

Color Pyramids for Image Processing*

Hubert Konik, Vincent Lozano, and Bernard Laget

Université Jean Monnet, Equipe Ingénierie de la Vision, Lab. T.S.I. UMR5516, 3 rue Javelin Pagusa, BP 505, 42007 Saint-Etienne Cedex 1, France

Color image analysis is still limited by the very significant amount of data. Most real-world images are, of course, not monochrome, but full color. Three-dimensional imaging utilizes large data sets, demanding computer storage and speedy algorithms of critical importance. With this aim in view, without taking into account both spatial and color information, a reducing step cannot be efficient. Thus, a multiresolution process seems to be well adapted, allowing simpler and faster computations. We present here a multiresolution tool, the Gaussian pyramid, first introduced for gray-scale images. We discuss the way to construct the color pyramid in a gamma-corrected *RGB* space, where the color mixing is additive.

Journal of Imaging Science and Technology 40: 535–542 (1996)

Introduction

Presently color image processing lacks effective fast algorithms, notably because of the number of bytes of data represented in a single image. Color images acquired from cameras or scanners or color images to be displayed on a monitor are, in fact, represented in three color bands: the red, green and blue bands. As a result, before all treatments, the information must be minimized. Two approaches are usually used: the quantization process^{1–3} and the multiresolution one. The quantization process involves selecting some representative colors from the color range of the image and then assigning each pixel to one of them. Our research work takes the multiresolution approach, and, more precisely, uses image pyramids, which are multiresolution image representations well matched to the human visual encoding.⁴

The use of pyramidal techniques in computer vision has been studied by many authors since Tanimoto and Pavlidis.⁵ The basic idea of the pyramid structure, formalized only for gray-level images, is to produce a stack of interrelated images with progressively reduced resolutions. The sampling rate of these lower-resolution images is reduced in accordance with the elimination of the higher frequencies. Note that many different functions have been introduced to realize such a representation. Typical ones are convolutions with different kernels, filters, or model fitting.⁶ The most useful scheme is that of Burt,⁷ where the principle is to apply a low-pass filter, generally a Gaussian one, repeatedly generating reduced resolution versions of the input image.

In this report, we propose to generalize such a construction for color images. More particularly, the problem of linear color mixing will be discussed. It allows one to compute, thanks to a real spatio-color approach, a new set of representative colors that is more relevant with regard to the full-resolution image. Last, we will focus on its possible utilization in color image segmentation, based on interactions between consecutive levels. We will conclude by describing the interest for such a tool in a database. Considering a relational database retrieval system, how can we, in fact, find the images with most similar features rapidly?

Tools for a Color Gaussian Pyramid

Gray Level Pyramid. Pyramidal techniques initially used only gray values.⁸ In fact, a gray-level pyramid is a hierarchy of fine to coarse resolution versions of an image, decreasing generally twofold from one to the next (Fig. 1).

Let $2^n \times 2^n$ be the original size of the image. The levels are then of sizes

$$2^{n-1} \times 2^{n-1}, 2^{n-2} \times 2^{n-2}, \dots, 2 \times 2, 1 \times 1, \quad (1)$$

As a result, an entire pyramid contains $(4^{n+1} - 1)/3$ elements. Generally, the values of the current level are computed by convolving the gray values at the previous level with a $K \times K$ kernel and by sampling them at half the current spatial frequency.

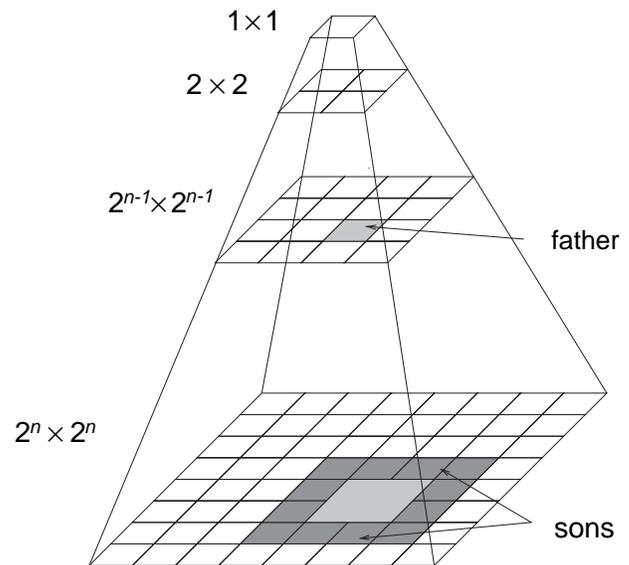


Figure 1. An overlapping pyramid.

