Linear Color Segmentation and its Implementation

Dmitry P. Nikolaev and Petr P. Nikolayev

Institute for Information Transmission Problems, Russian Academy of Sciences, Moscow 101447, Russia

Running title: LINEAR COLOR IMAGE SEGMENTATION

Address for correspondence: Petr P. Nikolayev, IPPI RAN, 101447, Bolshoi Karetny lane, 19, Moscow, Russia. E-mail: nikol@iitp.ru; fax: (095) 209 05 79.

Abstract

A framework for color image segmentation is presented, which combines color histogram analysis and region merging approach. Its main goal is to segment an image at material boundaries (i.e., discontinuities of reflectance properties) while ignoring spatial color inhomogeneities of uniformly pigmented (colored) objects, caused by accidents of illumination and viewing geometry. Theoretical examination of light spectrum transformations upon light reflection from material surfaces and upon interaction with a sensor system shows that in a wide variety of viewed scenes (even containing interreflections and highlight areas) uniformly pigmented objects are projected to the color space of the sensor as planar, linear, or point-like clusters, depending on lighting and viewing conditions and object geometry. To detect such clusters in the color space, three methods are suggested: Generalized Hough Transform method, gradient descent method, and eigenvectors method. A framework algorithm of color segmentation based on region merging approach is developed, which can use any of these methods. Testing this algorithm with both artificially generated and real images shows quite reliable results.

Key words: Color segmentation; Color space; Clusterization; Color invariants; Hough transform

1. Introduction

Static image color segmentation (CS) is used in many computer vision applications, such as object recognition, motion analysis, image compression, etc., as the first step of automatic image processing. The majority of those methods proposed for the delineation of objects in an image are based on the observation that the boundaries of objects in a scene tend to produce abrupt discontinuities of brightness or chromaticity in the image [1]. However, there are many types of physical effects in viewed scenes (shadows, highlights, interreflections, as well as abrupt changes of object surface orientation), which cause changes of image brightness or (to a lesser degree) chromaticity, whose edges do not coincide with the object boundaries. For this reason, gradient-based CS methods usually result in oversegmentation of the image and incorrect delineation of the object areas.

To solve these problems (i.e. to extract the object boundary information from an image containing excess details), some image analysis methods, based on taking into account physical effects in the scene, have been developed as both more reliable and more promising tools than traditional approaches are [2-9]. In principle, the physics-based models require a priori knowledge about some key features of the scene (e.g., the number of light sources and the characteristics of the objects). However, such information is often inaccessible, and therefore the CS task is, in the general case, an ill-posed problem.

Here we present a CS algorithm based on a linear theory of spectral stimulus formation, which summarizes some general geometrical rules of representing objects in the color space (CSp) of the sensor [2-4]. Such a model was first considered by P. Nikolayev for a few types of scenes of different complexity and was named a rank-M theory [2, see also 6 for a summary of Nikolayev's approach]. Shafer's model, called a "Dichromatic reflection model" [4, 7], can be considered as a particular case of the more general Nikolayev's model [6]. The linear model treats CS as splitting an image into regions corresponding to object parts that differ in coloration (staining). It shows that the CS task is well-posed and solvable under certain limitations imposed on the scene complexity (see below). Most real scenes meet these requirements. The linear theory reduces the CS task to the detection of linear submanifolds in the CSp but does not provide a method for solving the latter task. In this paper, we propose such a method as well as demonstrate the applicability of the general approach to object selection and segmentation of real images.

The paper is organized as follows. In Section 2, physical motivation of the generalized linear theory of spectral stimulus formation is given. Also in this section, the shapes of color clusters for a multicolored image in the CSp of the sensor are predicted in accordance with conclusions of the linear theory. In Section 3, the rank

classification of linear submanifolds in the CSp is given and a theoretical background for using the reduced CSp to simplify detecting linear submanifolds is considered. In Section 4, linear cluster detection algorithms are described. Section 5 presents the color version of watershed-based presegmentation algorithm. In Section 6, a region merging color segmentation algorithm based on detecting the linear submanifolds in the CSp is described. Section 7 presents some results on color segmentation for simulated and real images and a discussion of these results.

2. A linear theory of spectral stimulus formation

The linear theory of spectral stimulus formation expresses the overall reflection from a material surface (spectral stimulus) in terms of the sum of different components of reflection. Such a form has been found to be very suitable for a number of computer vision applications. The model is based on three main assumptions. First assumption is that the spectral radiance of illuminants is separable, i.e. can be factored into a spectral and a geometric part. Second, we suppose that nonlinear optical effects are negligible upon light propagation and reflection. The third assumption is that the bi-directional reflectance distribution function (BRDF) of an object surface [10] either does not change its shape when the wavelength of the incident ray changes or can be represented as a sum of such functions. The meaning of these assumptions will be expanded below.

To analyze the structure of a spectral stimulus, consider its formation. The light flux giving rise to a scene image is emitted by light sources, propagates in a transparent medium, and is scattered by object surfaces (Fig. 1). In the general case, a light source can be determined by the spectral radiance:

$$B_\lambda(\lambda,\vec{r},\vec{n}_{out}) = \frac{dP}{(\vec{n}_{out}\cdot\vec{n}_{surf})\cdot d\lambda\cdot d\omega\cdot dS},\ |\vec{n}_{out}|=1,\ |\vec{n}_{surf}|=1. \tag{1a}$$

Here dP is the total power of radiation with wavelengths ranging from λ to $\lambda+d\lambda$, which is emitted by a dS area of the source, located in the vicinity of point $\vec{r}$, in the direction defined by $\vec{n}_{out}$ within the solid angle $d\omega$; while $\vec{n}_{surf}$ is the normal vector to the area dS. If omitting a rare and complicated situation when the radiation spectrum of the source continuously depends on the angle of observation or the point of emission (in the case of a discrete dependence, one may consider such a source as a superposition of a few “simple” light sources), the distribution of the spectral radiance can be rewritten as

$$B_\lambda(\lambda,\vec{r},\vec{n}_{out}) = S(\lambda)\cdot B(\vec{r},\vec{n}_{out}),\ |\vec{n}_{out}|=1,\ \int_0^\infty S(\lambda)d\lambda = 1. \tag{1b}$$

Here $B(\vec{r},\vec{n}_{out})$ is the integral radiance, and $S(\lambda)$ is its relative spectral distribution (Fig. 1). Hereinafter, when considering several sources, we will assume that their spectral distributions are linearly independent (otherwise they can be represented by a superposition of a fewer number of "simple" sources).

It is worth noting that when calculating the irradiance of the surface of a scene object, one often takes into account only the irradiance directly produced by light sources. But, in some cases, secondary illumination produced by light reflected from the same or another surface (by interreflections) must not be ignored. This concerns, for example, narrow and deep folds, where illumination is mainly produced by secondary light sources. Another domain of interreflections: interior lighting design. In a room with light-colored walls, mutual illumination accounts for at least 20% of integral lighting. However, in many cases, similar to the latter, contribution of interreflections in illumination can be equated to the additional diffuse chromatic source.

Let us now generally describe the reflective properties of those objects building up the viewed scene. If omitting nonlinear optical effects, the surface of an object can be characterized by a spectral BRDF (SBRDF), $f_\lambda(\lambda,\vec{r},\vec{n}_{in},\vec{n}_{out})$, (see [10], p.31), which defines the ratio of the spectral radiance of the surface to its irradiance for the wavelength range from λ to $\lambda+d\lambda$ and for every point $\vec{r}$ of the object surface, when illuminated along the direction $\vec{n}_{in}$ and viewed along the direction $\vec{n}_{out}$ (Fig. 1).

It is usually assumed [4, 5, 11, 12] that light reflected from many types of material surfaces includes some primary components: 1) light that once specularly reflects off the interface between air and a smooth area of a material surface; 2) light that many times specularly reflects off the micro details of a rough interface; 3) light that passes through objects, is multiply refracted at their surfaces, is partially absorbed inside them, and, finally, gets back into air. The component of reflection arising from mechanisms 2 and 3 is usually called the diffuse component of reflection. For inhomogeneous dielectrics that include the majority of paints and natural colored materials, the diffuse component is mostly due to the third mechanism and can be called a body component. For a polished metal, there is an extremely weak diffuse component of reflection. For a rough metal, the diffuse component of reflection arises from the second mechanism. The component of reflection, which originates from mechanism 1, is the specular component or highlight. It is peculiar to both inhomogeneous dielectrics and metals. Consider now how to describe all these mechanisms using the SBRDF.

If neglecting dispersion of light, the spectral component of the SBRDF is determined only by features of light absorption in different components of the surface material. Assuming no particular structure of this material, we

shall consider such surfaces that show no dependence of the absorption spectrum upon both the incidence and the viewing angle for all the mechanisms of light absorption. (For example, there is no such a dependence upon a single reflection and/or diffuse scattering in the bulk of a medium). In this case, the SBRDF can be represented by a sum of the following structure:

$$f_\lambda(\lambda,\vec{r},\vec{n}_{in},\vec{n}_{out}) = \sum_k \Phi_k(\lambda)\cdot f_k(\vec{r},\vec{n}_{in},\vec{n}_{out}), \tag{2}$$

where $|\vec{n}_{in}|=1$; $|\vec{n}_{out}|=1$; $\int_0^\infty \Phi_k(\lambda)d\lambda = 1$; and $\Phi_k(\lambda)$ are linearly independent.

Let us call $f_k(\vec{r},\vec{n}_{in},\vec{n}_{out})$ integral BRDFs, and $\Phi_k(\lambda)$ – relative spectral distributions of the reflectance.

The number of significant terms in this sum depends on the properties of the surface. Hereinafter, we shall consider only two types of surfaces: unichromatic, for which the SBRDF can be approximated as

$$f_\lambda(\lambda,\vec{r},\vec{n}_{in},\vec{n}_{out}) = \Phi(\lambda)\cdot f(\vec{r},\vec{n}_{in},\vec{n}_{out}),\ |\vec{n}_{in}|=1,\ |\vec{n}_{out}|=1,\ \int_0^\infty \Phi(\lambda)d\lambda = 1, \tag{3}$$

and bichromatic, for which sum (2) is approximated by two terms. For example, a dielectric surface described by a dichromatic reflection model [4, 15], generally has a SBRDF of the bichromatic type. However, not all the surfaces producing highlights are bichromatic, and not all matte surfaces are unichromatic. Evidently, highlight-producing gray surfaces belong to unichromatic ones. Surfaces of metals (including those vividly pigmented, such as gold or copper) also well fit the unichromatic model [5]. On the other hand, a bichromatic surface should not be necessarily glossy. This case corresponds, for example, to a surface coated with matte flakes of two colors, such that the orientation distributions of the two flake types are different.

