Specifying color differences in a linear color space (LEF)

N. Rudaz, R. D. Hersch, V. Ostromoukhov
Ecole Polytechnique Fédérale, Lausanne
Switzerland
rudaz@di.epfl.ch

Abstract

This work presents a novel way of generating color differences for synthesizing artistically screened color images. A single color is specified by interacting with the mouse alternately on a constant luminance plane and on a constant hue plane within the LEF color space (the orthogonal space formed by the RGB cube's black-white axis (L) and by its E and F chrominance axes). By interactively selecting a second color point, a color difference is specified. We present a method for extrapolating this color difference throughout all colors of the RGB cube so as to generate consistent color differences, i.e. smoothly varying similar color differences for different colors. The produced artistically screened color patches show that significant luminance differences always generate significant visually perceived differences, whereas significant hue and/or saturation differences do not always generate significant visually perceived differences.

Introduction

For the purpose of generating artistically screened color images, i.e. images whose colors are rendered by artistic screen elements [Ostromoukhov95], we need to be able to specify additive color differences for each of the colors present in the images we intend to render. In order to avoid tedious interaction with the designer, the system should be able to infer from one given color difference pair color differences for all colors of the RGB display gamut.

To achieve these results, we developed the LEF color model, which combines the intuitiveness of the HSL model with the additivity and the linearity of the RGB model. We show how the properties of the LEF space provide a convenient way of picking a pair of difference colors whose combination (weighted mean) matches a given color. From a given color difference pair, we present a method for computing similar color difference pairs for other colors of the RGB cube. The automatic extrapolation of the given color difference to all colors paves the way to the artistic screening of color images.

Color artistic screening

Halftoning can be viewed as the projection of a dense, rich color space into a poorer, more sparse target color space. The color information that cannot be expressed in the target color space and that would be lost during the projection, is instead translated into spatial information that combines colors of the target space in order to provide the illusion of the original color. The relative sizes of the final space's color areas determine the resulting perceived color.
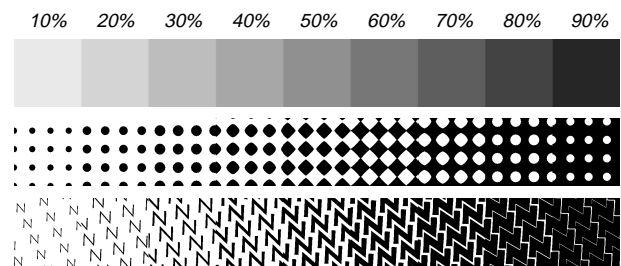

Fig. 1. Gray scale wedge, rendered with clustered dot and artistic dot shapes.

For example, in clustered dot halftoning, the shades of gray between 0% (white) and 100% (black) darkness are rendered by black dots of increasing size. Artistic screening [Ostromoukhov95] is based on the same principle, but conventional screen dots are replaced by artistic shapes.

As shown in fig. 2, the shades of gray that constitute a rectangular or parallelogram region of the original image are reduced to either black or white according to their intensities and their exact positions. The size of the so-obtained black shape is proportional to the darkness of the original region.
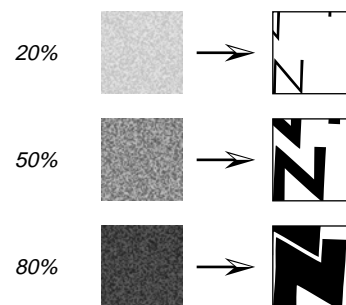

Fig. 2. Relative sizes of artistic dot shapes, determined by the desired gray level.

In this perspective, the final colors (black and white) can be conceived as *diverging* from the original gray shade into white and black screen element parts. The sum of their values (0 and 1) weighted by their relative areas corresponds to the original gray shade.

We would like to extend this principle in order to be able to create artistically screened color images. In other words, given a color **C** and a coverage percentage *s*, we want to find two colors **C.** and **C₊** diverging from **C**, so as to obtain:

$$\mathbf{C} = s\,\mathbf{C}_+ + (1 - s)\,\mathbf{C}_- \qquad (1)$$

If we consider *s* to be the coverage percentage of the screen element relatively to the screen cell, then **C₊** will be the foreground color, and **C.** the background color. Together, **C.** and **C₊** form a *color difference* around **C**.

Color picking and color models

In order to generate artistically screened color images and to keep the foreground shape at the same size, we accept to vary **C.** and **C₊** as a function of **C**. We can, for instance, work with color differences based on luminance, assuming that **C.** is a lighter shade compared to **C**, and that **C₊** is a darker shade. However, in order to offer color differences which may also incorporate hue and saturation differences, we give the user the freedom of associating to a given color **C** a freely selected color difference (**C.**, **C₊**).

