

Illumination Color and Intrinsic Surface Properties —Physics-based Color Analyses from a Single Image

ROBBY T. TAN[†] and KATSUSHI IKEUCHI^{††}

In the real world, the color appearances of objects are generally not consistent. It depends principally on two factors: illumination spectral power distribution (illumination color) and intrinsic surface properties. Consequently, to obtain objects' consistent color descriptors, we have to deal with those two factors. The former is commonly referred to as color constancy: a capability to estimate and discount the illumination color, while the latter is identical to the problem of recovering body color from highlights. This recovery is crucial because highlights emitted from opaque inhomogeneous objects can cause the surface colors to be inconsistent with regard to the change of viewing and illumination directions. We base our color constancy methods on analyzing highlights or specularities emitted from opaque inhomogeneous objects. We have successfully derived a linear correlation between image chromaticity and illumination chromaticity. This linear correlation is clearly described in inverse-intensity chromaticity space, a novel two-dimensional space we introduce. Through this space, we become able to effectively estimate illumination chromaticity (illumination color) from both uniformly colored surfaces and highly textured surfaces in a single integrated framework, thereby making our method significantly advanced over the existing methods. Meanwhile, for separating reflection components, we propose an approach that is based on an iterative framework and a specular-free image. The specular-free image is an image that is free from specularities yet has different body color from the input image. In general, the approach relies principally on image intensity and color. All methods of color constancy and reflection-components separation proposed in this paper are analyzed based on physical phenomena of the real world, making the estimation more accurate, and have strong basics of analysis. In addition, all methods require only a single input image. This is not only practical, but also challenging in term of complexity.

1. Introduction

The color appearance of an object is not the object's actual color. Several factors, mainly illumination and object surface's intrinsic properties, play significant roles in determining the object color appearance. In our daily life, we can easily find the roles of illumination color in many occasions, for instance, an outdoor scene under a clear sky will look redder in the evening than in the middle of the day, or an object will look greener if lit by a green lamp. However, although the color appearance of an object or a scene changes as a consequence of illumination change, we are still, at a certain level of accuracy, able to identify their actual color. This capability is called color constancy. It is inherent in human perception and one of the important aspects of object recognition processes.

In machine vision, color constancy is also a crucial requirement for various applications, e.g., color-based object recognition, color reproduction, image retrieval, reflection components

separation, image-based rendering, and so on. Unfortunately, up to now, the mechanism of human perception color constancy has not been well understood, making it impossible to be applied to machine vision. For decades, this has motivated researchers in machine vision to develop various color constancy algorithms, which do not necessarily correspond to human biological color constancy.

Generally, color constancy is defined as the capability to recover the actual color of an object. It implies that, although the illumination color changes, we can obtain a consistent color descriptor of the object. This consistency is the most fundamental aspect of color constancy. However, while it is correct for diffuse objects, the consistency is still partially correct for certain types of objects that exhibit highlights. In diffuse objects, if we have discounted the illumination color and obtained their actual color, we will have a consistent color descriptor even if either our viewing position or the illumination direction changes. On the contrary, for objects exhibiting highlights, although we have discounted the illumination color, the colors of certain patches of the objects are still

[†] Australian National University

^{††} The University of Tokyo

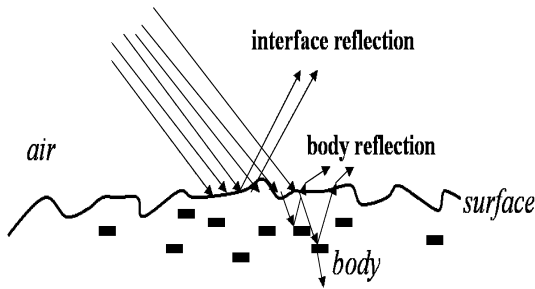


Fig. 1 The mechanism of reflected light on inhomogeneous opaque surface.

inconsistent w.r.t. the change of viewing and illumination directions. The reason is, the locations of highlights, which are caused by the presence of specular reflection, are inconsistent w.r.t. the changes of viewing and illumination directions. As a consequence, color constancy alone is insufficient to acquire a consistent color descriptor of general types of objects. For this reason, in this paper, instead of dealing solely with illumination color, we also deal with object surface intrinsic properties.