Original manuscript received January 16, 1996. Revised September 27, 1996.

* Presented in part at the 3rd IS&T/SID Color Imaging Conference, November 7–10, 1995, Scottsdale, Arizona.

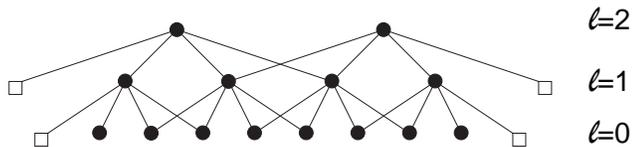


Figure 2. A 1-D overlapping pyramid.

Thus, the value of each element (x, y) at level ℓ is computed as follows:

$$f_{\ell}(x, y) = \sum_{i=0}^{K-1} \sum_{j=0}^{K-1} w(i, j) f_{\ell-1}(2x+i-z, 2y+j-z), \quad (2)$$

where z is $\left\lfloor \frac{K-1}{2} \right\rfloor$.

By definition, the K^2 pixels $(2x+i-z, 2y+j-z)$ at level $\ell-1$ are the *sons* of (x, y) at level ℓ . When (x, y) is used to compute an element at level $\ell+1$, each of them is one of his *fathers* (Fig. 1).

Different forms of the generating kernel $w(i, j)$ have been studied by Burt.⁷ The Gaussian form tends to preserve the shape of the objects and the contrast of the image. It is defined as follows:

$$w(i, j) = \begin{Bmatrix} 0.0169 & 0.0481 & 0.0481 & 0.0169 \\ 0.0481 & 0.1369 & 0.1369 & 0.0481 \\ 0.0481 & 0.1369 & 0.1369 & 0.0481 \\ 0.0169 & 0.0481 & 0.0481 & 0.0169 \end{Bmatrix}. \quad (3)$$

Such a kernel defines the *overlapping Gaussian pyramid* presented in Fig. 1. Figure 2 shows, for example, a 1-D overlapping pyramid built upon 8 pixels (\square symbols stand for computation of border pixels).

Before introducing a color image pyramid, we present in the next section some of the most common color spaces that are used in the field of image processing.

Colorimetry and Color Spaces. Colorimetry is a science that tries to quantify how the human visual system perceives color.⁹ Several standards defined by the “Commission Internationale de l’Eclairage” (CIE) can be applied to the field of computer graphics.

The difficulty of defining a relevant color space can be illustrated by the color matching experiment.¹⁰ In this experiment, the observer is asking to match a test light by specifying intensities of several control lights. These lights are sources at different wavelengths that together cover the visible spectrum.

Combining that experiment with the study of the receptors of our visual system, it has been shown that the human eye can match a test light with three types of receptor in the short, medium, and long wavelength range, respectively.

To be able to match a color with those three control lights, some matching curves have been defined experimentally. One important point is that for some test lights, the observer must add a color light in the left side of the experimental setup. In other words, for some colors, one has to add a control light with a *negative* intensity.

XYZ Color Space. For those reasons, the CIE decided to create a theoretical color space called CIE XYZ color space. This space is based on three hypothetical primaries (X, Y, Z) , which have the following properties:

- all colors in the *positive* octant
- equal integration of matching curves
- Y component matches the luminance.

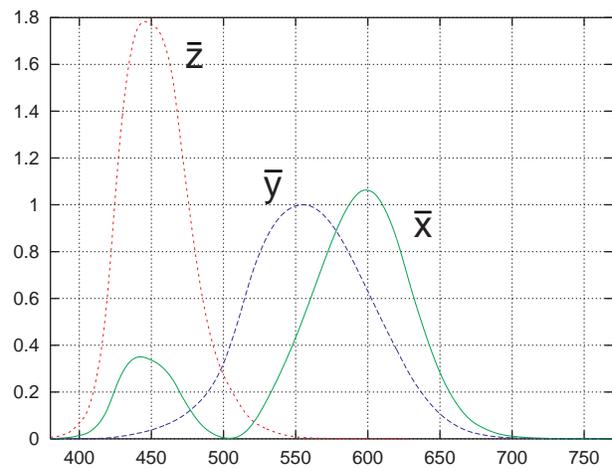


Figure 3. XYZ matching functions.