In order to interactively specify a color difference, it is necessary to use a color space that is at the same time intuitive and additive. Intuitive, because we want usual color concepts such as lightness, saturation and hue to have a meaning. Additive, because additivity of colors by weighted mean according to their respective surface coverage must be guaranteed in order to fulfill equation (1).

Classical color models cannot be used for this purpose. The calibrated, gamma-corrected RGB model, in which the phosphor's tristimulus CIE-XYZ values are known is consistent and additive, but offers no intuitive way of quickly picking a color. The HSL model [Foley93] derived from the RGB model by non-linear transformations is somehow intuitive, since hue, lightness and saturation can be specified independently. Nevertheless, HSL is non-linear and non-additive. Other perceptual spaces such as the CIE-Lab and the CIE-Luv color spaces are also non-linear and non-additive [Hunt91].

The LEF color space

It is possible to define a color model that combines the additivity of the RGB model with the intuitiveness of the hue-saturation-luminance models by applying a linear transformation to the RGB cube. After transformation, the RGB cube stands on its black corner (fig. 3). Its black-white

axis is vertical and becomes the L-axis in the newly defined LEF color coordinate space. The two other E and F orthogonal axes are lying in the plane perpendicular to the L-axis through point L = 0 (the EF-plane). The E-axis goes through the projection of vertex R of the RGB cube on plane EF. The transformation also includes a scaling operation which ensures that the transformed coordinates lie in the [-1, 1] interval for the horizontal components (E and F coordinates), and in the [0, 2] interval for the vertical component (luminance).
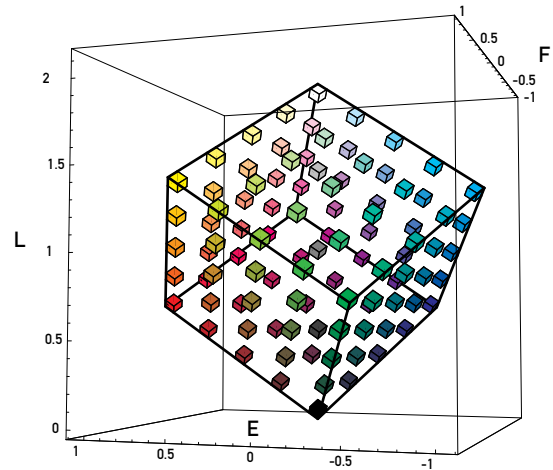

Fig. 3. The RGB cube viewed in LEF space.

This new *LEF* space is somehow similar to the IHS space known in the field of image processing [Niblack85]. Below are the exact formulas for the transformation from RGB to LEF and from LEF to RGB:

$$\begin{bmatrix} L \\ E \\ F \end{bmatrix} = \begin{bmatrix} \frac{2}{3} & \frac{2}{3} & \frac{2}{3} \\ 1 & -\frac{1}{2} & -\frac{1}{2} \\ 0 & \frac{\sqrt{3}}{2} & -\frac{\sqrt{3}}{2} \end{bmatrix} \cdot \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad \text{and} \quad \begin{bmatrix} R \\ G \\ B \end{bmatrix} = \begin{bmatrix} \frac{1}{2} & \frac{2}{3} & 0 \\ \frac{1}{2} & -\frac{1}{3} & \frac{1}{\sqrt{3}} \\ \frac{1}{2} & -\frac{1}{3} & -\frac{1}{\sqrt{3}} \end{bmatrix} \cdot \begin{bmatrix} L \\ E \\ F \end{bmatrix}$$

The vertical L component can be assimilated to the achromatic value of a given color, i.e. its luminance level. The E and F components convey the chromatic information pertaining to that color. More specifically, the quantity $\sqrt{E^2 + F^2}$ measuring the distance between a point and the vertical L-axis can be viewed as the saturation (or colorfulness), meanwhile the horizontal angle between this point and the E-axis, given by $\arctan\left(\frac{F}{E}\right)$ can be viewed as the hue.

As the selection of a particular color is normally made in 2-D, it is necessary to find convenient 2-D representations for selecting points in the LEF space. Based on its definition, there are two natural ways of slicing the LEF space. A first

family of horizontal, parallel planes divides the space into slices of constant luminance perpendicular to the L-axis (fig. 4). The second family of slices, generated by vertical, meridian half-planes, divides the space in slices of constant hue. These two families of slices are referenced hereafter by the symbols *cL* (constant luminance) and *cH* (constant hue).
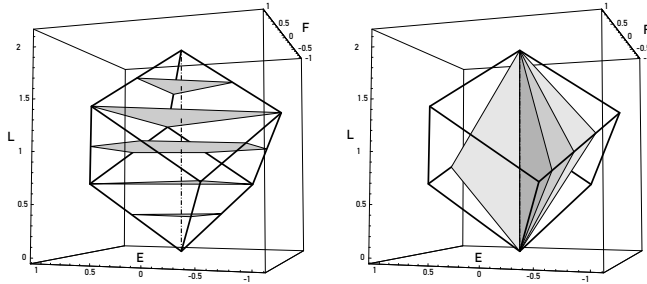

Fig. 4. Elements of the constant luminance (cL) and constant hue (cH) families of slices dividing the LEF space.

