of the reproduction using Neugebauer equations[10], methods based on photographic masking equations came to be preferred.

Masking equations compute the cyan, magenta and yellow (CMY) ink densities of the reproduction as linear combinations of the red, green and blue densities of the original. They compensate for unwanted ink absorptions. An additional step computes a fourth ink value for the black (K) separation. While the masking equations are no more accurate than the Neugebauer equations for modeling halftoned color, they provide more possibilities for controlling the reproduction. In the hands of an expert, they can give excellent results[11].

The popularity of masking equations in electronic color scanners hindered the adoption of tristimulus values in the printing industry. This was because masking essentially treats color reproduction as a duplication process in which the red, green and blue densities of the original, rather than its tristimulus values, are important. Efficiently obtaining densities meant scanners with narrow spectral sensitivity curves, unlike the broad overlapping spectral sensitivity curves of the Standard Observer. As a result, metamers—colors with different spectral curves but the same tristimulus values—would look different to a scanner and thus would reproduce differently. This was not as serious a problem as it might sound, because most originals used a few types of continuous tone transparency films, and adjustments to the masking equation parameters could compensate for most of the differences.

A color scanner with the same spectral sensitivities as the Standard Observer would eliminate the differential reproduction of metamers, but by itself would not improve the accuracy of color reproduction as long as the scanner used Neugebauer or masking equations to model the process. What would improve the accuracy are empirical techniques that use polynomials or interpolation formulas to convert input densities or tristimulus values to output ink values. Neugebauer was apparently the first to suggest using interpolation to transform tristimulus values to ink values[9]. This shift from model-based to empirical methods was accompanied by a shift from analog to digital computation, and is described in Ref. [12]. Digital lookup tables that use interpolation to compensate for ink, paper and press variables are now a common component in device-independent color systems. They have replaced craft techniques in color reproduction with engineering practices; as Rhodes[12] has pointed out, their users "must be obsessed with calibration and repeatability."

While lookup tables can automate the computation of ink values, color reproduction still requires artistic judgment and editorial changes. In analog systems that used masking equations, this judgment was supplied by a skilled operator who knew from experience how to set the controls to obtain the desired result. Feedback could be obtained from a meter that read the CMYK values of selected critical colors and an operator who knew what they should be. Otherwise, the operator or customer had to wait for a proof, which if unsatisfactory, meant rescanning or manual retouching. The lookup tables in digital systems had no controls equivalent to those offered by masking equations. All this pointed to the need for ways of improving productivity by allowing operators to easily and reliably determine the color the reproduction should have.

In the early 1950's, Neugebauer had the idea of using a projected image to preview the printed result before making separations. He described a "simulator-monitor" which was "similar to a closed-circuit TV set."[13] It included an analog computer that simulated the masking equations and produced ink values, a converter that transformed the ink values into additive signals for color reproduction, and a projector or additive display that showed the result. The operator would adjust the analog controls and immediately see the effect on the projected image, which could be arranged to look like the printed page. This idea was ahead of its time and it is not obvious that it influenced subsequent commercial systems, although the Hell Chromascope[14] closely resembles Neugebauer's simulator-monitor.

A more influential development was Korman's Digital Computer-Scanner System[15]. This was a research system that modified the colors and computed the ink values in separate steps. It was designed to need little operator training to use. Controls for modifying the hue, saturation and lightness of the reproduction were implemented in an approximately uniform color space suggested by Yule. These values were transformed to RGB tristimulus values, which were then converted to colorimetrically-matching ink values by a digital lookup table.

Color Television

While tristimulus values and colorimetry have always figured prominently in printing research, commercial systems have traditionally preferred to use ink values and densitometry. By comparison, commercial color television and colorimetry[7] have always gone hand in hand, and from the beginning color television systems have been "device independent."

In terms of the model of Figure 1, a television camera produces gamma-corrected red, green and blue tristimulus values U, which are converted to luminance and chrominance signals R for broadcast, and then converted at the receiver back to gamma-corrected red, green and blue values D for display. The user can alter the hue, saturation and brightness of the displayed image by modifying the luminance-chrominance signals before they are converted to display signals.

Even without these controls, the tristimulus values of the display and camera images are different, as the system tries to make the appearance of the two the same. It does this by reproducing the display image at higher contrast then the camera image to offset the effect of the dim surround in which the display is typically viewed.

Device-Independent Color Systems

By the 1970's, most of the components of a device-independent color system had been described or demonstrated. It only remained to bring them together.