Then, if the two main assumptions of the linear theory, related to the scene object surfaces are met (namely, nonlinear optical effects are absent, and the SBRDF can be represented by sum (2)), then the spectral stimulus $F(\lambda,\vec{r})$ of a scene containing several light sources with different relative spectral distributions of radiation and several colored surfaces with different SBRDFs, can be expressed through a sum of several terms, each being the product of a spectral and a spatial function (interreflections are also taken into account):

$$F(\lambda,\vec{r}) = \sum_i g_i(\vec{r})\cdot S_i(\lambda) + \sum_{i,n,k} g_{i,n}(\vec{r})\cdot S_i(\lambda)\cdot\Phi_{n,k}(\lambda) + \\ + \sum_{i,n_1,n_2,k_1,k_2} g_{i,n_1,n_2,k_1,k_2}(\vec{r})\cdot S_i(\lambda)\cdot\Phi_{n_1,k_1}(\lambda)\cdot\Phi_{n_2,k_2}(\lambda)+\ldots\ , \tag{4a}$$

where $S_i(\lambda)$ is the relative power spectral density of i-th light source; $\Phi_{n,k}(\lambda)$ is the k-th spectral distribution

of the reflectance of n-th surface; and $g_{i,n_1,n_2,\ldots,n_m,k_1,k_2,\ldots,k_m}(\vec{r})$ are geometrical factors for a light ray emitted by i-th light source and reflected m times from surfaces $n_1, n_2, \ldots, n_m$. These geometrical factors depend on the integral radiance of the illuminant, the integral BRDFs of the objects that form the viewed scene, and on the geometry of scene lighting and viewing. The first sum in Eq. (4a) represents the component of the stimulus, caused by light that directly comes from the light sources. The second sum originates from light once reflected somewhere in the scene, the third sum concerns light reflected twice, and so on.

Let us now define the rank of an object. For this purpose, consider a family of spectral functions Ω whose elements are differences between the spectral stimuli from each point of the object and the spectral stimulus from an arbitrarily fixed point $\vec{r}_0$ also belonging to the object: $\Omega = \{F(\lambda,\vec{r}) - F(\lambda,\vec{r}_0)\}$. Assume that the rank of Ω in the linear space of the relative spectral functions is n. Then Eq. (4a) can be rewritten as:

$$F(\lambda,\vec{r}) = g_c \cdot M_c(\lambda) + \sum_{i=1}^{n} g_i(\vec{r}) \cdot M_i(\lambda), \tag{4b}$$

where $\{M_i(\lambda)\}$ is the linear basis of Ω, $M_c(\lambda)$ is orthogonal to this basis (if g_c is not zero). Here we assume that if the rank of Ω is zero, there are no terms depending on $\vec{r}$ in Eq. (4b).

It can be seen from Eq. (4b) that the spectral stimulus $F(\lambda)$ from each point of the object is an element of an *n*-dimensional linear submanifold of the spectral function space. This dimension *n* is exactly the rank of the object in the given scene, which we are now defining.

Thus, the ranks allow one to classify the scene types according to the complexity of mathematical models describing them. Rank 0 distribution of the spectral stimuli is produced, for example, by a non-concave uniformly colored (regardless of the number of terms in expansion (2)) surface illuminated by a diffuse light source. A non-concave bichromatic surface illuminated by a single close light source corresponds to rank 2. The same rank 2 also describes a unichromatic non-concave surface illuminated by two close light sources. Thus, the rank of an object depends both on the type of its surface and on the conditions of illumination and viewing.

An appropriate analysis shows that, in the general case (with an arbitrary number of the summands), the linear models adequately describe a wide range of objects and scenes, and that many objects in realistic scenes have ranks lower than 3 [see below and also Refs. 2-6, 9, 13]. The linear theory has been tested experimentally over a wide range of materials, and it is shown to be a good approximation in many cases [5, 14, 15].

If the reflected light field is imaged onto a sensor, at each point (x, y) of the image field the sensor transforms the spectral stimulus determined by Eq. (4) into a vector-stimulus, $\vec{a}(x, y)$:

$$\vec{a}(x,y) = \int_0^\infty F(\lambda, \vec{r}(x,y)) \cdot \vec{\chi}(\lambda) d\lambda, \tag{5}$$

where $\vec{\chi}(\lambda)$ is the vector of the spectral sensitivities of sensor inputs (this vector is 3-dimentional for "human-like" sensors). Sensor-induced nonlinear distortions are neglected here, because they can be eliminated by proper calibration [7].

Thus, the sensor projects the stimulus from a Hilbert space of spectra onto an *N*-dimensional space of vector-stimuli (*N*-dimensional CSp) (Fig. 1). This projection stage is the one limiting the rank of the scene objects that can be analyzed by means of the linear theory. Only if the rank is lower then the CSp dimension, the corresponding color distribution has distinctive linear features in the CSp. As it also follows from (5), the image formed is not affected by the spectral features of the $F(\lambda, \vec{r})$ stimulus within those spectral ranges where all the components of the $\vec{\chi}(\lambda)$ vector are close to zero. Thus, all the above-mentioned requirements of the linear theory should be met only within the spectral range the sensor is capable to detect.

Figure 2 shows examples of RGB color histograms (color distributions) for simulated and real object images. In accordance with predictions of the linear theory, the vector-stimuli corresponding to the objects shown fill up the CSp with linear and planar clusters. Clearly, real color distributions do not fit lines or planes so well as simulated ones do. But nevertheless, in the general case, when the scene contains objects with ranks not exceeding 2, the color histograms of the objects (distorted by a noise) are represented in the 3D CSp by separate compact clusters (rank 0), rod-like clusters (rank 1), or clusters having the form of thin planar layers (rank 2). For all these cases, the CSp does not contain sufficiently large volumetric clusters that would correspond to uniformly pigmented objects.

Thus, the problem of object-based CS is to find the set of all the pixels belonging to a single uniformly pigmented object. Taking into account the conclusions of the linear theory and using the color histograms, it is possible to reformulate this problem in the following form: in order to segment the image, it is enough to decompose the color histogram of the image into point-like, linear and planar distributions.

3. Rank classification and reduction of the color space

It is obvious that the performance of any algorithm of detecting linear submanifolds in an N-dimensional space increases as N decreases (for a fixed resolution in each dimension). In this section, we shall demonstrate that for some types of scenes, the corresponding linear clusters can be detected in a 2D or even 1D space, which makes it possible to considerably speed up the segmentation. To specify these types of scenes, we will have to refine our rank classification and associate it with physical processes that occur in the scene.

According to our definition, the rank of a cluster is the dimensionality of the minimal linear submanifold containing this cluster. Consider now the location of this submanifold in the CSp of the sensor. All the following considerations remain correct for an arbitrary 3D linear CSp. However, for the sake of definiteness, we shall consider the RGB CSp.

Without loss of generality it can be assumed that that the following condition will be met for all sensor channels:

$$\int_0^\infty \chi_i(\lambda)d\lambda = 1. \tag{6}$$

For any sensor that does not satisfy condition (6) we can always go over to the equivalent sensor, satisfying it, by means of CSp scaling. For such sensors the spectral stimulus from the equal-energy illuminants lays on the main diagonal of the CSp. Under equal-energy illuminants we will assume illuminants with $S(\lambda) \approx const$ for the whole spectral range the sensor is capable to detect. Hereinafter such illuminants will be called achromatic, and the main diagonal of the CSp – achromatic axis.

Let us define the following notation: if the linear submanifold includes at least one “achromatic” point ($a_i = a_j \neq 0$) or it is parallel to the achromatic axis (i.e., it includes an infinitely distant “achromatic” point), suffix “a” is added to the rank of the cluster; if the linear submanifold includes a “black” point ($a_i = 0$), suffix “b” is added to the rank of the cluster; if the linear submanifold includes only “chromatic” points ($a_i \neq a_j$) and is not parallel to the achromatic axis, suffix “c” is added to the rank of the cluster. Then, taking into account that earlier we confined our considerations to scenes with ranks from 0 to 2, the new rank classification is a set of 10 ranks: 0_b, 0_a, 0_c, 1_b, 1_a, 1_{ab}, 1_c, 2_b, 2_a and 2_{ab} (Fig. 3). Clusters of the 0_{ab} rank do not exist for apparent reason: a point cannot be located in two different places simultaneously. There are no clusters of the 2_c rank in the 3D space as well: a plane not crossing the achromatic axis is parallel to it, and therefore it belongs to the 2_a rank.

That we unify the cases of crossing and parallelism is connected to the fact that such cases correspond to the scenes of the same type (or, more precisely, too many stipulations are required to define a scene that forms clusters of only one of these two cases). In addition, as it will become evident below, such clusters are indistinguishable with respect to the CSp reduction. We would like to stress that our classification concerns not a cluster itself but a linear submanifold spanned by this cluster. For example, a rank 1_b cluster itself not necessarily includes a "black" point.

Table 1 gives some examples of scenes, according to the classification defined above (there are many other examples for each of the ranks). As it can be seen from the table, different ranks of clusters correspond to sufficiently different types of scenes. Thus, the introduced classification includes 10 types of scenes that significantly differ in physical characteristics of objects and light sources.

Consider the projection of the 3D CSp onto a so-called chromaticity plane [8, 16]. There are a number of ways of defining the chromaticity plane. We have chosen the following one:

$$\begin{cases} \alpha_c = \dfrac{2 \cdot B - (R+G)}{R+G+B} \\ \beta_c = \dfrac{\sqrt{3} \cdot (R-G)}{R+G+B} \end{cases}, \tag{7}$$

where (α_c, β_c) are the coordinates of the projection of the (R,G,B) point onto the chromaticity plane.

Eq. (7) is a central projection transformation of a 3D space onto a plane. Therefore, in the common case, it maps points to points and lines to lines (Fig. 4a). It is also evident that any plane passing through the projection center (the "black" point) is mapped to a line. Thus, if a scene contains no regions of rank 0_b (the "black" point) and rank 2_a (a plane that does not pass through the "black" point), the projections of all the clusters onto the plane (α_c, β_c) are either points or lines. This makes it possible to detect clusters of most ranks rather in the 2D chromaticity space than in the 3D RGB space.