Interaction for LEF color picking

The properties of the LEF space are used in an application providing an easy way of interactively creating, adjusting and viewing colors and color differences.
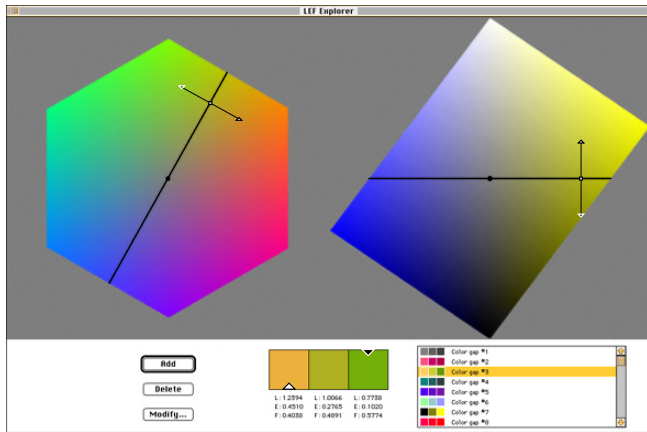

Fig. 5. The main window of the LEF picker application.

The main elements of the color picker are two views of the *cL* and *cH* slices. In each slice, the projection of the other slice is also represented by a thin black line. These lines, corresponding to the projections of the *cH* slice into the *cL* slice and of the *cL* slice into the *cH* slice are referenced as respectively the *pH* line and the *pL* line. They can be moved so as to change the corresponding selected slice. The lines' movement is constrained by the geometrical nature of the slice family they represent: the projection *pH* of the *cH* slice may be rotated around the center of the *cL* slice (the L-axis), while the projection *pL* of the *cL* slice can slide vertically on the *cH* slice, along the L-axis. Adjusting *cL* and *cH* may be viewed

as specifying the luminance and the hue of a particular color **C**.

On each slice projection, a small, square outline is also displayed, which represents the currently selected color **C**. These small squares work like cursors that can be slided along *pH* or *pL*. As both cursors represent the projection of the same point on *cL* and *cH*, sliding one of them along *pH*, resp. *pL*, automatically displaces the other one by the same distance along *pL*, resp. *pH*. Adjusting the color cursors can be viewed as specifying the saturation of a particular color.
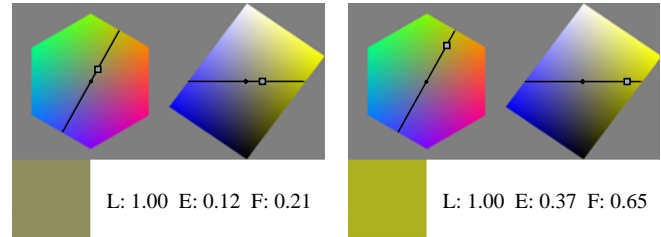

| L: 1.00 E: 0.12 F: 0.21 | L: 1.00 E: 0.37 F: 0.65 |

*Fig. 6. Adjusting the current color **C** by sliding the color cursors.*

Once a color **C** has been selected, the color difference **C₋** and **C₊** diverging from color **C** can be specified by pulling apart from the color cursor one of the small triangles that are flanking it. The white triangle represents **C₋**, and the black triangle, **C₊**. In the examples below, color differences are selected for a coverage percentage $s = 0.5$. Consequently, moving one of these *difference cursors* by a certain amount in a given direction has the effect of displacing the other cursor by the same distance in the opposite direction. Just like the color cursors, the difference cursors represent the projection of two points on *cL* and *cH*; thus modifying their position on one of these slices affects their configuration on the other one.
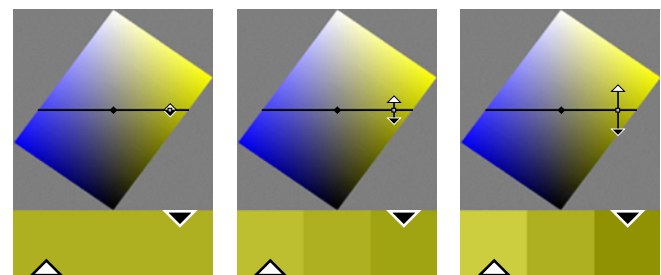
*Fig. 7. Adjusting the color difference (**C₋**, **C₊**) on the cH slice.*

Since all colors must remain within the RGB cube, there is a restriction on how far the color differences **C₋** and **C₊** can diverge from **C**. For a color **C** located on a facet of the RGB cube, the color differences **C₋** and **C₊** must also be located on that facet, i.e. **C₋** and **C₊** must also be colors having a maximal saturation. In the case a color **C** is located on one of the cube edges, there can only be color differences **C₋** and **C₊**

along that edge. And if a color C is located on one of the cube vertices, no non-zero color differences exist.