The global shape of the XYZ matching functions is shown in Fig. 3.

RGB Color Space. In the field of computer graphics, one manipulates the three RGB_d color components as three 8-bit integers (here RGB_d is the *digital* color space), corresponding to the output from cameras and to the way the video cards send their data to the monitor. But these components do not give any information about the color. For example, the triplet $(255,0,0)$ is a red but we don’t know *which* red it is!

Thus, in order to have a colorimetric approach to the RGB_d color space, it is important to understand the links between the CIE XYZ and the RGB_d color space.

Gamma Correction. Because of the nonlinear response of the electron gun of a cathode ray tube (CRT) to voltage input, the RGB_d color space must be compensated. The commonly used correction is the gamma correction (noted γ), which takes into account the specification of a given CRT. The actual value of gamma may range from 2.3 to 2.6.¹¹ From now on, we will note as RGB the gamma-corrected RGB_d space.

RGB to XYZ Transformation. The transformation of the RGB color space to the XYZ color space is a linear transformation taking into account the chromaticity of the three primaries. Usually, these primaries are the set of primaries of the RGB monitor.

Equation 4 gives the transformation formula:

$$\begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} X_r & X_g & X_b \\ Y_r & Y_g & Y_b \\ Z_r & Z_g & Z_b \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix}, \quad (4)$$

where (X_r, Y_r, Z_r) , (X_g, Y_g, Z_g) , and (X_b, Y_b, Z_b) are the XYZ coordinates of the red, green, and blue primaries, respectively.

L*a*b* Color Space. One other restriction of the RGB color space is its *non-uniformity*. In other words, the same distance between two colors does not give the same *perceptual* difference, depending on the color space area considered.

Therefore, in 1976, the CIE L*a*b* color space was introduced by the CIE. This color space, which is a non-linear transformation of the XYZ space, has uniformity. It is built from the XYZ color space with the following relations:¹²

$$\begin{aligned}
L^* &= 116(Y/Y_0)^{\frac{1}{3}} - 16 && \text{for } Y/Y_n > 0.008856 \\
L^* &= 903.3(Y/Y_0) && \text{for } Y/Y_n \leq 0.008856 \\
a^* &= 500[F(X/X_0) - F(Y/Y_0)] \\
b^* &= 200[F(Y/Y_0) - F(Z/Z_0)]
\end{aligned} \quad (5)$$

where (X_0, Y_0, Z_0) are the XYZ coordinates of the *illuminant* and

$$F(u) = \begin{cases} u^{\frac{1}{3}} & \text{for } u > 0.008856 \\ 7.787u + 16/116 & \text{for } u \leq 0.008856 \end{cases} \quad (6)$$

The CIE $L^*a^*b^*$ space, which is often used in colorimetry applications, is not widely used in the field of computer graphics. The main reason is that it implies a space- and time-consuming manipulation of real numbers. The $L^*a^*b^*$ space is well designed for any applications where color distances are computed.

$I_1I_2I_3$ Color Space. Another problem of the color spaces is the *correlation* of the three axes. One common solution, if one wants to work in a decorrelated space, is to perform the Karhunen–Loeve transformation. This transformation consists of three steps:

- compute the covariance matrix \mathbf{W} of an image (or region of an image) I
- compute the eigenvalues of \mathbf{W} , $\lambda_1, \lambda_2, \lambda_3$, with $\lambda_1 \geq \lambda_2 \geq \lambda_3$
- compute the eigenvectors w_i corresponding to the λ_i .

We can then compute the three new components X_1, X_2 , and X_3 as follows:

$$\begin{aligned}
X_1 &= w_{1R}R + w_{1G}G + w_{1B}B \\
X_2 &= w_{2R}R + w_{2G}G + w_{2B}B \\
X_3 &= w_{3R}R + w_{3G}G + w_{3B}B
\end{aligned} \quad (7)$$

This transformation is used to derive color features with large discriminant power. More precisely, the first axis, X_1 , has the largest variance for the region I , and X_2 is the “best” axis among those orthogonal to X_1 .

Ohta, Kanade, and Sakai¹³ have proposed a method to avoid the computation of the eigenvalues for each region or image. They have defined an experimental “well-decorrelated” color space, derived by computing the Karhunen–Loeve transform on eight test images. This experiment involved

$$\begin{aligned}
I_1 &= 1/3(R + G + B) \\
I_2 &= 1/2(R - B) \\
I_3 &= 1/4(2G - R - B)
\end{aligned} \quad (8)$$

as three important components for representing color information.