Consider now the behavior of the so-called hue. As in the previous case, there is arbitrariness in the definition of this color characteristic. Let us define the hue, φ_c, and saturation, s_c, as follows:

$$\begin{cases} \varphi_c = \arctan\left(\dfrac{\beta_c}{\alpha_c}\right) = \arctan\left(\sqrt{3} \cdot \dfrac{R-G}{2 \cdot B - (R+G)}\right) \\ s_c = \sqrt{\alpha_c^2 + \beta_c^2} = \dfrac{\sqrt{R^2+G^2+B^2-R \cdot G - R \cdot B - G \cdot B}}{R+G+B} \end{cases}. \tag{8}$$

Note that the domain of φ_c is from $-\pi/2$ to $\pi/2$ and so-called complementary colors are characterized by the same value of φ_c. It can be easily shown that definition (8) implies that the hue of all the points of a cluster is the same for clusters of rank 0_c, 1_b, 1_a and 2_{ab} (Fig. 4b). This allows us to reduce the problem of detecting these clusters to finding spikes in a hue histogram.

Thus, for the scenes corresponding to 8 of the 10 ranks, which the method of finding linear submanifolds in the (R,G,B) space can be applied to, there is no need in analyzing the whole 3D CSp as the distribution of pixels in the (α_c,β_c) space is also linear. Moreover, for 4 of the ranks (Table 2) it is enough to simply examine the hue distribution. Note that if we consider only scenes with achromatic sources and neglect chromatic reflexes, the whole variety of scenes reduces to scenes that form either constant-hue clusters or clusters that entirely lie on the achromatic axis. For these cases, we can confine ourselves to analyzing one-dimensional distributions only.

The main drawbacks of the CSp reduction defined by Eqs. (7) and (8) are the decrease in the number of scene types suitable for processing and considerable “spreading” of those clusters occupying the regions of low brightness (for both cases) or low saturation (for the hue case). Sometimes the reduction may even cause some degeneration. For example, two rank 0 clusters that can be easily separated in the RGB space may have no difference in hue. It should be noted that some earlier works on scene segmentation and color constancy did introduce CSp reduction for simplifying computations [8, 16-18]. However, this approach had no detailed theoretical background, and the question about CSps most suitable for CS was often solved heuristically.

It might seem that the classification that we have introduced is not very practical because there are no achromatic illuminants in the real world. Note, however, that the difference between achromatic and chromatic spectra of the illuminants is quantitative, not qualitative. If the deviation of the illuminant spectrum from the model spectrum of the achromatic illuminant is not very significant, it will result in just some cluster spreading after the CSp reduction. In any case, all kinds of noise are present on the processed images, and one should not expect that the clusters will look like extremely narrow peaks. The methods of linear submanifold detection offered below are not only noise-resistant, but also can persist the inevitable distortions caused by the imperfection of the real world.

Note that CSp reduction by one or two dimensions can be similarly performed in a CSp of a higher dimensionality.

4. Detection of linear submanifolds

So, decomposing a color image histogram into linear model distributions (further in this section, we shall call them cluster models or, even simpler, models) is the way to segment the image. Then, let us define parameterization of these models and methods for their detection in the color distribution of the image.

A model cluster of rank 0 is defined as point $\vec{p}$ of the CSp. The squared discrepancy between an image pixel with color coordinates $\vec{p}_i$ and the model is the following:

$$\rho_0^2(\vec{p}_i) = (\vec{p}_i - \vec{p})^2 . \tag{9a}$$

A model cluster of rank 1 is defined by a straight-line equation written in its canonical form. The parameters of the model are the coordinates of point $\vec{p}$, which this line passes through, and the coordinates of vector $\vec{v}$ that determines the direction of the line. Not all of these coordinates are independent. The connection between them is usually defined by two conditions: $\vec{v}^2 = 1$ and $\vec{p} \cdot \vec{v} = 0$. Then, the squared discrepancy, $\rho^2(\vec{p}_i)$, between image pixel $\vec{p}_i$ and the model is the following:

$$\rho_1^2(\vec{p}_i) = (\vec{p}_i - \vec{p})^2 - (\vec{p}_i \cdot \vec{v})^2 . \tag{9b}$$

A model cluster of rank 2 is defined by the normal equation of a plane. The parameters of the model are the coordinates of a vector, $\vec{n}$, which is normal to this plane, and the distance, d, between the plane and the coordinate origin. Only the direction of vector $\vec{n}$ is important, so we may require that $\vec{n}^2 = 1$ without sacrificing generality. Then, the squared discrepancy between an image pixel, $\vec{p}_i$, and the model is the following:

$$\rho_2^2(\vec{p}_i) = (\vec{p}_i \cdot \vec{n} - d)^2 . \tag{9c}$$

We tested several different methods for determining parameters of the model that optimally describes the cluster of known rank formed in the CSp by the pixel set $\{\vec{p}_i\}$. These are the least-squares method, Generalized Hough Transform (GHT) method, and gradient descent method, the first being the simplest one.

The least-squares method implies finding the minimum (in the space of the parameters of the model) of the root-mean-square discrepancy:

$$S = \sqrt{\frac{\sum_{i=0}^{n-1} \rho^2(\vec{p}_i)}{n}} , \tag{10}$$

where n is the number of pixels in the set $\{\vec{p}_i\}$. In this case, the problem of finding the optimal model is reduced to linear regression that yields an analytical solution.

For a point, the solution is

$$\vec{p} = \frac{\sum_{i=0}^{n-1} \vec{p}_i}{n}; \tag{11a}$$

for a line, we obtain

$$\vec{p} = \frac{\sum_{i=0}^{n-1} \vec{p}_i}{n},\ \vec{v} = \vec{v}_{\max}; \tag{11b}$$

while the solution for a plane is the following:

$$\vec{n} = \vec{v}_{\min},\ d = \vec{n} \cdot \frac{\sum_{i=0}^{n-1} \vec{p}_i}{n}. \tag{11c}$$

Here, $\vec{v}_{\min}$ and $\vec{v}_{\max}$ are the eigenvectors of matrix

$$M_{ij} = \sum_{k=0}^{n-1} \left(\vec{p}_k - \frac{\sum_{l=0}^{n-1} \vec{p}_l}{n} \right)_i \left(\vec{p}_k - \frac{\sum_{l=0}^{n-1} \vec{p}_l}{n} \right)_j, \tag{12}$$

which correspond to the minimum and the maximum eigenvalue of this matrix, respectively. The eigenvalues and eigenvectors of the square matrix with rank 4 or lower can be found analytically.

The eigenvector method is robust with respect to a normally distributed noise. However, it is very sensitive to the presence of noise spikes in the analyzed distribution. Such a noise can appear during the segmentation, for example, when a segment captures a pixel belonging to a different object. This drawback is not inherent to the GHT method [19-21]. The GHT method allows one to pick out of the input dataset those elements well fitting some model and to determine the parameters of this model. For this purpose, the model parameter space is sampled, and each element of the input dataset "votes" for such a combination of the parameters, which makes the element fit the model (as a rule, there are several such combinations). The parameter combination accumulated the maximum number of votes is considered as optimal for the input dataset, while the elements voted for this combination are considered as well described by the model (Figs. 5b and 5c). In our case, the number of votes corresponding to a given parameter combination can be expressed as:

$$V_j = \sum_i g\left(\rho_j\left(\vec{p}_i\right)\right), \tag{13a}$$

where $g(\rho) = \begin{cases} 0, \rho > \Delta\rho \\ 1, \rho \le \Delta\rho \end{cases}$; $\Delta\rho$ is the sampling period; and *j* is the rank of the model.

To suppress noises, which are especially critical when the set $\vec{p}_i$ is scanty, we used Gaussian smoothing of the above distribution in the parameter space. This is equivalent to substituting the vote function of Eq. (13a) with the following one:

$$V_j = \sum_i exp\left(-\frac{\rho_j^2\left(\vec{p}_i\right)}{\sigma^2}\right), \tag{13b}$$

where σ is a parameter determining the smoothing level.

To find the maximum of the vote function, we used either an exhaustive search in the discrete space of parameters of model (which is exactly the GHT) or the gradient descent method. The latter method is much faster then the GHT and requires no parameter space sampling. Therefore, it can provide more accurate results. The main drawback of the gradient descent method is that it requires the knowledge of an initial approximation. In case of the inadequate initial approximation, the method does not guarantee the convergence to the global optimum of the approximation. The final choice on the method should be done taking into account the information about the hardware that will be used for processing. For a fine-grained parallel computer system, the best method will be the GHT method, which can be easily implemented on such systems [21]. However, for serial systems, this method does not suit well because of a large amount of computations. This amount can be lowered using color space reduction: the dimension of the Hough space for a rank 1_c 3D cluster is four, while for a 2D one it is only two (in the chromaticity space). In the case when the GHT method does not provide necessary performance even after color space reduction, we suggest to use the gradient descent method with the initial approximation calculated by the eigenvector method. The raw eigenvector method is the fastest one, but it cannot provide good results when spike noise is present (Fig. 5a). There are also other, often more effective than gradient descent, iterative methods of approximation of the experimental distributions, for instance, those used in the robust regression task [22]. However these methods suffer from the same shortcoming as the method of gradient descent – the lack of guaranty of the convergence to the global optimum under inadequate initial approximation.

All the above methods can be applied to the original CSp as well as to the distribution on the chromaticity plane. As for the analysis of the hue distribution, two peculiarities appear. First, in this case the discrepancy should

be calculated modulo π, that is: $\rho'_0(\varphi_i) = min(|\varphi_i - \varphi|, \pi - |\varphi_i - \varphi|)$. Second, no averaging is defined for hue, therefore either exhaustive search or gradient descent methods should be applied to find a model cluster.

The problem of implementing the linear submanifold detection as an algorithm of color segmentation is that the color distributions of objects geometrically separated in the image may substantially overlap in the CSp. The opposite case also occurs, when the color distribution associated with a single object has the shape of several spatially separated subclusters (for example, if the object has ribs – Figs. 2a and 2e). Therefore, the algorithms of CSp clusterization do not always yield perfect results if the geometrical coordinates of pixels are omitted. Obviously, to overcome this difficulty, we should also take into account the relative arrangement of pixels in the image plane. One of the possible solutions is to reduce the CS task to the decomposition of the color histogram into model distributions that correspond only to connected regions of the image.