The LEF and RGB coordinates of such color differences can be saved in a text file, which may later be retrieved by other applications. For instance, the color difference associated to a single element of the RGB colors can be used to produce color differences for the whole RGB cube.

Extrapolating color differences

A given color difference $(\mathbf{C}_-, \mathbf{C}_+)$ is always associated to a single color $\mathbf{C}$. In order to get a similar color difference $(\mathbf{C}_-', \mathbf{C}_+')$ for any RGB color $\mathbf{C}'$, it is necessary to extrapolate the given color difference $(\mathbf{C}_-, \mathbf{C}_+)$. The extrapolation scheme has to respect some constraints so as to produce consistent color differences across the whole RGB cube. Ideally, we would like to keep the perceived color difference constant across all colors of the RGB cube. However, according to our knowledge, no color space exists which provides metrics for large perceptual color differences. As a first approach we therefore choose to extrapolate color differences so as to keep the same euclidian distance in LEF space for the original color difference and for the extrapolated color difference. In addition, we try as much as possible to keep the same differences with respect to the perceptual attributes of luminance, saturation and hue.

Due to the geometry of the color cube in the LEF space, it is often impossible to satisfy both constraints at the same time. In some cases, the euclidian distance of the color difference $\mathbf{C}_-'$ and $\mathbf{C}_+'$ is maintained and the perceptual attributes are allowed to vary. Furthermore, if a variation of the perceptual attributes is necessary, the difference in luminance will always be privileged with respect to the differences in hue and chrominance. We give priority to luminance differences since experience shows that significant luminance differences always generate significant visually perceived differences, whereas significant hue and/or saturation differences do not always generate significant visually perceived differences.

Based on these constraints, the algorithm used to extrapolate a color difference $\Delta\mathbf{C}' = (\mathbf{C}', \mathbf{C}_-', \mathbf{C}_+')$ based on the reference color difference $\Delta\mathbf{C} = (\mathbf{C}, \mathbf{C}_-, \mathbf{C}_+)$ is divided into five cases, which can be summarized as follows.

1) In the special case where $\mathbf{C}$ and $\mathbf{C}'$ are on the same horizontal radius, and $\mathbf{C}'$ lies between $\mathbf{C}$ and the L axis, a color difference is created at the extremity of the radius that lies on the L axis. This color difference, $\Delta\mathbf{N} = (\mathbf{N}, \mathbf{N}_-, \mathbf{N}_+)$, has the same Euclidian distance as $\Delta\mathbf{C}$ and is parallel to the L axis. For points $\mathbf{C}'$ lying between $\mathbf{C}$ and $\mathbf{N}$, the extremities $(\mathbf{C}_-', \mathbf{C}_+')$ of the color difference $\Delta\mathbf{C}'$ are interpolated between $(\mathbf{N}_-, \mathbf{N}_+)$ and $(\mathbf{C}_-, \mathbf{C}_+)$. As shown in fig. 8, the interpolation is not linear, but spherical so as to smoothly rotate the $(\mathbf{C}_-', \mathbf{C}_+')$ vec-

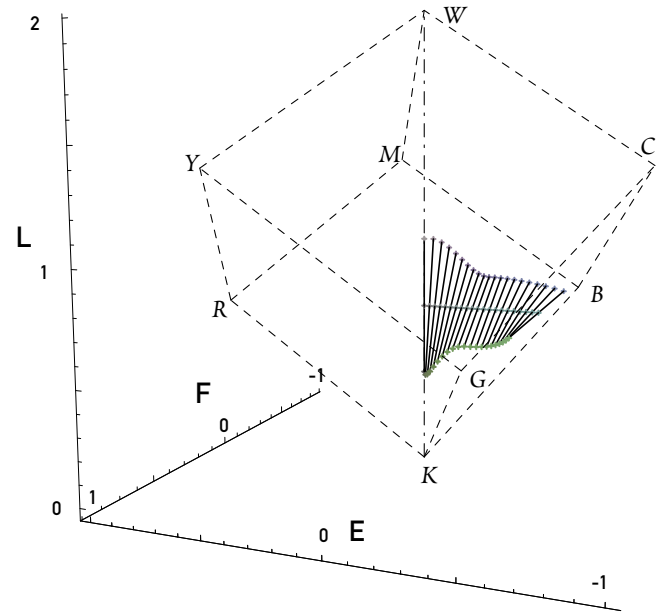tor and to keep its euclidian length constant. This interpolation is performed by using quaternions [Morrison92].