Based on its reflection components, basically intrinsic surface properties can be divided into two components: diffuse (body) and specular (interface) reflections. **Figure 1** shows a pictorial mechanism of reflected light rays. Once a bundle of light rays enters an inhomogeneous opaque surface, some of the rays will immediately reflect back into the air, while the remainder will penetrate the body of the object. Some of these penetrating light rays will go through the body; others will reflect back onto the surface and then into the air. The immediately reflected light rays are called *interface* or *specular* reflection, while those that have penetrated and then reflected back into the air are called *body* or *diffuse* reflection. Note that, besides those two reflections, physically there is another component called *specular spike*^{3),32)}. However, since its presence is very minor in inhomogeneous object, we can ignore it. Thus, highlights emitted from inhomogeneous objects are the combination of diffuse and specular reflections. Unlike diffuse reflection, the location of specular reflection depends on viewing and illumination directions, causing its appearance to be inconsistent. On the contrary, diffuse reflection is independent from viewing position, and dependent only on illumination direction in term of its intensity magnitude. This means that the color descriptor of diffuse reflection,

which is usually a normalized value, is independent of both viewing position and illumination directions. As a consequence, to be able to obtain a consistent color descriptor, we have to decompose or separate the reflection components and then acquire diffuse only reflection. Moreover, once we acquire diffuse only reflection, we become able to observe the body color beneath highlights.

1.1 Goals

Considering the importance in various machine vision applications, therefore, the ultimate purpose of this paper is to describe how to extract the actual color of diffuse reflection components. Basically, two processes are required to achieve our purpose, namely, color constancy and reflection components separation. We base our color constancy methods on analyzing highlights or specularities emitted from opaque inhomogeneous objects. We have successfully derived a linear correlation between image chromaticity and illumination chromaticity. This linear correlation is clearly described in inverse-intensity chromaticity space, a novel two-dimensional space which we introduce. Through this space, we become able to effectively estimate illumination chromaticity (illumination color) from both uniformly colored surfaces and highly textured surfaces in a single integrated framework, thereby making our method significantly more advanced than the existing methods. Moreover, unlike the existing methods based on specularities, thanks to the linear correlation, we do not need to segment surface colors beneath the highlights. Meanwhile, for separating reflection components, we propose an approach based on intensity and color differences between highlights and diffuse reflections.

In general, the flow of our framework can be depicted in **Fig. 2**. Top of the Fig. 2 (a) shows an opaque inhomogeneous object lit by an incandescent lamp. By using our proposed color constancy method, we estimate the illumination color and then normalize the image, making the illumination color becomes pure-white as shown in Fig. 2 (b). Then, after normalizing the image, we decompose it into its reflection components. Figures 2 (c)–(d) shows the decomposition results: diffuse reflection component and specular reflection component, respectively. All approaches of color constancy and reflection-components separation in this paper are analyzed based on physical phenomena of

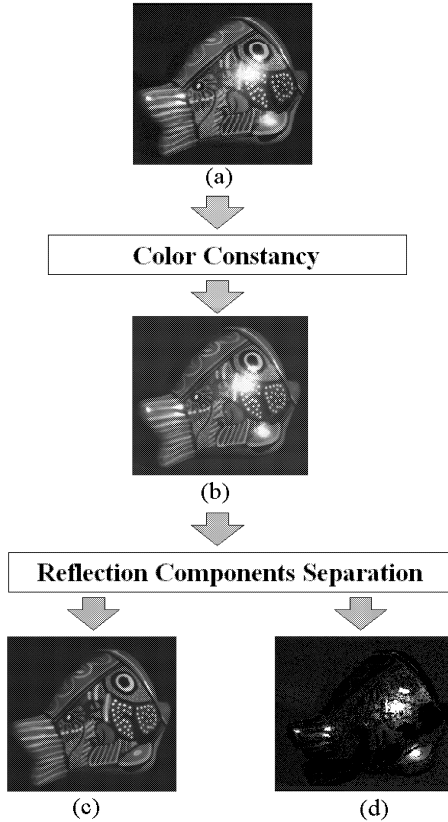


Fig. 2 The flow of our framework to extract actual body color of an object. (a) Input image lit with unknown illumination. (b) Color constancy result, transforming unknown illumination color into pure-white illumination. (c) Diffuse only reflection, which is able to produce a consistent color descriptor of the object. (d) Specular only reflection, conceiving important properties such as surface roughness.

the real world, making the computation more accurate and have strong basics of analysis. In addition, for all approaches, we require only a single image as the input.