Color Pyramid Construction

Considering the pyramid in gray levels and the different color spaces, let us then introduce the color pyramid. The Gaussian pyramid construction consists mainly of two steps: a Gaussian convolution and a subsampling operation. Applied to a color image, the first step—which is a *linear* operation—is to perform an *additive color mixture*.

Grassman’s Laws. Grassman’s laws¹⁴ specify that, for an additive color mixture, if two colors C_1 and C_2 , with tristimulus values (X_1, Y_1, Z_1) and (X_2, Y_2, Z_2) , respectively,

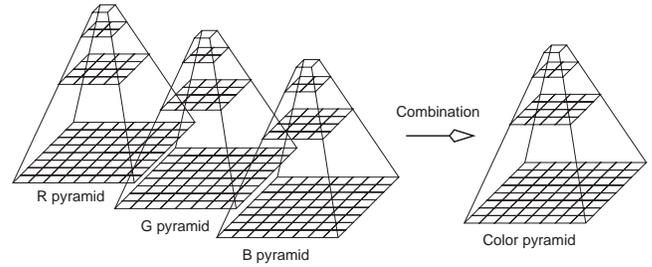


Figure 4. Color pyramid.

are mixed with proportion α and β , respectively, then the tristimulus values (X, Y, Z) of the mixture color C are:

$$\begin{aligned}
X &= \alpha X_1 + \beta X_2 \\
Y &= \alpha Y_1 + \beta Y_2 \\
Z &= \alpha Z_1 + \beta Z_2
\end{aligned} \quad (9)$$

The RGB color space (γ -corrected RGB_d) can then be used to perform an additive color mixture as outlined by Berns, Motta, and Gorzynski.¹⁵

Construction. Equations 10 through 12 give us a way to build each pixel of level l , using a Gaussian kernel. The level l is computed using three Gaussian pyramids, one per color component. For example, in an RGB color space (see Fig. 4), the color $C_l = (R_l, G_l, B_l)$, can be computed as follows:

$$R_l(x, y) = \sum_{i=0}^{K-1-K} \sum_{j=0}^{K-1-K} w(i, j) R_{l-1}(2x + i - z, 2y + j - z) \quad (10)$$

$$G_l(x, y) = \sum_{i=0}^{K-1-K} \sum_{j=0}^{K-1-K} w(i, j) G_{l-1}(2x + i - z, 2y + j - z) \quad (11)$$

$$B_l(x, y) = \sum_{i=0}^{K-1-K} \sum_{j=0}^{K-1-K} w(i, j) B_{l-1}(2x + i - z, 2y + j - z), \quad (12)$$

where z is $\left\lfloor \frac{K-1}{2} \right\rfloor$.

Nonlinear Color Spaces. Let us consider the CIE $L^*a^*b^*$ color space, which, as we said previously, is a non-linear transformation of the CIE XYZ color space. Note that in such a space, the mixture of two colors C_1 and C_2 is not located on the straight line that links the two initial colors. We can formulate this fact by:

$$C_1, C_2 \in \{L^*a^*b^*\} \Rightarrow C \neq \alpha C_1 + \beta C_2. \quad (13)$$

For example, if we map the RGB cube in the $L^*a^*b^*$ color space, we can see in Fig. 5 that the yellow, a mixture of green and red, is not on the straight line linking those colors.

The metric is, in fact, no longer Euclidean, but Riemannian, implying geodesic lines instead of straight ones.¹⁶ Thus, the color pyramid cannot be mathematically constructed in the $L^*a^*b^*$ color space.

Nevertheless, the $L^*a^*b^*$ color space is well designed for the calculation of color distance and can be used in this context. For example, we show here a color quantization process applied to a level of a Gaussian pyramid. Such a process generally consists of two steps: (1) clustering the color space of the level to be quantized, and (2) assigning the centroid of the nearest cluster to each color of this level.