Fig. 8. Spherical interpolation of ΔC.

2) In the other special case where $\mathbf{C}$ and $\mathbf{C}'$ are on the same horizontal radius, and $\mathbf{C}'$ lies between $\mathbf{C}$ and the intersection $\mathbf{P}$ of the radius with the RGB cube, a color difference is created at $\mathbf{P}$. This color difference, $\Delta\mathbf{P} = (\mathbf{P}, \mathbf{P}_-, \mathbf{P}_+)$, has in the best case the same scalar length as $\Delta\mathbf{C}$ and is lying on the projection of $\Delta\mathbf{C}$ on the intersected facet. If necessary, the size of the resulting color difference $\Delta\mathbf{P}$ is reduced so as to ensure that it entirely lies within the intersected cube facet. The extremities $\mathbf{C}_-'$ and $\mathbf{C}_+'$ of the color difference $\Delta\mathbf{C}'$ are then calculated by means of a spherical interpolation between $\Delta\mathbf{P}$ and $\Delta\mathbf{C}$, as in the previous case.

3) If $\mathbf{C}$ and $\mathbf{C}'$ are on the same *cL* slice, but belong to different *cH* slices, then a new color difference $\Delta\mathbf{K} = (\mathbf{K}, \mathbf{K}_-, \mathbf{K}_+)$ is generated by rotating $\Delta\mathbf{C}$ along the L axis so as to bring $\mathbf{C}$ into the same *cH* slice as $\mathbf{C}'$ (fig. 9). Then, depending on the position of $\mathbf{C}'$ relatively to $\mathbf{K}$, $\Delta\mathbf{C}$ is interpolated either between $\Delta\mathbf{K}$ and $\Delta\mathbf{N}$ as in 1), or between $\Delta\mathbf{K}$ and $\Delta\mathbf{P}$ as in 2).

4) If $\mathbf{C}$ and $\mathbf{C}'$ are on the same *cH* slice, but belong to different *cL* slices, then a new color difference $\Delta\mathbf{K} = (\mathbf{K}, \mathbf{K}_-, \mathbf{K}_+)$ is generated by translating $\Delta\mathbf{C}$ so as to bring $\mathbf{C}$ in the same *cL* slice as $\mathbf{C}'$ and to ensure that the relative positions of $\mathbf{K}$ on its new radius R' and of $\mathbf{C}$ on its radius R are identical, i.e. $\dfrac{\text{r}}{\text{R}} = \dfrac{\text{r}'}{\text{R}'}$ (fig. 10). Then, depending on the position of $\mathbf{C}'$ relatively to $\mathbf{K}$, $\Delta\mathbf{C}$ is interpolated either between $\Delta\mathbf{K}$ and $\Delta\mathbf{N}$ as in 1), or between $\Delta\mathbf{K}$ and $\Delta\mathbf{P}$ as in 2).
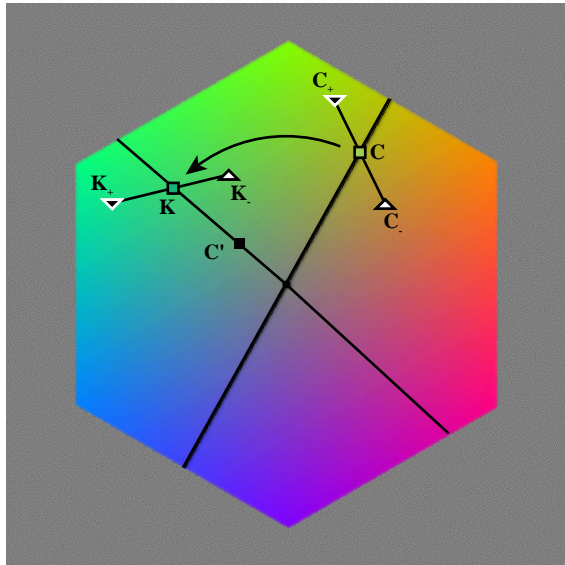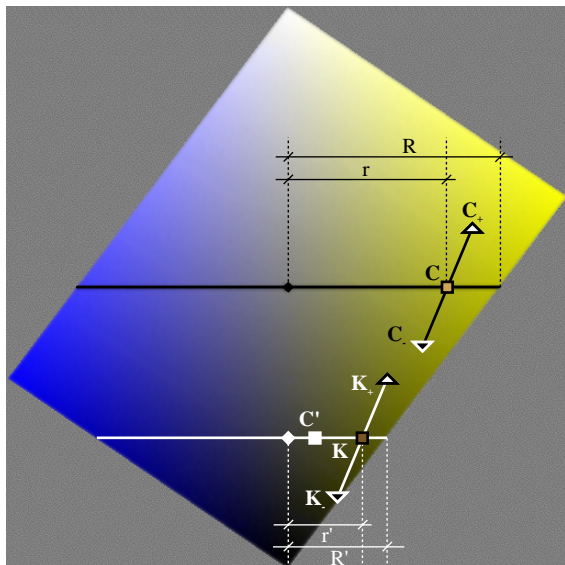
Fig. 9. Rotating ΔC along the L axis.