1.2 Previous Work

1.2.1 Color Constancy

Finlayson, et al.¹¹⁾ categorized color constancy methods into two classes: statistics-based and physics-based methods. Statistics-based methods utilize the relationship between color distributions and statistical knowledge of common lights and surfaces^{4),7),9),34),41),43)}. One drawback of these methods is that they require many colors to be observed on the target surfaces. On the other hand, physics-based methods^{6),8),14),23),24)}, which base their algorithms on understanding the physical process of reflected light, can successfully deal

with fewer surface colors, even to the extreme of a single surface color^{11),12)}. In addition, based on the surface type of the input image, physics-based methods can be divided into two groups: diffuse-based and dichromatic-based methods. Diffuse-based methods assume that input images have only diffuse reflection, while dichromatic-based methods assume both diffuse and specular reflections occur in the images. Geusebroek, et al.^{16),17)} proposed a physical basis of color constancy by considering the spectral and spatial derivatives of the Lambertian image formation model. Andersen, et al.¹⁾ provided an analysis on image chromaticity under two illumination colors for dichromatic surfaces. Since our aim is to develop an algorithm that is able to handle both a single and multiple surface colors, in this section, we will concentrate our discussion on existing physics-based methods, particularly dichromatic-based methods.

Methods in dichromatic-based color constancy rely on the dichromatic reflection model proposed by Shafer³⁶⁾. Klinker, et al.²¹⁾ introduced a method to estimate illumination color from a uniformly colored surface, by extracting a T-shaped color distribution in the RGB space. However, in real images, it becomes quite difficult to extract the T-shape due to noise, making the final estimate unreliable.

Lee²³⁾ introduced a method to estimate illumination chromaticity using highlights of at least two surface colors. The estimation is accomplished by finding an intersection of two or more dichromatic lines in the chromaticity space. While this simple approach based on the physics of reflected light provides a handy method for color constancy, it suffers from a few drawbacks. First, to create the dichromatic line for each surface color from highlights, one needs to segment the surface colors underneath the highlights. This color segmentation is difficult when the target object is highly textured. Second, nearly parallel dichromatic lines caused by similar surface colors can make the intersection sensitive to noise. Consequently, for real images, which usually suffered from noise, the estimation for similar surface colors becomes unstable. Third, the method does not deal with uniformly colored surfaces. Parallel to this, several methods have been proposed in the literature^{6),40),42)}.

Recently, three methods have been proposed which extend Lee's algorithm²³⁾: Lehmann, et

al.²⁸⁾ developed a more robust technique to identify the dichromatic lines in the chromaticity space. The success of this technique depends on an assumption that, in each highlight region, the surface color is uniform. As a consequence, the technique fails when dealing with complex textured surfaces, which usually have more than one surface color in their highlight regions. Finlayson, et al.¹⁰⁾, proposed imposing a constraint on the colors of illumination. This constraint is based on the statistics of natural illumination colors, and improves the stability in obtaining the intersection, i.e., it addresses the second drawback of Lee's method. Furthermore, Finlayson, et al.¹¹⁾ proposed the use of the Planckian locus as a constraint to accomplish illumination estimation from uniformly colored surfaces. This Planckian constraint on the illumination chromaticity makes the estimation more robust, especially for natural scene images. However, the method still has a few drawbacks. First, the position and the shape of the Planckian locus in the chromaticity space make the estimation error prone for certain surface colors, such as blue or yellow color. Second, as they include diffuse regions in obtaining dichromatic lines, the result could become inaccurate. While the fact that their method does not require reflection separation is one of the advantages, the diffuse cluster, due to noise, usually has a different direction from the specular cluster; as a result, the dichromatic line can be shifted from the correct one. Third, like other previous methods, for multicolored surfaces, color segmentation is required.