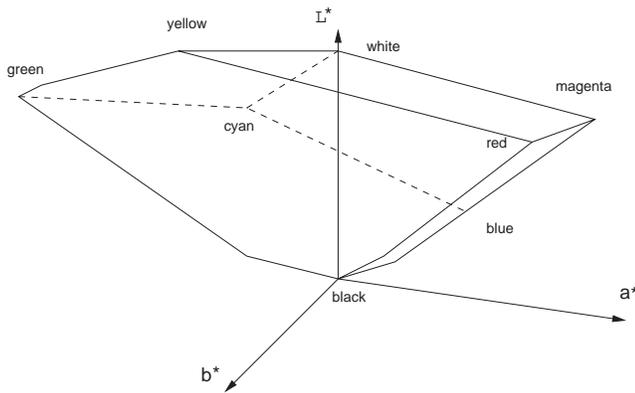


Figure 5. RGB cube in the $L^*a^*b^*$ space.

The second step is often called the *mapping* process and involves many color distance calculations. Performing the mapping in the $L^*a^*b^*$ color space can often increase the quality of the resulting quantized level (see Fig. 6).

Results and Discussion

Pyramid Construction. Figure 7 shows classical images to test our multiresolution process: *airplane*, *mandrill* and *miss*; *airplane* has relatively little contrast in the white colors, *mandrill* is interesting because of its texture, and *miss* covers a wide range of color in the color space.

Figure 8 presents the RGB pyramids of 4 levels associated with each test image, with consecutive images reduced two times in both horizontal and vertical directions. The overlapping kernel is Gaussian.

For better visualization of the propagation of the information, Fig. 9 presents all levels at the same size as the original.

At the same time, we construct a special diagram in the a^*b^* plane for each level (Fig. 10); this diagram is a “view” of the color cluster associated with a level as seen from the white point looking toward the black point along the L^* axis. It can be seen that the more characteristic colors are retained in the higher levels, until the subsampling is not too high. This is particularly evident with the image *miss*, where the small details disappear rapidly. In this connection, it can be demonstrated that the maximum level h_{\max} where an object of diameter d can be detected is:¹⁷

$$h_{\max} \leq \log_2 \left(\frac{d + K - 2}{K - 1} \right). \quad (14)$$

Thus, the structure seems to be quite limited for describing elongated objects, as is the case in the *airplane* image. The object is disproportionate along one direction, and the subsampling is the same for the two different spatial frequency components. A solution exists for luminance images using a local approach.¹⁸ Considering the *mandrill* image, the filtering smooths the texture in accordance with the human visual analysis, especially in the coat. The color associated to the coat is then still representative at the third level, where the color seems to be a logical “average.”

In conclusion, the color range of the image is well retained at each level while the spatial resolution decreases. Moreover, the coarser the resolution is, the more homogeneous the relevant regions appear. The effect of this is a convergence of the clusters to their center of gravity (Fig. 10).

Considering this attractive tool, we now present some possible uses in color image segmentation and in a database. In fact, such a process permits simpler and faster computations.

Segmentation Process. Color image segmentation permits us to delineate the meaningful areas that are usually homogeneous in a color and a texture sense. With this aim in view, histogram-based methods are classically used,¹⁹ but this approach suffers objectively from the lack of local spatial knowledge. Consequently, the chosen representative colors unfortunately can be in poor agreement with the reality. Other methods exist; among them the multiresolution approach seems the most relevant.^{20,21} In fact, this approach simulates the human visual system by taking into account the focus-of-attention principle, assuming that there exists a resolution where the detection and the delineation of a region are easier. More precisely, the color pyramid is attractive because the lower resolutions provide a global view of the image, while the higher resolutions provide the details. The higher resolutions permit us to take into account local information that is important in the human visual process of seeing an image. Generally speaking, the segmentation method depends on a bottom-up process for detecting and a top-down process for delineating.

Starting at the bottom of the pyramid, we must decide on the best cell to represent a region: its root. This detection depends on the analysis of the links inside the pyramid between each cell and its fathers. For example, a color homogeneity criterion, defined as follows can be used:

$$d_h(x, y) = \frac{1}{3} \sum_{i=1}^3 \sqrt{\frac{1}{K^2} \sum_{i=0}^{K-1} \sum_{j=0}^{K-1} (f_h^i(x, y) - f_{h-1}^i(2x+i-z, 2y+j-z))^2}. \quad (15)$$