Fig. 10. Vertical translation of ΔC.

5) In the most general case, when **C** and **C'** are neither in the same *cL* slice nor in the same *cH* slice, a new color difference $\Delta K = (K, K_-, K_+)$ is generated by applying a translation to ΔC as in 4), followed by a rotation as in 3). Then, depending on the position of **C'** relatively to **K**, $\Delta C'$ is either interpolated between ΔK and ΔN as in 1), or between ΔK and ΔP as in 2).

Application: Color Artistic Screening

The plate accompanying this paper in the "Appendix to the CD-ROM Edition" shows color samples produced by artistic screening using the color differences synthesized by the method described in the previous section. This plate is also available on the Web version of the present paper, at the URL <http://diwww/w3lsp/pub/colour/>.

Plate 1 shows six series of linear color wedges, where each series of wedges is screened with three generalized color differences. The first color difference ΔL diverges from the original color along the L axis, and is therefore a pure luminance difference. The second color difference ΔH represents a difference in hue only. The third difference ΔS is orthogonal to ΔL and ΔH and represents a pure saturation difference. The norms of ΔL, ΔH and ΔS are identical. Colour differences ΔL, ΔH and ΔS are specified for colour $C_{LEF} = (2/3, 1/2, 0)$, i.e. $C_{RGB} = (175, 40, 40)$ (half saturated red) and their norm in LEF space is equal to 0.6.

Patches (1) and (2) represent planes cut through the RGB cube perpendicular to the black-white axis, at luminances 1/3 and 2/3 of the maximal luminance (L = 2/3 and L = 4/3 for $L_{max} = 2$). The triangles' vertices are located at 70% of the maximal saturation. Pure luminance differences provide at all wedge positions clearly visible background-foreground differences. Pure hue or saturation differences however do not produce clearly visible background-foreground differences everywhere. Table 1 below summarizes the RGB and LEF values of the extremities of the original color patches (1) and (2).

Table 1

Patch	Top	Bottom	Right	Left
1	*[20, 210, 20]$_{RGB}$* *[0.65, -0.35, 0.61]$_{LEF}$*	*[20, 20, 210]$_{RGB}$* *[0.65, -0.35, -0.61]$_{LEF}$*	*[210, 20, 20]$_{RGB}$* *[0.65, 0.7, 0.0]$_{LEF}$*	*n/a*
2	*[230, 230, 40]$_{RGB}$* *[1.31, 0.35, 0.61]$_{LEF}$*	*[230, 40, 230]$_{RGB}$* *[1.31, 0.35, -0.61]$_{LEF}$*	*n/a*	*[40, 230, 230]$_{RGB}$* *[1.31, -0.7, 0.0]$_{LEF}$*

Wedges (3) and (4) represent line segments in the RGB cube starting from a slightly unsaturated color and ending at an opposite, slightly unsaturated color. Again, luminance differences produce more predictable visible differences than hue or saturation differences.

Wedges (5) and (6) represent color lines in the RGB cube ranging respectively from saturated red to saturated cyan and from saturated blue to saturated yellow. At the wedges' extremities (vertices of the RGB cube), no color differences exist. The gray values at the center of the wedges correspond to the intersection between the color lines and the black-white achromatic axis. At neutral gray, the generated color differences comprise therefore only luminance differences. Table 2 below summarizes the RGB and LEF values of the extremities of the original color wedges (3), (4), (5) and (6).

Table 2

Wedge	Left	Right
3	$[223, 31, 223]_{RGB}$ $[1.25, 0.35, -0.61]_{LEF}$	$[127, 223, 127]_{RGB}$ $[1.25, -0-18, 0.31]_{LEF}$
4	$[223, 223, 31]_{RGB}$ $[1.25, 0.35, 0.61]_{LEF}$	$[127, 127, 223]_{RGB}$ $[1.25, -0-18, -0.31]_{LEF}$
5	$[255, 0, 0]_{RGB}$ $[0.67, 0.94, 0.0]_{LEF}$	$[0, 255, 255]_{RGB}$ $[1.33, -0.94, 0.0]_{LEF}$
6	$[0, 0, 255]_{RGB}$ $[0.67, -0.47, -0.82]_{LEF}$	$[255, 255, 0]_{RGB}$ $[1.33, 0.47, 0.82]_{LEF}$

Conclusion

For the purpose of generating artistically screened color images, we present an interactive method for specifying a color **C** and an associated color difference Δ**C**. We use the LEF space, which combines the additivity and the linearity properties of the RGB color space with the intuitiveness of the hue-luminance-saturation color spaces. Picking a particular color in the LEF space can be performed as easily as it can be done in the HSL space. However, unlike the HSL space, the linear nature of the LEF space enables the consistent specification of color differences diverging from a specific color. We describe a method for automatically extrapolating the color difference associated to a particular color to all colors of the RGB cube.