1.2.2 Reflection Components Separation

Many works also have been developed for separating reflection components. Wolff, et al.⁴⁵⁾ used a polarizing filter to separate reflection components from gray images. The main idea of their method is that, for most incident angles, diffuse reflections tend to be less polarized than the specular reflections. Nayar, et al.³¹⁾ extended this work by considering colors instead of using the polarizing filters alone. They identified specular pixels and the illumination color vector in RGB space by using intensity variation produced by a polarizing filter. A specular pixel, which is partially composed of a specular reflection component, will have a different intensity if the polarization angle of the filter is changed. The combination of polarizing filter and colors is even for textured surfaces;

however, utilizing such an additional filter is impractical in some circumstances. Sato, et al.³⁵⁾ introduced a four-dimensional space, temporal-color space, to analyze the diffuse and specular reflections based on colors and image intensity. While this method has the ability to separate the reflection components locally, since each location contains information of diffuse and specular reflections, it requires dense input images with variation of illuminant directions. Lee, et al.^{26),27)} introduced color histogram differencing to identify specularities. The key idea is that colors of diffuse pixels are independent of the changing of viewing positions, while colors of specular pixels are dependent on it. They transform the pixels of images taken from different viewing directions into RGB space, and then identify the specular pixels. Later, Lin, et al.²⁹⁾ extended this method by adding multi-baseline stereo. Criminisi, et al.⁵⁾ developed an Epipolar Plane Image (EPI)-based method to detect specularities. They found that in two-dimensional spatio-temporal space, highlights' straight lines have larger gradients than diffusers' straight lines. Lin, et al.³⁰⁾, unlike previous methods, introduced a method using sparse images (at least two images) under different illumination positions. They proposed an analytical method that combines the finite dimensional basis functions³³⁾ and a dichromatic model to form a closed form equation, by assuming that the sensor sensitivity is narrow-band. This method can separate the reflection component locally.

The aforementioned methods are considerably effective in separating reflection components; however, for many applications, using multiple images is impractical. Shafer³⁶⁾, who introduced the dichromatic reflection model, was one of the early researchers who used a single colored image. He proposed a separation method based on parallelogram distribution of colors in RGB space. Klinker, et al.²¹⁾ then extended this method by introducing a T-shaped color distribution. This color distribution represents body and illumination color vectors. By separating these vectors, the reflection equation becomes a closed form equation and directly solvable. Unfortunately, for many real images, this T shape is hardly extractable due to noise, etc. Bajscy, et al.²⁾ proposed an approach that introduced a three dimensional space composed of lightness, saturation and hue. In their method, the input

image has to be neutralized to pure-white illumination using a linear basis functions operation. For every neutralized pixel, the weighting factors of the surface reflectance basis functions are projected into the three-dimensional space, where specular and diffuse reflections are identifiable due to the difference of their saturation values.

1.3 Overview

The rest of this paper is organized as follows: in Section 2, we discuss the reflection model used in all methods proposed in this paper. In Section 3, we will explain the derivation and detail algorithm of the proposed color constancy method. In Section 4, we will focus the discussion on the method of separating reflection components. A number of experimental results using real images will be shown in Section 5. Finally, in Section 6, we offer several conclusions.

2. Reflection Model

2.1 Image Formation

Most inhomogeneous objects, such as those made of plastics, acrylics, etc., exhibit both diffuse and specular reflections. The diffuse reflection is due to the varying refractive indices in the objects' surfaces and bodies, while the specular reflection is mainly due to the refractive index difference between objects' surfaces and the air. Considering these two reflection components, Shafer³⁶⁾ introduced the dichromatic reflection model, which states that reflected lights of inhomogeneous objects are linear combinations of diffuse and specular reflection components. As a result, an image's pixel of inhomogeneous objects taken by a digital color camera can be described as:

$$\mathbf{I}(\mathbf{x}) = w_d(\mathbf{x}) \int_{\Omega} S(\lambda, \mathbf{x}) E(\lambda) \mathbf{q}(\lambda) d\lambda \quad (1)$$

$$+ w_s(\mathbf{x}) \int_{\Omega} E(\lambda) \mathbf{q}(\lambda) d\lambda$$

where $\mathbf{I} = \{I_r, I_g, I_b\}$ is the color vector of image intensity or camera sensor. The spatial parameter, $\mathbf{x} = \{x, y\}$, is the two dimensional image coordinates. $\mathbf{q} = \{q_r, q_g, q_b\}$ is the three-element-vector of sensor sensitivity. $w_d(\mathbf{x})$ and $w_s(\mathbf{x})$ are the weighting factors for diffuse and specular reflection, respectively; their values depend on the geometric structure at location \mathbf{x} . $S(\mathbf{x}, \lambda)$ is the diffuse spectral reflectance function, while $E(\lambda)$ is the spectral power distribution function of illumination. $E(\lambda)$ is in-

dependent of the spatial location (\mathbf{x}) because we assume a uniform illumination color. The integration is done over the visible spectrum (Ω). Note that we ignore the camera gain and camera noise in the above model, and assume that the model follows the neutral interface reflection (NIR) assumption²⁵⁾, i.e., the color of specular reflection component equals the color of the illumination. For the sake of simplicity, Eq. (1) can be written as:

$$\mathbf{I}(\mathbf{x}) = w_d(\mathbf{x}) \mathbf{B}(\mathbf{x}) + w_s(\mathbf{x}) \mathbf{G} \quad (2)$$

where $\mathbf{B}(\mathbf{x}) = \int_{\Omega} S(\lambda, \mathbf{x}) E(\lambda) \mathbf{q}(\lambda) d\lambda$, and $\mathbf{G} = \int_{\Omega} E(\lambda) \mathbf{q}(\lambda) d\lambda$. The first part of the right side of the equation represents the diffuse reflection component, while the second part represents the specular reflection component.

2.2 Chromaticity

Besides the dichromatic reflection model, we also use chromaticity or *normalized rgb*, which is defined as:

$$\boldsymbol{\sigma}(\mathbf{x}) = \frac{\mathbf{I}(\mathbf{x})}{I_r(\mathbf{x}) + I_g(\mathbf{x}) + I_b(\mathbf{x})} \quad (3)$$

where $\boldsymbol{\sigma} = \{\sigma_r, \sigma_g, \sigma_b\}$. Based on the equation, for the diffuse only reflection component ($w_s = 0$), the chromaticity will be independent from the diffuse weighting factor w_d . We call this *diffuse chromaticity* ($\boldsymbol{\Lambda}$) with definition:

$$\boldsymbol{\Lambda}(\mathbf{x}) = \frac{\mathbf{B}(\mathbf{x})}{B_r(\mathbf{x}) + B_g(\mathbf{x}) + B_b(\mathbf{x})} \quad (4)$$

where $\boldsymbol{\Lambda} = \{\Lambda_r, \Lambda_g, \Lambda_b\}$. On the other hand, for the specular only reflection component ($w_d = 0$), the chromaticity will be independent from the specular weighting factor (w_s), and we call it *specular or illumination chromaticity* ($\boldsymbol{\Gamma}$):

$$\boldsymbol{\Gamma} = \frac{\mathbf{G}}{G_r + G_g + G_b} \quad (5)$$

where $\boldsymbol{\Gamma} = \{\Gamma_r, \Gamma_g, \Gamma_b\}$. Consequently, with regard to Eqs. (4) and (5), Eq. (2) becomes able to be written in term of chromaticity:

$$\mathbf{I}(\mathbf{x}) = m_d(\mathbf{x}) \boldsymbol{\Lambda}(\mathbf{x}) + m_s(\mathbf{x}) \boldsymbol{\Gamma} \quad (6)$$

where

$$m_d(\mathbf{x}) = w_d(\mathbf{x}) [B_r(\mathbf{x}) + B_g(\mathbf{x}) + B_b(\mathbf{x})]$$

$$m_s(\mathbf{x}) = w_s(\mathbf{x}) (G_r + G_g + G_b) \quad (7)$$

As a result, we have three types of chromaticity: image chromaticity ($\boldsymbol{\sigma}$), diffuse chromaticity ($\boldsymbol{\Lambda}$) and illumination chromaticity ($\boldsymbol{\Gamma}$). The image chromaticity is directly obtained from the

input image using Eq. (3). In addition, without loss of generality, we can have $(\sigma_r + \sigma_g + \sigma_b) = (\Lambda_r + \Lambda_g + \Lambda_b) = (\Gamma_r + \Gamma_g + \Gamma_b) = 1$.