Thus, the task of object segmentation implies finding such simply connected domains that all the pixels of each domain match the model of the object (with a certain accuracy), while those pixels neighboring with this domain do not match the model even roughly. We will describe an algorithm that solves this task in the following two sections. The proposed algorithm includes two stages. At the first stage, the algorithm performs initial fine-grained segmentation of the image (Section 5). At the second stage, it builds a correct segmentation map, applying a region merging method to the fine-grained segmentation map with the use of linear cluster detection in the CSp (Section 6).

5. Preliminary image segmentation

The goal of the preliminary image segmentation is to reduce the number of entities that should be processed by the main algorithm. So, the preliminary processing may not produce correct segmentation map itself: the image can be oversegmented, but the pixels belonging to different objects should not be merged into one segment. For such initial partition of the image, we propose a color version of a watershed segmentation algorithm. The description of the watershed algorithm for monochromatic image segmentation can be found in [23, 24]. The main postulate of the watershed approach is that each segment contains one and only one local minimum of some edge-detector response and gradient descent from any pixel of the segment leads to this minimum. In other words, segment boundaries pass along the “ridges” of the edge-detector response (each point of the boundary is a point of local maximum of the response, lying on the line normal to the boundary).

The simplest but nevertheless effective choice on the edge-detector response for monochromatic images is intensity gradient magnitude. This value characterizes the rate of the scalar function change in the direction of the maximum change, and thus allows to distinguish the edge of any orientation. However, the classical gradient operator can be applied only to scalar fields, not to vector ones (a color image is an example of the latter). This problem is often solved using the gradient of some scalar function of the color vector, for example, intensity

$$I(x_1,x_2)=\frac{1}{n}\cdot\sum_{i=1}^{n}C_i(x_1,x_2), \tag{14}$$

where $C_i(x_1,x_2)$ are the color components of the image, n is the number of color components (3, in our case). However, for any edge detector of that sort there exist pairs of colors that are very dissimilar in the CSp but indistinguishable by this edge detector. For instance, intensity edge detector is insensible to arbitrarily large changes of the chromaticity. This shortcoming is missing in the vector analog of gradient, suggested in [25]. This is an operator on the vector field characterizing the rates of the vector function change in the directions of the maximum and minimum change. It can be demonstrated that the directions of the maximum and minimum change of the argument will be ortogonal, and the values of change rates will be equal to the eigenvalues of the matrix 2 by 2:

$$\left|\Sigma-\lambda\cdot E\right|=0\Rightarrow\lambda_{\pm}=\frac{\Sigma_{11}+\Sigma_{22}\pm\sqrt{(\Sigma_{11}-\Sigma_{22})^2+4\cdot\Sigma_{12}^2}}{2}\text{, where} \tag{15}$$

$$\Sigma_{jk}(x_1,x_2)=\sum_{i=1}^{n}\left(\frac{\partial C_i}{\partial x_j}(x_1,x_2)\right)\cdot\left(\frac{\partial C_i}{\partial x_k}(x_1,x_2)\right).$$

Here, E – identity matrix 2 by 2, λ_+ is rate of the function change in the direction of the maximum change. Let λ_+ is called magnitude of a “color gradient” by analogy with magnitude of classical gradient. λ_- is rate of the function change in the direction of the minimum change. For ideal boundaries, changes of the color components $C_i(x,y)$ are spatially matched, and $\frac{\lambda_-}{\lambda_+}\approx 0$. However, at some points the $\frac{\lambda_-}{\lambda_+}$ may be high (such situations are usually caused by noise). One could say that $\frac{\lambda_-}{\lambda_+}$ is the “uncertainty” of the calculated gradient magnitude ($\rho^2(\vec{p}_i)$). It is clear that point with the lowest “uncertainty” is the best edge candidate among points with the same λ_+, so it is wrong to use λ_+ as edge-detector response while ignoring value of λ_-. To overcome this, we use

$\lambda_+ - \lambda_-$ instead, as suggested in [26]. Note that in the case of a monochromatic image ($n = 1$), $\lambda_+ = \left(\frac{\partial C}{\partial x}(x, y)\right)^2 + \left(\frac{\partial C}{\partial y}(x, y)\right)^2$ and $\lambda_- = 0$, so the "color gradient" turns into the classical gradient.

In most cases, the watershed transform leads to oversegmentation. This is due to high sensitivity of the algorithm to edge-detector response variations. The number of segmented regions can be controlled by applying Gaussian blurring with some standard deviation σ right before computing the watershed transform. It should be noted that although an increase in σ reduces the oversegmentation, at the same time it increases the probability of losing object boundaries and the degree of boundary distortion [27]. The final choice on the value of σ should be based on estimating the noise level on the original image.

Examples of watershed segmentation using the color gradient are presented in Figs. 7b and 8b. Of course, the image is oversegmented. Nevertheless, this result is acceptable for us since there are no lost boundaries and now we should process rather small segments than single pixels. As a result, the number of the entities to process is reduced by a factor of several tens.

6. Linear segmentation using a region merging technique

A region merging technique is widely used for object segmentation purposes [24, 28]. Usually, image data is rearranged for this technique in the form of a region adjacency graph (RAG), where each node of the graph represents a segment of the image and contains all necessary information about this segment. Two nodes are connected by an edge if and only if the corresponding segments are adjacent. A weight is assigned to each edge, which characterizes the difference between these segments in the properties the segmentation deals with. Region merging implies successively merging those node pairs connected by the edge having the smallest weight. This is followed by refreshing the adjacent region of the graph. As a rule, the merging procedure is stopped when the weight of the minimum-weight edge exceeds some threshold. The quality of segmentation is determined by the choice of the weight function and (to a lesser degree) by the threshold level. Let us show how to calculate both the weights and the threshold level on the basis of the linear theory presented above.

At the current stage of processing the scene, we already have a preliminary segmentation map obtained through the watershed segmentation. All the pixels of any segment of this map belong to the same uniformly colored object, but the objects are, as a rule, split into several segments. Our goal is now to build the final segmentation map,

using the preliminary map. In the final map, any uniformly colored object or a uniformly colored region of an object should produce a single segment, its boundary coinciding with that of the object or the region.

For the following analysis, it is convenient to define a new characteristic called the model discrepancy of a segment. The rank j model discrepancy of segment s is the standard deviation of the pixels of segment *s* from the optimal model cluster of rank j, which is calculated for segment *s* by any of the methods discussed in Section 4. Of course, the method applied should be the same for all the segments.

Suppose now that all the uniformly colored regions of the original image have the same known rank j. Then, the final segmentation map, which is sought for, should have the following properties. First, each segment should be well described by a model cluster of rank j (with the model discrepancy comparable with the root-mean-square level of noise on the image). Second, any region being the composition of any two neighboring segments could be described by no model cluster (the model discrepancy considerably exceeds the noise level).

Such a map can be obtained by the region merging technique if set the weight of each edge proportional to the model discrepancy of the region consisting of the pair of the segments connected by this edge. For example:

$$d_j[k,l]=\sqrt{\frac{\sum_{i=1}^{n_k}\rho_j^2(\vec{p}_{k,i})+\sum_{i=1}^{n_l}\rho_j^2(\vec{p}_{l,i})}{n_l+n_k}}, \tag{16a}$$

where $d_j[k,l]$ is the weight of the edge that connects the nodes corresponding to segments k and l; j is the rank of the model distribution; $\rho_j(\vec{p})$ is the deviation of pixel $\vec{p}$ from the ideal model for the region joining segments k and l; $\vec{p}_{x,y}$ is pixel y of segment x; and n_x is the total number of pixels in segment x.

However, weight function (11a) has a considerable drawback: the value of $d_j[k,l]$ can be small not due to proximity of the color properties of the two segments but due to the fact that the first segment is much larger than the second and the color distribution of the first is close to the ideal one. This drawback can be eliminated by substituting the segments used to calculate the edge weight with segments equivalent to these in the sense of their relative distribution in CSp but having equalized areas:

$$d_j[k,l]=\sqrt{\frac{\sum_{i=1}^{n_k}\rho_j^2(\vec{p}_{k,i})}{n_k}+\frac{\sum_{i=1}^{n_l}\rho_j^2(\vec{p}_{l,i})}{n_l}}, \tag{16b}$$

Here, a sum of normalized distributions of segments k and l is used as the input for determining the model cluster. Each distribution is normalized to the number of pixels in the corresponding segment.

The region merging using weight function (11b) yields good results when both the noise level and the rank of the image objects (the same for all the objects) are known. Indeed, in most practical applications, the average level of noise can be estimated. However, the second assumption (about the ranks) hardly matches the real conditions. Let us now describe the algorithm for the case when we do not know the ranks of the uniformly colored objects in the scene.

The idea is that region merging with a weight function of rank i does not destroy the boundaries of a rank j zone, if $q_i = q_j \neq 0$. Thus, if we apply the region merging, successively increasing the rank of the weight function, earlier processing stages do not distort the latter ones. However, this does not guarantees correct segmentation yet. Any two points can be connected with a line, and any point and a line can be connected with a plane. Thus, at the latter processing stages, there is a risk of destroying some correctly built areas of the segmentation map. To prevent this from happening, we have to exclude from processing those segments which could appear to be correctly found low-rank uniformly pigmented zones of objects.

Let us now determine those properties allowing one to distinguish between the segments correctly found at the currents stage and the segments to be further processed. Evidently, the latter, i.e., segments corresponding to different zones of an object of rank j after region merging with a weight function of rank i ($i < j$), neighbor both on the image and in the CSp. Therefore, each of these has at lest one neighbor connected by a edge with a near-threshold weight. Let us call those segments, for which this condition is not met, isolated segments of rank i. Correctly found low-rank uniformly pigmented zones of objects will produce isolated segments of the corresponding rank.

On the other hand, such segments can be produced by the illumination conditions. In this case, they should not be excluded from the further processing. For example, a shadow on a matte object illuminated by two sources is an isolated segment of rank 1. However, such shadow segments have a distinctive feature: they neighbor with a higher-rank segment, and joining them to this segment does not increase its rank. These segments can be found after all the non-isolated segments of rank i are merged using a weight function of rank $i+1$.

In accordance with the above ideas, the following segmentation scheme has been implemented for the case of image objects of unknown ranks.

1) Applying the region merging technique with weight function $d_0[k,l]$ and threshold $\vec{p}_{x,y}$ comparable with the noise level of the image.