References

[Foley93]
J.D. Foley, A. vanDam, S.K. Feiner, J.F. Hughes, Computer Graphics: Principle and Practice, Addison Wesley, 1993, p. 590.

[Hunt91]
R. W. G. Hunt, Measuring Colour, 2nd Edition, Ellis Horwood, 1992

[Ostromoukhov95]
V. Ostromoukhov, R.D. Hersch, "Artistic Screening", SIGGRAPH'95, Computer Graphics Proceedings, Annual Conference Series, 1995, pp. 219-228.

[Marcu96]
G. Marcu, S. Abe, "Color Clusterization using a Modified HSL Space", The Fourth Color Imaging Conference: Color Science, Systems and Applications, 1996, pp. 151-155.

[Morrison92]
J. Morrison, "Quaternion Interpolation with Extra Spins", in D. Kirk (Ed.), Graphic Gems III, Academic Press, Inc., 1992, pp. 96-97, 461-462.

[Niblack85]
W. Niblack, An Introduction to Image Processing, Prentice-Hall, 1985

Appendix

Specifying color differences
in a linear color space (LEF)

N. Rudaz, R. D. Hersch, V. Ostromoukhov
Ecole Polytechnique Fédérale, Lausanne
Switzerland
rudaz@di.epfl.ch

Note: these color plates will look best on a color monitor, but they can also be output to a dye-sublimation printer or even a color laser printer. Try to zoom into the figures to have a better look at the details.
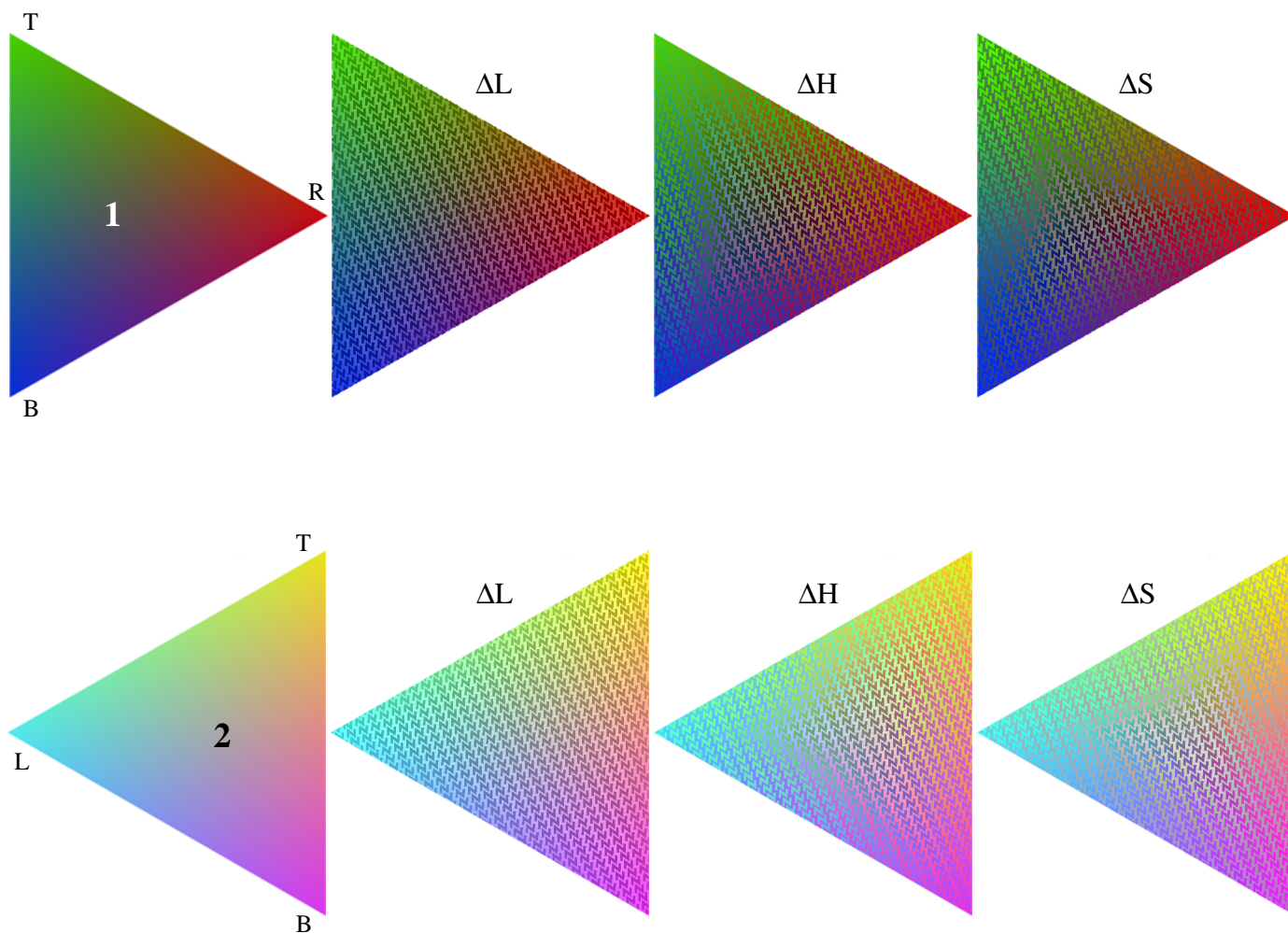