Based on the dichromatic reflection model and chromaticities definitions derived above, we describe our goal: given image intensities $(\mathbf{I}(\mathbf{x}))$ whose illumination chromaticity $(\mathbf{\Gamma})$ is estimated by a color constancy method; we intend to decompose them into their reflection components: $m_d(\mathbf{x})\mathbf{\Lambda}(\mathbf{x})$ and $m_s(\mathbf{x})\mathbf{\Gamma}$.

3. Color Constancy

3.1 Inverse-Intensity Chromaticity Space

By substituting each color channel's image intensity in Eq. (3) with its definition in Eq. (6) and considering pixel-based operation, the image chromaticity can be written in terms of dichromatic reflection model:

$$\sigma = \frac{m_d\mathbf{\Lambda} + m_s\mathbf{\Gamma}}{m_d[\Lambda_r + \Lambda_g + \Lambda_b] + m_s[\Gamma_r + \Gamma_g + \Gamma_b]} \quad (8)$$

Since $(\Lambda_r + \Lambda_g + \Lambda_b) = (\Gamma_r + \Gamma_g + \Gamma_b) = 1$, we can obtain the correlation between m_s and m_d :

$$m_s = m_d \frac{(\mathbf{\Lambda} - \sigma)}{(\sigma - \mathbf{\Gamma})} \quad (9)$$

Then, by plugging Eq. (9) into Eq. (6), the correlation between image intensity (\mathbf{I}) and image chromaticity (σ) can be described as:

$$\mathbf{I} = m_d(\mathbf{\Lambda} - \mathbf{\Gamma}) \left(\frac{\sigma}{\sigma - \mathbf{\Gamma}} \right) \quad (10)$$

The last equation shows that the correlation between image intensity (\mathbf{I}) and image chromaticity (σ) is not linear. Consequently, by projecting a uniformly colored surface into chromaticity-intensity space, the specular pixels will form a curved cluster (non-linear correlation), as illustrated in **Fig. 3** (b). On the other hand, the diffuse pixels will form a straight vertical line, since their image chromaticity (σ) which equals to their diffuse chromaticity $(\mathbf{\Lambda})$ is independent from image intensity (\mathbf{I}) .

3.2 Image Chromaticity and Illumination Chromaticity

By introducing $\mathbf{p} = \{p_r, p_g, p_b\}$ which we define as $\mathbf{p} = m_d(\mathbf{\Lambda} - \mathbf{\Gamma})$, we can derive from Eq. (10) that:

$$\frac{\mathbf{I}}{\sigma} = \frac{\mathbf{p}}{\sigma - \mathbf{\Gamma}} \quad (11)$$

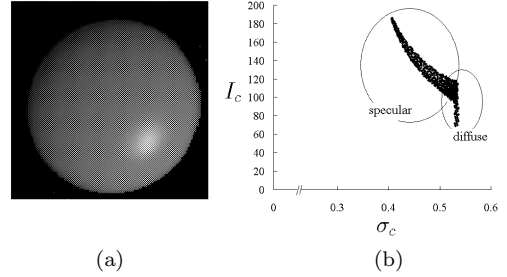


Fig. 3 (a) Synthetic image with a single surface color. (b) Projection of the diffuse and specular pixels into the chromaticity-intensity space, with index c representing g (the green channel).