In the late 1970's, Hell, Scitex, Crosfield and DaiNippon Screen developed color prepress systems in which a digital computer captured the CMYK image from a color scanner and displayed it on a color monitor. The operator would edit the CMYK values with reference to the monitor and when satisfied with the result, output them to a film plotter to make separations. Because they operated on CMYK or printer coordinates, these systems were device dependent. They were mainly used for page composition and retouching, and were essentially digital extensions to a conventional color scanner. While the display was essential for operator feedback, claims that it was adequate for judging the appearance of the final printed result were unconvincing, and the operator still relied on CMYK values to determine what the color would look like when printed. Nevertheless, these systems, which were very expensive, were quite successful.

These systems added digital image processing and a TV display to a graphic arts scanner. The results are rather different the other way round, when a graphic arts printer is added to a digital image processing system with a TV display. This is essentially what Schreiber did in developing what was evidently the first complete device-independent color system[16], beginning in 1978. Although none knew it at the time, his lab at the MIT Research Laboratory of Electronics was already active in device-independent color. For example, uniform color spaces were used in research on color facsimile[17] and adaptive gamut mapping[18].

Two things in particular distinguished the device-independent color system that Schreiber and his students built from the commercial prepress systems that had come before. First, the system used tristimulus values, mainly luminance and chrominance, for editing and storing images. For output, tristimulus values were converted to ink values using a lookup table. Second, the operator interactively adjusted the tristimulus values of the image to give the desired appearance on a monitor designed to simulate the printed page.

The monitor had a white surround so that it was viewed under the same conditions, including visual adaptation state, as the printed page. This was important because color appearance depends on viewing conditions, including surround, as well as on tristimulus values. Because a monitor can show tristimulus values outside the typical printer gamut, an excess-gamut alarm was inserted in the monitor's video path. It alerted the operator when the displayed color was unprintable so that it could be adjusted. But it is also possible for the printer to print colors, mainly yellows, outside the monitor gamut. To minimize this possibility, a monitor with NTSC phosphors was used.

The MIT system was a research project sponsored by Providence Gravure that, while it demonstrated the advantages of device-independent color, was not used in production. As Schreiber noted, "giving up many habits and skills developed by craftsman over years of experience with conventional methods" was undoubtedly an obstacle. This was apparently also true of the Eikonix Designmaster 8000[19], a similar device-independent prepress system developed at around the same time; it was introduced in 1982 but had limited commercial success. On the other hand, device-independent color is now used in color desktop systems by image editing products such as Cachet from Electronics for Imaging, Inc. and Adobe PhotoShop 2.5.

Standards

The idea of device-independent color is now sufficiently well accepted that several standards have been written to implement it.

With reference to Figure 1, one set of standards specifies the reference coordinates R used for interchange. It is up to the originating application to explicitly transform the coordinates U it uses to reference coordinates R. At the receiving device, reference coordinates R are converted to device coordinates D. Examples of this kind of standard are the NTSC standard for analog color television broadcasting, the Xerox Color Encoding Standard described earlier, and the proposal adopted by the ITU-TSS earlier this year that makes CIELAB mandatory for color facsimile transmission. As a rule, these standards focus on data compression and black-and-white compatibility. They occur in regulated or controlled environments and simplify the design of the receiver at the destination.

Another approach to device-independent color data inter-change specifies the reference coordinates *R* and the transform S, which is equivalent to specifying user coordinates *U* and the transform S. This is analogous to the approach that page description languages use to describe spatial coordinates. Applying it to color was proposed by Eastman Kodak in 1988[20, 21]. It was revised and adopted by the ISO to add color to the Office Document Architecture (ODA) standard for document interchange[22] and is recognizable in the way that PostScript Level 2[23] and the ISO Standard Page Description Language implement device-independent color. In all these standards, the reference coordinates are XYZ.

In practice, these standards interchange user coordinates *U* and calibration data, so called because it specifies the parameters of the transform S which calibrates user coordinates by defining their relationship to XYZ reference coordinates. In effect, the calibration data "tags" the user color data. These standards all allow S to take the form of a series of matrices and coordinate mapping tables. With the appropriate parameters, RGB (with or without gamma correction), linear and non-linear luminance-chrominance values, and CIELAB can be calibrated and thus interchanged as user coordinates. The ODA Color Addendum also allows S to be a 4-dimensional lookup table, so that CMYK can be calibrated and interchanged as well. In a related effort, the ANSI IT8 standards committee is standardizing the form of this lookup table.

These standards defer the transform S until it can be combined with the transform T, so that the overall system only implements a single transformation ST. In some cases, this can be the identity transformation. The standards that take this approach emphasize backward compatibility with existing methods of describing color and are designed to work with the widest possible range of applications.