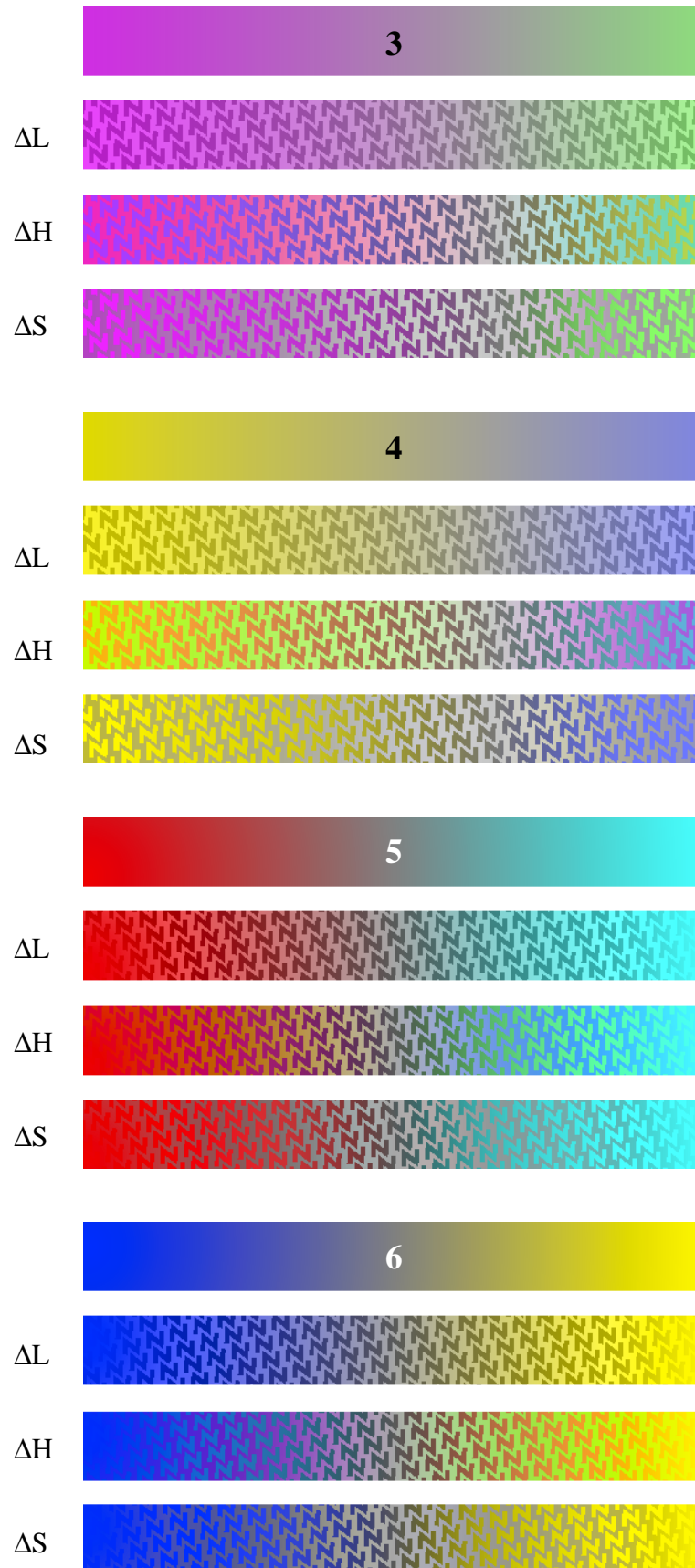*Fig. 1. Patches 1) and 2): two cL slices rendered with three generalized color differences.*

Fig. 2. Patches 3), 4), 5) and 6): four color wedges rendered with three generalized color differences.