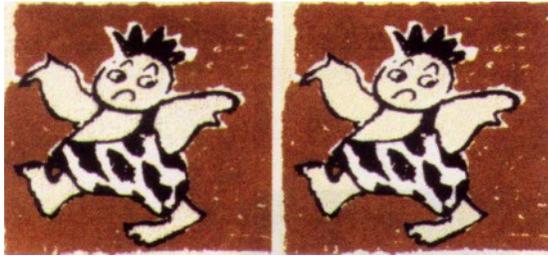
The more homogeneous a set of colors is, the more sensitive this distance is to the color contrast. Specifically, we can allow nodes to refuse to link to any of their fathers if the gray-level value of the current father is some number m times the color dispersion, and we then consider the region homogeneous. In fact, we stop just before the region disappears or is connected with another neighboring one because of the subsampling. After that, each homogeneous region is represented by a root at an optimal level in the pyramid, where it contrasts best from its neighborhood. From this root, a top-down process can be performed until the full resolution image is obtained, keeping at each level the closest sons.

As a conclusion, the results are improved by using two color pyramids: one RGB and one $L^*a^*b^*$ or $L^*u^*v^*$. In fact, the pyramid is constructed in the RGB color space, and consecutive levels are compared in the $L^*a^*b^*$ or $L^*u^*v^*$ space. These color spaces are objectively more uniform and more conceivable because each level is transformed in the $L^*a^*b^*$ system only after the construction in the RGB system.²²

Conclusion

Regarding the propagation of both spatial and color information through the pyramid, the structure defined here can be efficient in image processing (compression, segmentation, database, . . .). The construction must be done in a linear color space such as RGB, and comparisons between consecutive levels are improved in more uniform color spaces such as $L^*a^*b^*$. At present, the number of elements is divided by four at each level, and this growth may be too fast for some applications. We can then work on fractional color pyramids to increase the number of levels.²³

The increasing importance of databases and the need for short computation times have the potential to lead to



(a) RGB quantization

(b) $L^*a^*b^*$ quantization

Figure 6. Mapping in the RGB and $L^*a^*b^*$ color space.



(e) Lut



Figure 7. Test images *airplane*, *mandrill* and *miss*.

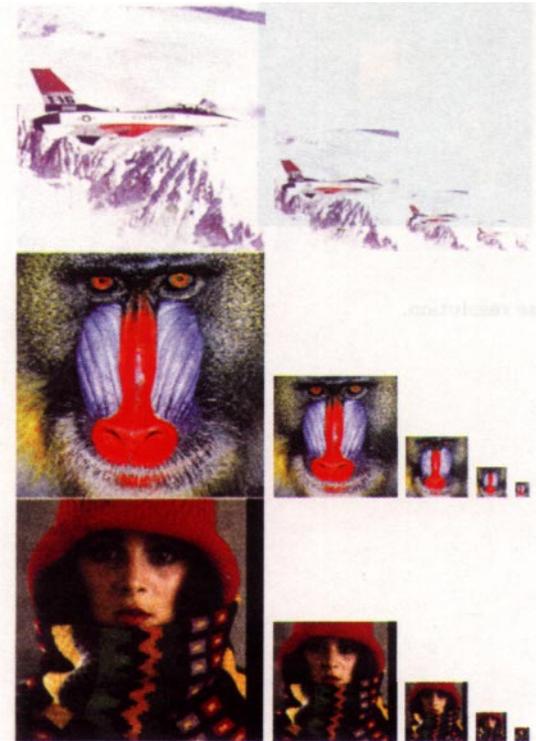


Figure 8. RGB pyramids on the test images.



Figure 9. RGB pyramid levels 1 to 4 with base resolution.