2) Marking isolated segments of rank 0, i.e., those segments for which the minimal edge weight $d_{\min}$ (among all the adjacent edges) considerably exceeds a certain threshold: $d_{min} > \sigma_G$, where $\sigma_G >> \sigma_0$.

3) Applying the region merging technique with weight function $d_1[k,l]$ and threshold σ_0, ignoring those edges leading to isolated segments.

4) Applying the region merging technique with weight function $d_1[k,l]$ and threshold σ_0, ignoring those edges connecting two isolated segments.

5) Additionally marking rank 1 isolated segments with threshold σ_G.

6) Applying the region merging technique with weight function $d_2[k,l]$ and threshold σ_0, ignoring those edges leading to isolated segments.

7) Applying the region merging technique with weight function $d_2[k,l]$ and threshold σ_0, ignoring those edges connecting two isolated segments.

If the thresholds, σ_0 and σ_G, are properly chosen, the above scheme provides correct segmentation, avoiding the problem of determining the rank of a specific segment of the image.

It is worth noting that a number of specific tasks allow applying simplified segmentation algorithms that use color space reduction (see section 3). For example, if processing sequences of video frames captured indoors under an illumination having a color temperature of 5000-5500K (almost achromatic source), the scene being processed will contain only objects of ranks 0, 1_b, 1_{ab}, and 2_{ab}, provided that the ceiling and the walls are mainly white. For such a task, segmentation can be done in a 1D CSp (hue being the only color axis), with prior localization of weakly saturated regions. For example, the following scheme can be used:

1) Calculating the histogram of saturation (s_c) for pixels of nonzero brightness.

2) With the histogram calculated, determining the threshold, s_{otsu}, that separates low- and high-saturation areas of the image (according to the binarization method proposed by Otsu [29]).

3) If the threshold determined is less than a certain value s_{thres} (comparable with the noise level of the image), those regions where $s_c < s_{otsu}$ should be considered as corresponding to achromatic objects of the scene and should be segmented using simple flood-fill algorithm.

4) Applying the region merging technique for chromatic areas of the image with weight function $d_0[k,l]$ and threshold σ_0 comparable with the noise level of the image. The weight function is calculated in the 1D space of the hue (φ_c). The distances $\rho_j(\vec{p})$ are computed modulo π as it was described in Section 4.

However, if the room walls are colored (as a first approximation, this can be taken into account by introducing a chromatic diffuse light source), the above-mentioned scene might also contain objects of rank 1_c and 2_a, for which the hue is not invariant with respect to illumination conditions. As a result, in such a case the simplified algorithm no longer yields a correct result. Thus, the algorithms based on hue analysis are not universal (see Fig. 6). Nevertheless, they are often used [8] because of their computational simplicity and noise robustness (as compared with those algorithms analyzing the 3D color space). The sub-rank classification allows the researcher to correctly choose the segmentation algorithm at the stage of task definition. For example, in some cases it is reasonable to use an intermediate option – segmentation on the chromaticity plane [16].

7. Results and discussion

Two examples of image segmentation performed by the 3D CSp algorithm are shown in Figs. 7 and 8. They illustrate the segmentation steps and allow one to estimate the quality of the result. Figure 7 presents an image of a simulated scene comprising a two-color cylinder suspended above a plane and in front of a wall. The objects of the scene are illuminated by two close light sources with different chromaticities. A normal white additive noise was superimposed on the ideal image. The root-mean-square amplitude of the noise was $1/256 \cdot L_{max}$, where L_{max} is the dynamic range of the image. At the stage of computing the watershed map, the image was smoothed with a Gaussian filter with $\sigma = 1.0$. Figure 7, f shows the final result of the region merging process for this image. Threshold values of $\sigma_0 = 1/64 \cdot L_{max}$ and $\sigma_G = 1/32 \cdot L_{max}$ were used during the process.

A gradient descent in the model parameter space was used as the algorithm of detecting linear clusters. The initial estimate for this procedure was obtained by the eigenvector method. It can be seen that segment boundaries follow the boundaries of uniformly colored materials but do not delineate those details caused by variations of object

geometry and shading. For example, the algorithm marks out the whole bottom plane, emphasizing no color shadows on it (see also Fig. 8).

In spite of the fact that, on the whole, the result is good, it is worth paying attention to a segmentation flaw visible at the bottom of the cylinder. Such artifacts periodically appear during the segmentation. This is because the rank of a cluster and its parameters are ill-conditioned for small segments. We think that applying algorithms of the region competition class [1] at the final stage of processing could improve the result.

Fig. 8 shows the result of algorithm testing with an image obtained from a camera with unknown properties. It can be seen that the algorithm yields a reasonable result even for a non-calibrated camera, although it must not run correctly when there is no linear dependence between a spectral stimulus and the corresponding vector stimulus. Further, we are going to accumulate statistics for the performance of the proposed segmentator in the case of processing real images captured by a calibrated camera. Another question of our potential interest is auto-calibration of the camera under the assumption that the scene contains several rank 1 regions.

8. Conclusion

An algorithm of linear color segmentation based on the detailed analysis of the physics of color stimulus formation has been successfully used to segment color images of 3D scenes containing a wide variety of materials, surface geometries and light sources. An important feature of the developed algorithm is that it combines some approaches already proved promising, such as watershed segmentation and region merging, with the physical approach. This ensures that the segmentation process is invariant with respect to geometric factors: the shape of object surfaces and the geometry of illumination. We suppose this approach worthwhile not only because it is more natural for the color segmentation task itself, but also because, due to describing segments through linear submanifolds, the results of such segmentation can be used for solving the problems of color constancy. At present, we are developing a working software package incorporating color segmentation and color constancy algorithms in the same framework based on a generalized linear theory of spectral stimulus formation and Gaussian approximation of spectral functions. One version of such a framework was presented at the 25th European Conference on Visual Perception, Glasgow, Scotland, 2002.

Acknowledgments

This study was supported partly by Russian Foundation Basic Research, grant N 03-04-49161 and by Samsung Advanced Institute of Technology, South Korea.

The authors thank Nikita G. Iroshnikov, Ilja P. Nikolaev and Valentina P. Bozhkova for several helpful conversations and suggestions.

References

1. S. Zhu and A. Yuille, Region competition, *IEEE Trans. Pattern Anal. Mach. Intell.* **18**, 1996, 880-913.

2. P. P. Nikolayev, Some algorithms of recognition of surface coloration, In *Modelling Learning and Behaviour*, pp. 121-151, Nauka, Moscow, 1975. (In Russian).

3. P. P. Nikolayev, Monocular color discrimination of three-dimensional objects in different conditions of illumination, *Biophysics*. **33**, 1988, 148-153. (In Russian and English translation).

4. S. Shafer, Using color to separate reflection components, *Color Res. and Application.* **10**, 1985, 210-218.

5. G. Healey, Using color for geometry-insensitive segmentation, *J. Opt. Soc. Am. A.* **6**, 1989, 920-937.

6. M. Brill, Image segmentation by object color: a unifying framework and connection to color constancy, *J. Opt. Soc. Am. A.* **7**, 1990, 2041-2047.

7. G. Klinker, S. Shafer and T. Kanade, A physical approach to color image understanding, *Int. J. Comp. Vision*. **4**, 1990, 7-38.

8. T. Gevers and A. Smeulders, Color-based object recognition, *Pattern Recognit.* **32**, 1999, 453-464.

9. J.-M. Geusebroek, R. van den Boomgaard, A. Smeulders and H. Geerts, Color invariance, *IEEE Trans. Pattern Anal. Mach. Intell.* **23**, 2001, 1338-1350.

10. F. E. Nicodemus, J. Richmond, J. Hsia, I. Ginsberg and T. Limperis, Geometrical considerations and nomenclature for reflectance, *U.S. Bureau Standarts (U.S.)*, Monograph 160, 1977.

11. D. B. Judd and G. Wyszecki, *Color in business, science and industry*, Wiley, New York, 1975

12. H. Tagare and R. DeFigueiredo, A framework for the construction of reflectance maps for machine vision, *Comput. Vision Graphics Image Process*: *Image Understanding*. **57**, 1993, 265-282.

13. B. Funt and M. Drew, Color space analysis of mutual illumination, *IEEE Trans. Pattern Anal. Mach. Intell.* **15**, 1993, 1319-1326.

14. H.-C. Lee, E. Breneman and C. Shulte, Modeling light reflection for computer color vision, *IEEE Trans. Pattern Anal. Mach. Intell.* **12**, 1990, 402-409.

15. S. Tominaga, Surface identification using the dichromatic reflection model, *IEEE Trans. Pattern Anal. Mach. Intell.* **13**, 1991, 658-670.

16. H.-C. Lee, Method for computing the scene-illuminant chromaticity from specular highlights, *J. Opt. Soc. Am. A.* **3**, 1986, 1694-1699.

17. G. Healey, Segmenting images using normalized color, *IEEE Trans. Syst. Man Cyber.* **22**, 1992, 64-73.

18. J. Liu and Y.-H. Yang, Multiresolution color image segmentation, *IEEE Trans. Pattern Anal. Mach. Intell.* **16**, 1994, 689-700.

19. P. V. C. Hough, Methods and means for recognizing complex patterns, *U.S. Patent 069654*, 1962.

20. D. Ballard, Generalizing the Hough Transform to detect arbitrary shapes, *Pattern Recognit.* **13**, 1981, 111-122.

21. Ch. Nagendra, M. Borah, M. Vishwanath, R. Owens and M. J. Irwin, Edge detection using fine-grain parallelism in VLSI, in *Proceedings, IEEE Int. Conference on Acoustics, Speech and Signal Processing, Minneapolis*, 1993, **1**, pp. 401-404.

22. P. Rousseeuw and A. Leroy, *Robust Regression and Outlier Detection*, Wiley, New York, 1987.

23. J. Gauch, Image segmentation and analysis via multiscale gradient watershed hierarchies, *IEEE Trans. on Image Process.* **8**, 1999, 69-79.

24. K. Harris, S. Efstratiadis, N. Maglaveras and A. Katsaggelos, Hybrid image segmentation using watersheds and fast region merging, *IEEE Trans. on Image Process.* **7**, 1998, 1684-1699.

25. S. Di Zenzo, A note on the gradient of multi-image, *Comput. Vision Graphics Image Process.* **33**, 1986, 116-125.

26. G. Sapiro and D. Ringach, Anisotropic diffusion of multivalued images with applications to color filtering, *IEEE Trans. on Image Process.* **5**, 1996, 1582-1586.