Besides interchange, the model of Figure 1 also provides a framework for describing color management systems, which transform color data between applications and devices. For example, in Apple ColorSync™, the reference coordinates *R* can be XYZ. The transforms S and T are called color matching methods and are obtained from device profiles.

Conclusion

If it hadn't existed, device-independent color would have had to have been invented to deal with the spread of color over networks and to a new group of users. Device-independent color systems are easier to learn and to use than device-dependent systems because they allow users to operate on images in psychophysical and appearance terms. Also device-independent color is more suited to the horizontally distributed systems enabled by networks than is device-dependent color, which is preferred in the vertically integrated systems used in the graphic arts industry. One place to look for future developments is at the interface between these two systems.

This has been a too-brief history of a still-evolving subject, and device-independent color has not yet reached its full potential. Its success will ultimately be determined the value of the solutions it provides and the quality of the images it produces.

References

1. P. A. Woodsford, The Design and Implementation of the GINO 3D Graphics Software Package, *Software–Practice and Experience*, **1**:335-365 (1971)
2. J. Warnock and D. K. Wyatt, A Device Independent Graphics Imaging Model for Use with Raster Devices, *Computer Graphics*, **16**:313-319 (1982)
3. M. C. Stone, W. B. Cowan and J. C. Beatty, A Description of the Color-Reproduction Methods Used for this Issue of *Color Research and Applications*, *Color Res. Appl.*, **11** Supplement: S83-S88 (1986)
4. Xerox Corp., *Color Encoding Standard*, XNSS 289107, Xerox System Integration Standard, Palo Alto, 1991
5. R. R. Buckley, Interpress Standard for Color Printers and Workstations, *Conference Proceedings, Lasers in Graphics*, Anaheim, September 1986, Vol. II, pp. 24-28, and *Seybold Seminars,* Beverly Hills, March 1987
6. P. G. Roetling and R. R. Buckley, Encoding Color for Page Description Languages and other Network Applications, *NCGA Computer Graphics '87 Conf.*, Philadelphia, March 1987
7. W. T. Wintringham, Color Television and Colorimetry, *Proc. IRE*, **39**:1135-1172 (1951)
8. A. C. Hardy and F. L. Wurzburg Jr., Color Correction in Color Printing, *J. Opt. Soc. Am.*, **38**:300-307 (1948)
9. H. E. J. Neugebauer, *TAGA Proc.*, p. 37 (1955)
10. V. L. Marquart and J. S. Rydz, The Application of Neugebauer Equations to Electronic Color Correction, *TAGA Proc.*, pp. 15-27 (1955)
11. J. A. C. Yule, *Principles of Color Reproduction*, John Wiley and Sons, New York, 1967
12. W. L. Rhodes, Proposal for an Empirical Approach to Color Reproduction, *Color Res. Appl.*, **3**:197-201 (1978)
13. H. E. J. Neugebauer, Monitoring as a means of improving color correction, in *Advances in Printing Science and Technology,* W. H. Bank, Ed., Pergamon Press, Oxford, 1961, pp. 145-157
14. U. Gast, The Chromascope–A New Approach to Color Scanner Adjustments, *TAGA Proc.*, pp. 399-410 (1976)
15. N. I. Korman, Accuracy of Color Reproduction with the Digital Computer-Scanner System of Color Separation, *TAGA Proc.*, pp. 156-166 (1972)
16. W. F. Schreiber, A Color Prepress System Using Appearance Variables, *J. Imaging Technol.*, **12**:200-211 (1986)
17. R. D. Solomon, *Color Coding for a Facsimile System*, Ph.D. Thesis, Dept. of Electrical Engineering, Massachusetts Institute of Technology, 1975
18. R. R. Buckley, *Reproducing Pictures with Non-Reproducible Colors*, S. M. Thesis, Dept. of Electrical Engineering, Massachusetts Institute of Technology, 1978
19. A. Masia, A Digital Color Separating System Based on the Principles of Colorimetry, *TAGA Proc.*, pp. 346-361 (1984)
20. S. F. Ring, Device Independent Color–Electronic Printing and Publishing Systems, *Seybold Seminars*, San Francisco, March 1988
21. E. K. Zeise, Device Independent Electronic Color Rendition, *IS&T Fourth Int'l. Congress on Advances in Non - Impact Printing Technologies*, New Orleans, March 1988
22. ISO, International Standards Organization, *Office Document Architecture and Interchange Format*, ISO 8613, Amendment 2 (Colour), 1991
23. Adobe Systems Inc., *PostScript Language Reference Manual*, 2nd Ed., Addison Wesley, Reading, Mass 1990