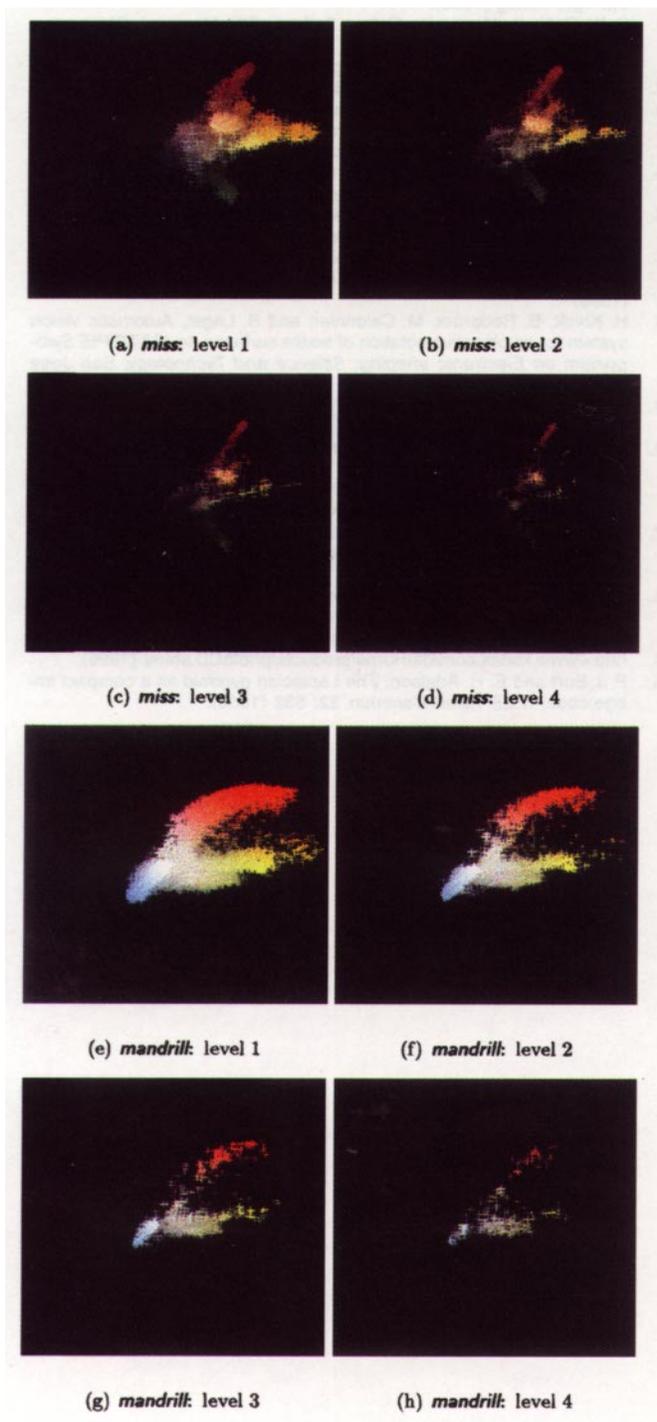


Figure 10. Color cluster projection onto the a^*b^* plane.

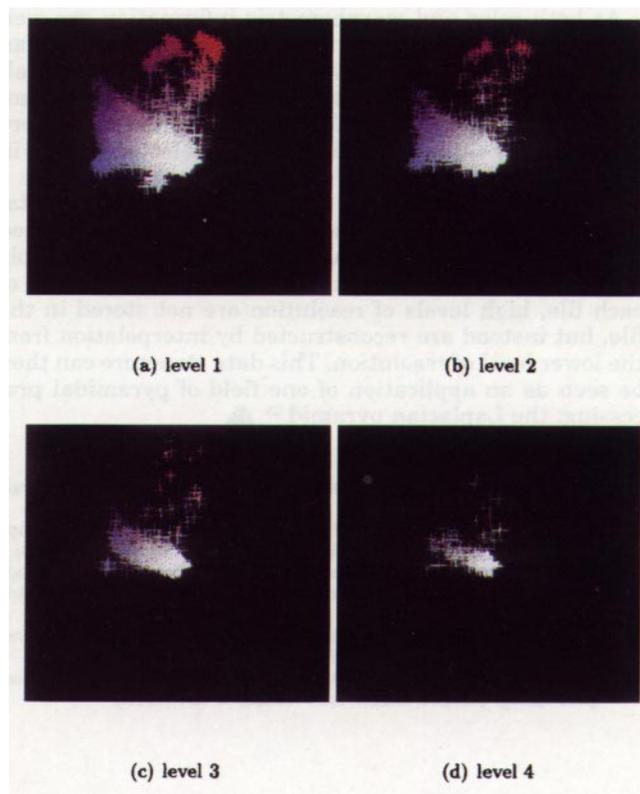


Figure 11. Color cluster projection onto the a^*b^* plane.

the utilization of such a tool. Particularly, fast algorithms are needed for query-by-content retrieval or image matching. In fact, image-by-image review to delineate content features is not feasible when the collection contains a large number of images. Our tool can then be used, for example, in a coarse retrieval process and improve the ability to retrieve images efficiently.

As both color and morphometric information are well propagated through the pyramid, the main features can be at first extracted at a lower resolution level. In fact, all the details are not absolutely necessary for finding images with the most similar features. Subsequently, a more precise query can be launched only for selected images in a higher resolution.