27. D. Marr and E. Hildrith, Theory of edge detection, *Proc. R. Soc. Lond. B.* **207**, 1980, 187-217.

28. X. Wu, Adaptive split-and-merge segmentation based on piecewise least-square approximation, *IEEE Trans. Pattern Anal. Mach. Intell.* **15**, 1993, 808-815.

29. N. Otsu, A threshold selection method from gray-level histograms, *IEEE Trans.Systems, Man, Cybern.*,**9**, 1979, 62-66.

Figures

Fig. 1. Visual stimulus formation for one image pixel. The scene contains two point-like close illuminants with relative spectral distributions $S_1(\lambda)$ and $S_2(\lambda)$. An example of the shape of an integral radiance function, $B_1(\vec{n}_s)$, is shown for the first illuminant. Light flux from the illuminants forms a spectral stimulus, $F(\lambda)$, being scattered by a unichromatic surface with relative spectral distribution of reflectance $\Phi(\lambda)$ and integral BRDF $f(\vec{n}_{in},\vec{n}_{out})$. This stimulus is perceived by a sensor with spectral sensitivities of inputs $\vec{\chi}(\lambda)$. Direction to the sensor from the point of scattering is denoted as $\vec{n}_{sens}$. Then, the sensor produces vector-stimulus $\vec{a}$ as an output.

Fig. 2. Color distributions for simulated and real images. (a) Simulated RGB image of a scene referred to as "Cubes". The scene contains gray-lilac cubes with a yellow-green stripe on a blue-violet mount, illuminated by two close chromatic sources: light-green on the left, purple on the right. The image has numerous color shadows. (b) Image of the same scene, illuminated by achromatic light sources (for comparison). (c) Captured RGB image of a scene referred to as "Akiyo". (d, e) Shape of the RGB color distribution for the "Cubes" image (two projections). 1 – color distribution for the mount part, 2 – color distribution for the basic (gray-lilac) part of the cubes, 3 – color distribution for the stripe. (f, g) Shape of the RGB color distribution for the basic part of the cubes in the "Cubes" image (two projections). (h, i) Color distribution for the whole "Akiyo" image (two projections). 1 – color distribution for woman's face, 2 – color distribution for the jacket, 3 – color distribution for the shirt, 4 – color distribution for the gray screen and the background, 5 – color distribution for the blue screens.

Fig. 3. Examples of the shape of clusters of different ranks in the RGB color space. Thick lines and suns denote clusters, dense dotted lines are parts of corresponding linear submanifolds not belonging to the clusters, scarce dotted line is achromatic axis.

Fig. 4. RGB color space reduction. (a) Projection of clusters of different ranks onto the chromaticity plane (α_c,β_c). (b) Projection of clusters of different ranks onto the hue semicircle φ_c. Here: 1 – rank 0_c cluster projection, 2 – rank 2_a cluster projection, 3 – rank 1_c cluster projection, 4 – rank 1_b cluster projection, 5 – rank 2_{ab} or rank 1_a cluster projection.

Fig. 5. Effect of noise spikes on the accuracy of determining the model parameters. The original distribution consists of 7 patches, one of those being noise spike. The model of the distribution is a line arbitrarily positioned on a plane. (a) Original distribution (patch 1 is a noise spike) and the solution obtained with the eigenvector method

(gray line). (b) Hough Transform of the original distribution. The coordinates of the parameter space are φ and ρ (each point of the space corresponds to the line characterized by slope angle φ and distance ρ from the coordinate origin). The numbers mark the components of the distribution in the model parameter space, produced by the corresponding patches. The cross marks the maximum of the vote function. (c) Solution obtained with the Hough Transform method (gray line). The line found corresponds to the maximum of the vote function.

Fig. 6. Example of segmentation of simulated images by the simplified algorithm. (a) Original image synthesized by ray tracing: a neutral bichromatic brown sphere above a gray plane illuminated by a close achromatic light source (ranks 2_{ab}, 0_a). The image size is 256*256 pixels. (b) The result of image segmentation by the algorithm based on hue analysis. (c) The same scene with an additional bluish diffuse light source (ranks 2_a, 0_c). (d) The result of image segmentation by the same algorithm.

Fig. 7. Example of segmentation of a simulated image. (a) Original image synthesized by ray tracing. The image size is 256*256 pixels. (b) The result of watershed segmentation. (c-e) Preliminary results of segmentation (after stages 2, 4, and 6 of the main algorithm, respectively). Those boundaries whose weight exceeds σ_G are marked with black color. (f) The final result of segmentation.

Fig. 8. Example of segmentation of a captured image. (a) Original image: yellow and blue wool clews and a fabric ball with sectors of different colors (red, blue, and yellow) on a white table-cloth. The table-cloth has sharp folds looking like contrasting linear stripes. The scene is illuminated by a bluish light source located at upper left. The image size is 512*410 pixels. (b) The result of watershed segmentation (after smoothing with $\sigma = 3.0$). (c) The final result of segmentation.

Table 1

Examples of scenes of different ranks for the 3D CSp

Rank	Scene examples
0_b	Completely shaded object
0_a	Diffuse achromatic source[1]; convex achromatic object[2] illuminated by an achromatic diffuse source; flat small-aperture achromatic object illuminated by a distant achromatic source
0_c	Diffuse chromatic source; convex object illuminated by a diffuse chromatic source; flat small-aperture chromatic object illuminated by a distant achromatic source
1_b	Convex unichromatic object illuminated by a close chromatic source or by achromatic sources, among which at least one is not diffuse
1_a	Convex achromatic object illuminated by a close chromatic source and by an achromatic diffuse source
1_{ab}	Achromatic object illuminated by achromatic sources, among which at least one is not diffuse (with taking into account interreflections of any order)
1_c	Convex unichromatic object illuminated by a close chromatic source and by a diffuse chromatic source
2_b	Convex unichromatic object illuminated by two close chromatic sources; convex bichromatic object illuminated by a close chromatic source; convex unichromatic object illuminated by a close chromatic source and an interreflctions from another convex unichromatic object (with taking into account the first-order interreflections)
2_a	Convex unichromatic object illuminated by two close chromatic sources and a diffuse source; convex neutral bichromatic[3] object illuminated by a close chromatic source and a diffuse source
2_{ab}	Convex achromatic object illuminated by a close chromatic source and a close achromatic source; convex neutral bichromatic object illuminated by achromatic sources, among which at least one is not diffuse

[1] Here, the achromatic source is an illuminant with $S(\lambda) \approx const$ for the whole spectral range the sensor is capable to detect.

[2] Here, the achromatic object is a uniformly colored unichromatic object with $\Phi(\lambda) \approx const$ for the whole spectral range the sensor is capable to detect.

[3] Here, the neutral bichromatic object is a uniformly colored bichromatic object with $\Phi_1(\lambda) \approx const$ for the whole spectral range the sensor is capable to detect.

Table 2

Rank transformation upon reduction of CSp dimension

Rank in (R, G, B) space	Rank in (α_c, β_c) space	Rank in (φ_c) space
0_c, 1_b	0	0
1_a, 2_{ab}	1	0
0_a, 1_{ab}	0	-
1_c, 2_b	1	-
0_b, 2_a	-	-

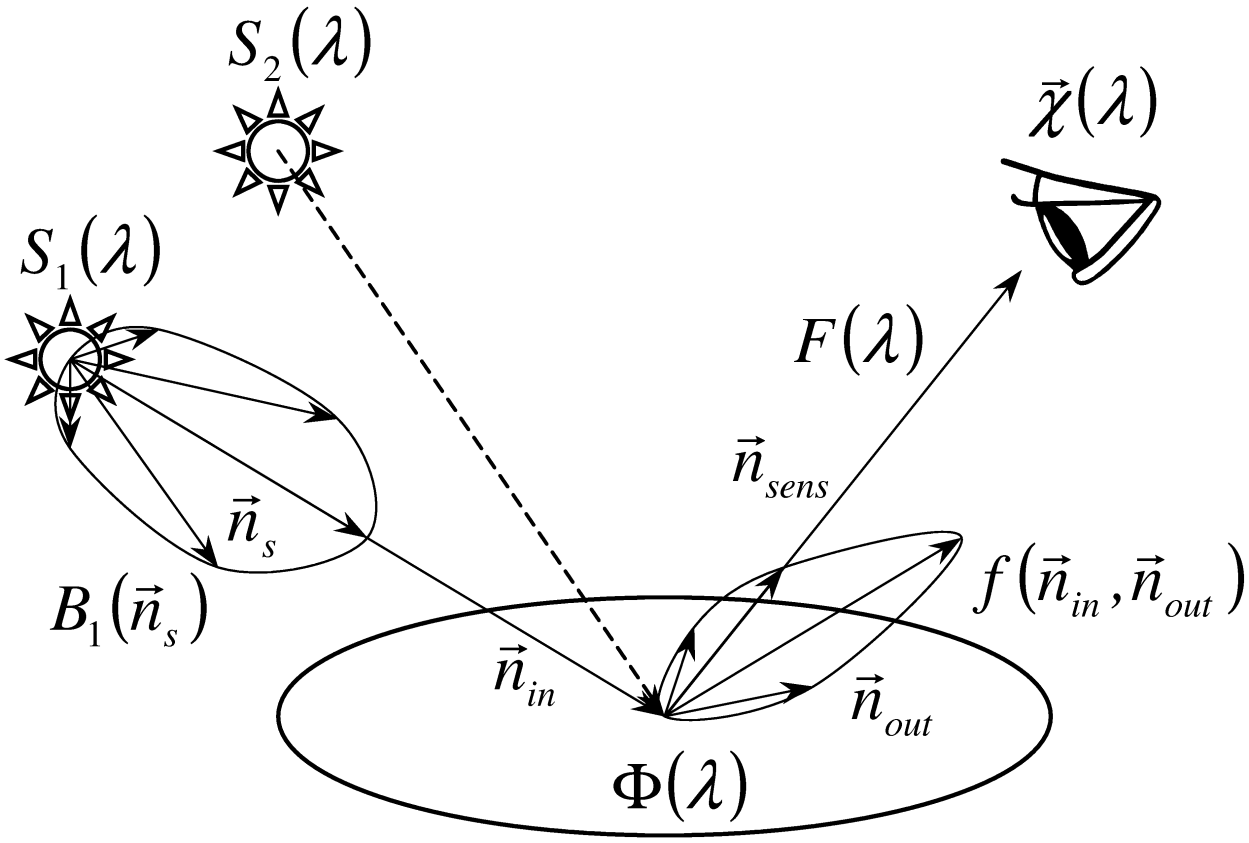
$S_2(\lambda)$
$\vec{\chi}(\lambda)$
$S_1(\lambda)$
$F(\lambda)$
$\vec{n}_{sens}$
$\vec{n}_s$
$B_1(\vec{n}_s)$
$f(\vec{n}_{in},\vec{n}_{out})$
$\vec{n}_{in}$
$\vec{n}_{out}$
$\Phi(\lambda)$

Fig. 1.

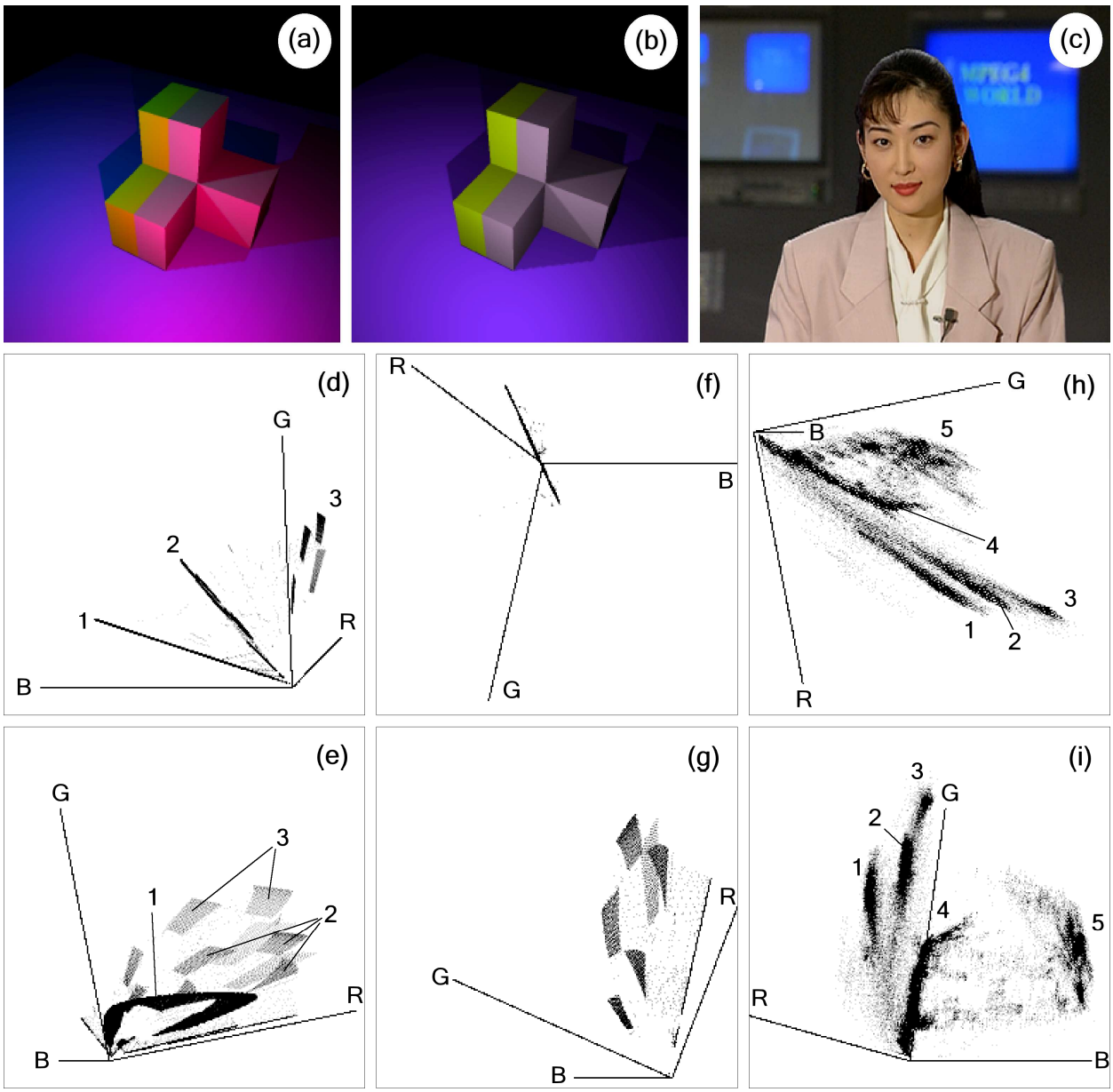

Fig. 2.

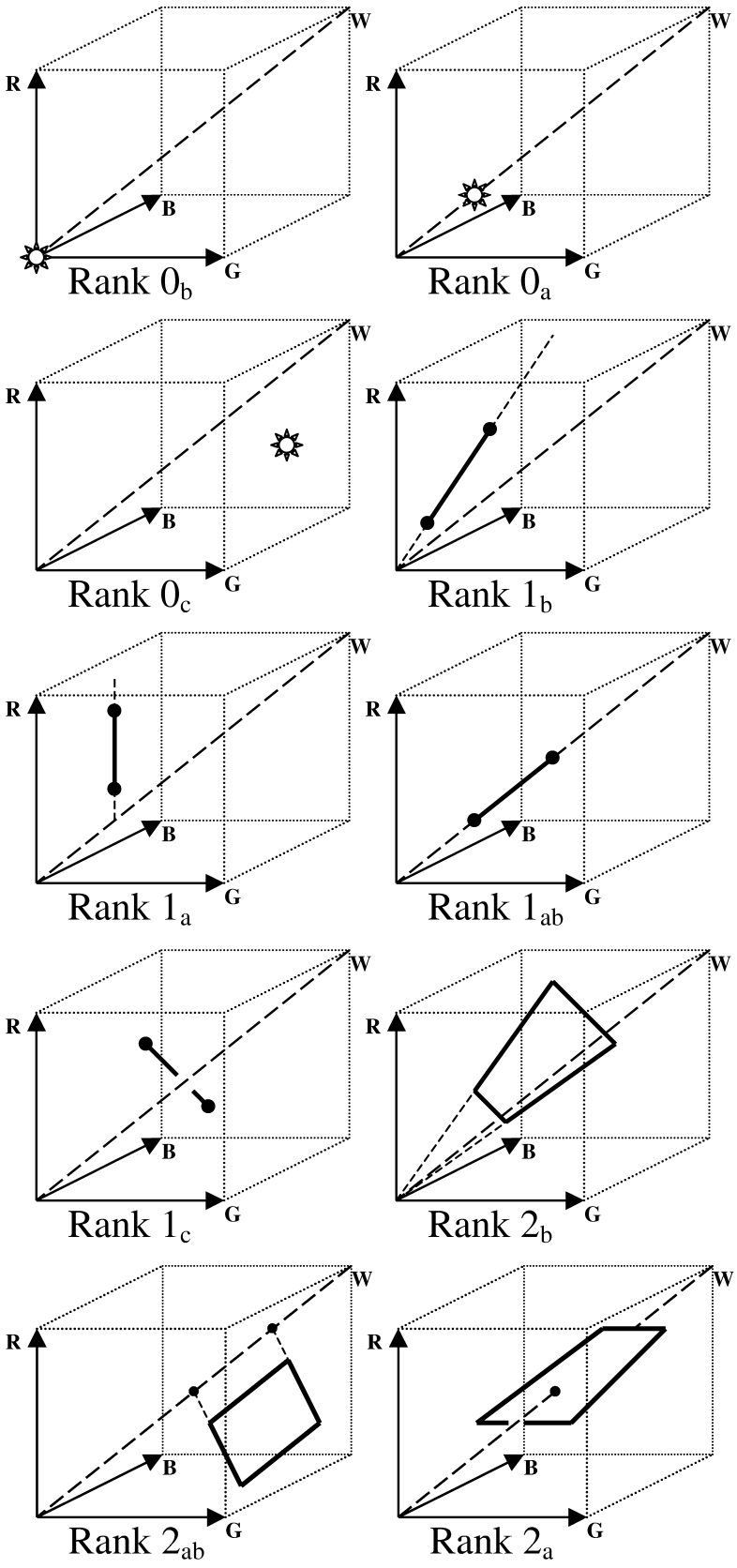
R
W
B
G
Rank 0_b
Rank 0_a
Rank 0_c
Rank 1_b
Rank 1_a
Rank 1_{ab}
Rank 1_c
Rank 2_b
Rank 2_{ab}
Rank 2_a

Fig. 3.

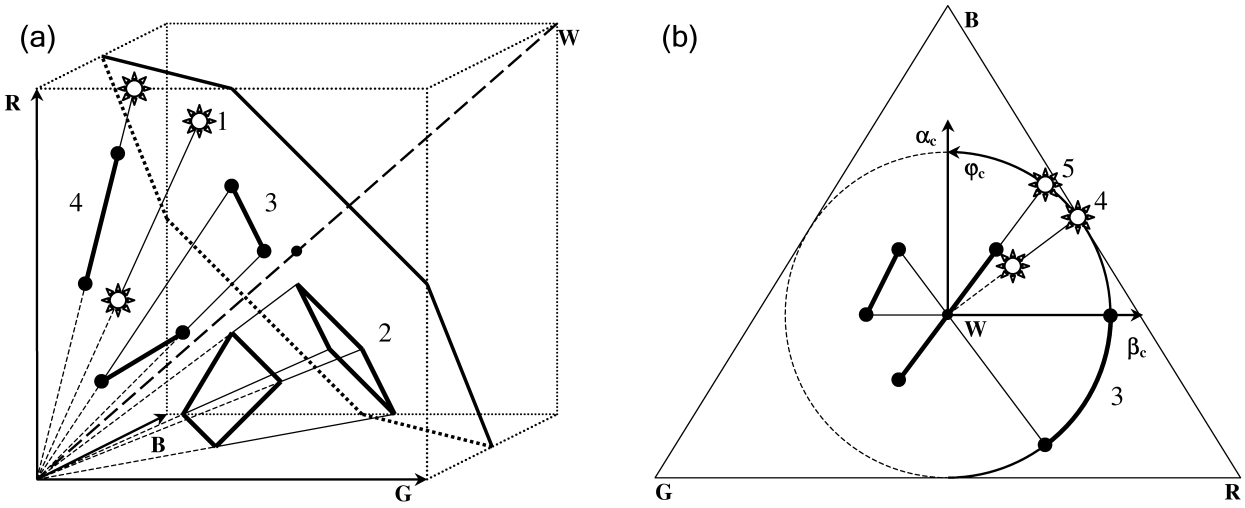

Fig. 4.