Finally, there is at least one commercial implementation of multiresolution: the PhotoCD format introduced by Kodak.²⁴ This file format stores an image at multiple levels of resolution. To minimize the amount of data of each file, high levels of resolution are not stored in the file, but instead are reconstructed by interpolation from the lower level of resolution. This data structure can then be seen as an application of one field of pyramidal processing: the Laplacian pyramid.²⁵ ▲

References

1. P. Heckbert, Color image quantization for frame buffer display, *Comput. Graph.* **16**: 297 (1977).
2. S. Wan, P. Prusinkiewicz, and S. Wong, Variance-based color image quantization for frame buffer display, *Color Res. Appl.* **15**: 52 (1990).
3. R. Balasubramanian, J. P. Allebach, and C. A. Bouman, Color-image quantization with use of a fast binary splitting technique, *Opt. Soc. Am.* **11**: 2777 (1994).
4. B. A. Wandell, *Foundations of Vision*, Sinauer Associates, Sunderland, Massachusetts, Chap. 8 (1995).
5. S. Tanimoto and T. Pavlidis, A hierarchical data structure for picture processing, *Comput. Graph. Im. Process.* **4**: 104 (1975).
6. F. Chin, A. Choi, and Y. Luo, Optimal generation kernels for image pyramids by piecewise fitting, *IEEE Trans. Patt. Anal. Mach. Intell.* **14**: 1190 (1992).
7. P. J. Burt, Fast filter transforms for image processing, *Comput. Graph. Im. Process.* **16**: 20 (1981).
8. J.-M. Jolion and A. Rosenfeld, *Pyramid framework for early vision*, Kluwer (1994).
9. R. Hall, *Illumination and Color in Computer Generated Imagery*, Springer-Verlag (1989).
10. R. W. G. Hunt, *Measuring Colour*, 2nd ed., Ellis Horwood, Chichester, England, Chap. 2 (1992).
11. C. A. Poynton, Gamma and its disguises, *J. SMPTE* **102**: 1099 (1993).
12. AFNOR, *Couleur Colorimétrie*, p. 518. Recueil de normes Française (1989).
13. Y.-I. Ohta, T. Kanade, and T. Sakai, Color information for region segmentation, *Comput. Graph. Im. Process.* **13**: 222 (1980).
14. D. L. MacAdam, *Color Measurement*, Springer-Verlag (1985).
15. R. S. Berns, R. J. Motta, and M. E. Gorzynski, CRT Colorimetry, part 1: Theory and practice, *Col. Res. Appl.* **18**: 293 (1993).
16. G. Wyszecki and W. Stiles, *Color Science: Concepts and Methods, Quantitative Data and Formulae*, John Wiley and Sons, 2nd ed., Chap. 8 (1982).
17. W. I. Grosky and R. Jain, A pyramid-based approach to segmentation applied to region matching, *IEEE Trans. Patt. Anal. Mach. Intell.* **8**: 639 (1986).
18. H. Konik, B. Redortier, M. Calonnier, and B. Laget, Automatic vision system for an objective cotation of textile surfaces, In *IS&T/SPIE Symposium on Electronic Imaging: Science and Technology*, San Jose (1996) p. 177.
19. S. Tominaga, Color classification of natural color images, *Col. Res. Appl.* **17**: 230 (1992).
20. J. Liu and Y. Yang, Multiresolution color image segmentation, *IEEE Trans. Patt. Anal. Mach. Intell.* **16**: 689 (1994).
21. M. Levine, Region Analysis Using Pyramid Data Structure, in *Computer Vision models*, S. Tanimoto and A. Klinger, New York, pp. 57–100 (1980).
22. H. Konik, A. Tremeau, and B. Laget, Multiresolution color image analysis, In *IS&T/SID Third Color Imaging Conference*, Scottsdale (1995) pp. 82–85.
23. V. Lozano, H. Konik, and B. Laget, Object delineation via local fractional pyramids, In *IEEE Ninth Workshop on Image and Multidimensional Digital Signal Processing*, Belize (1996).
24. <http://www.kodak.com/daiHome/products/photoCD.shtml> (1996).
25. P. J. Burt and E. H. Adelson, The Laplacian pyramid as a compact image code, *IEEE Trans. Commun.* **32**: 532 (1983).