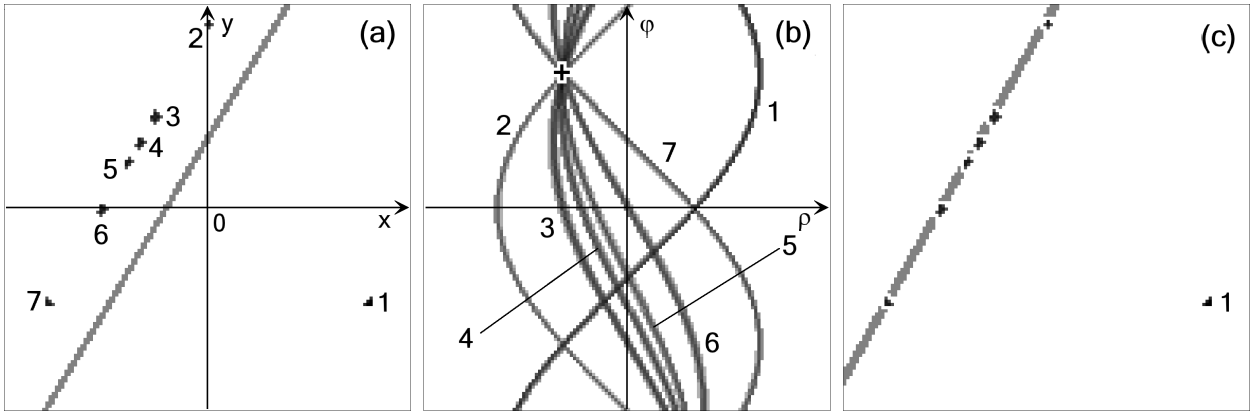

Fig. 5.

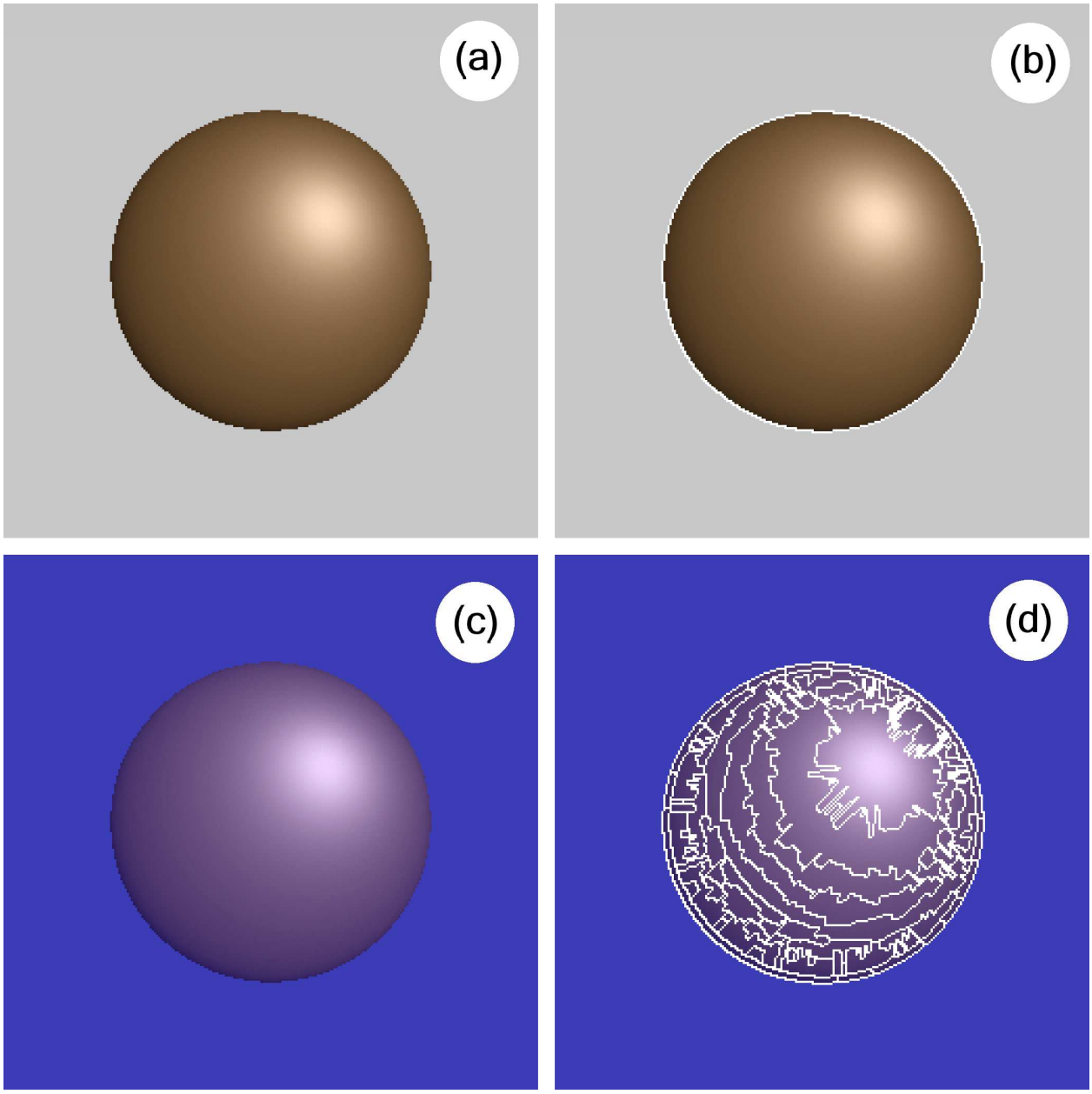

Fig. 6.

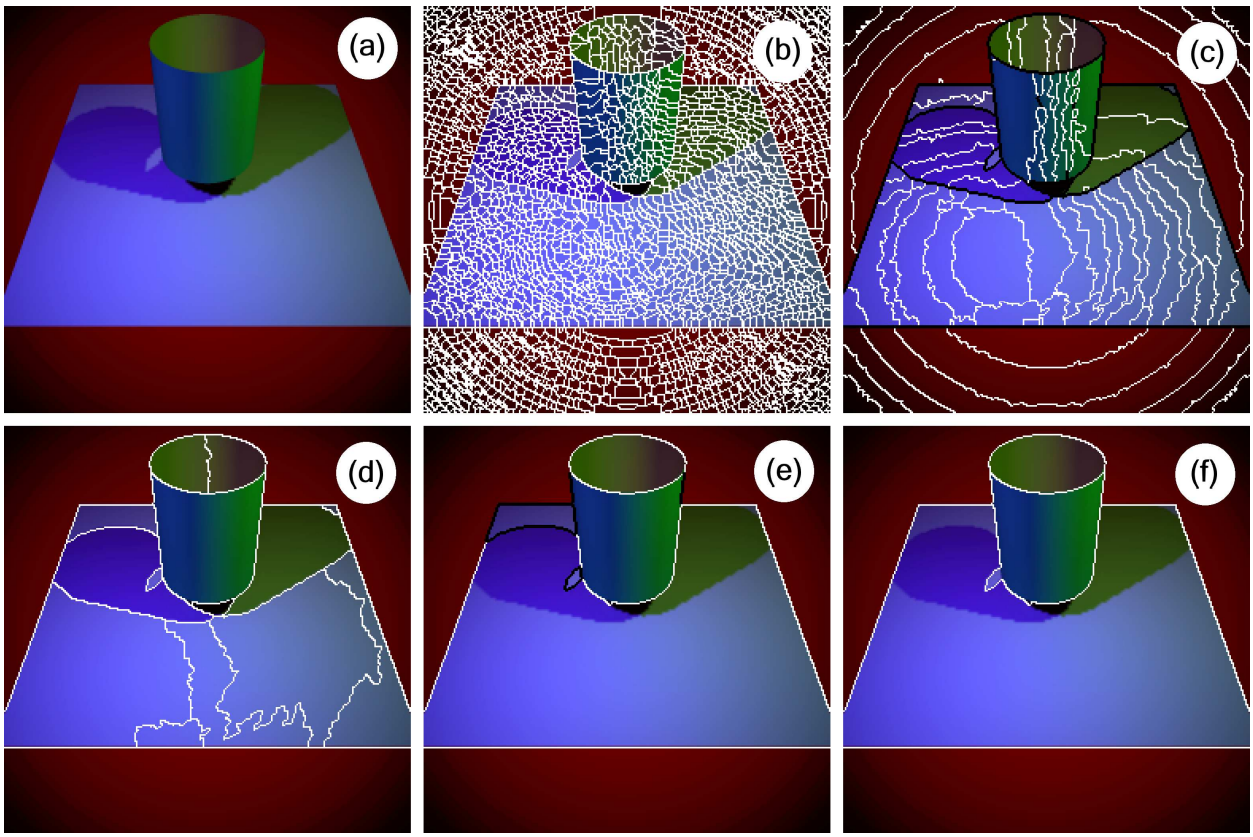

Fig. 7.

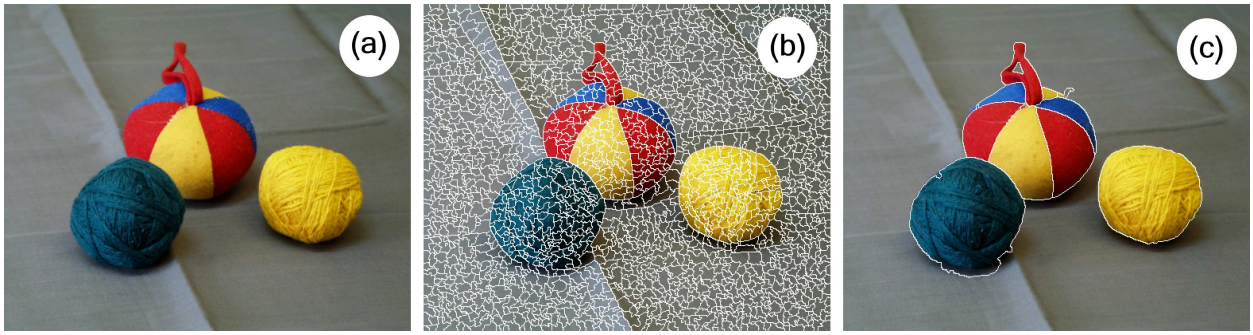

Fig. 8.