Since $\mathbf{I}/\sigma = \Sigma I_i$, where $\Sigma I_i = (I_r + I_g + I_b)$, then the correlation between image chromaticity and illumination chromaticity becomes:

$$\sigma = \mathbf{p} \frac{1}{\Sigma I_i} + \mathbf{\Gamma} \quad (12)$$

This equation is the core of our method. It shows that by solely calculating the value of \mathbf{p} , we are able to determine the illumination chromaticity $(\mathbf{\Gamma})$, since image chromaticity (σ) and total image intensity (ΣI_i) can be directly observed from the input image. Moreover, based on the equation we can solve the illumination estimation independently for each color channel, which is expressed as:

$$\sigma_c = p_c \frac{1}{\Sigma I_i} + \Gamma_c \quad (13)$$

where index c represents one of the three color channels $(\{r, g, b\})$ we want to estimate. The details are as follows.

If the values of p_c are constant and the values of ΣI_i vary throughout the image, the last equation becomes a linear equation, and the illumination chromaticity (Γ_c) can be estimated in a straightforward manner by using general line fitting algorithms for each color channel. However, in most images, the values of p_c are not constant, since p_c depends on m_d , Λ_c and Γ_c . For the sake of simplicity, until the end of this section, we temporarily assume that the values of Λ_c are constant, making the values of p_c depend solely on m_d , as Γ_c has already been assumed to be constant.

Equation (7) states that $m_d = w_d(B_r + B_g + B_b)$. According to the Lambert's Law²², w_d is determined by the angle between lighting direction and surface normal, while $(B_r + B_g + B_b)$ is determined by diffuse albedo and intensity of incident light (L) . For a surface with a uni-

form color, the value of the diffuse albedo is constant. The angles between surface normals and light directions depend on the shape of the object and the light distribution. The angle will be constant if an object has planar surface and illumination directions are the same for all points in the surface. While, if the surface is not planar or the illumination directions are not uniform, then the angle will vary. The values of intensity of incident light (L) are mostly determined by the location of illuminants, which will be constant if the locations of the illuminants are distant from the surface. For relatively nearby illuminants, the values of L may vary w.r.t. the surface point. Considering all these aspects, as a result, in general conditions the value of m_d can be either constant or varied. Yet, in most cases the value of m_d will be varied because, most shapes of objects in the real world are not planar and the assumption on uniform illumination direction, in some conditions, cannot be held.

Consequently, Eq. (13) poses two problems: first, whether there are a number of specular pixels that have the same m_d , and second, whether these pixels that have the same m_d also have different ΣI_i . If we consider a single surface color, then the solution of the first problem depends on w_d and L . In microscopic scale of the real world, the combination of w_d and L could be unique. Fortunately, in the scale of image intensity, for some set of surface points, the differences of the combination of w_d and L are small and can be approximated as constant. We can take this approximation for granted, as current ordinary digital cameras automatically do it for us as a part of their accuracy limitation.

The second problem can be resolved by considering Eq. (6). In this equation, two specular pixels will have the same m_d but different \mathbf{I} , if their values of m_s are different. Equation (7) states that $m_s = \tilde{w}_s(G_r + G_g + G_b)$. In Torrance and Sparrow reflection model⁴⁴⁾, which is reasonably accurate to model specularly, \tilde{w}_s is expressed as:

$$\tilde{w}_s = FG \frac{1}{\cos \theta_r} \exp\left(-\frac{\alpha^2}{2\phi^2}\right) \quad (14)$$

where F is the Fresnel reflection, G is the geometrical attenuation factor, θ_r is the angle of surface normal and viewing direction, α is the angle between the surface normal and the bi-

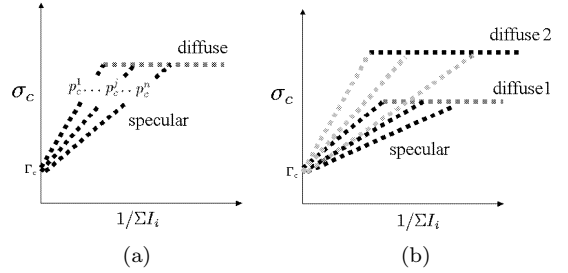


Fig. 4 (a) Sketch of specular points of uniformly colored surface in inverse-intensity chromaticity space. (b) Sketch of specular points of two surface different colors.

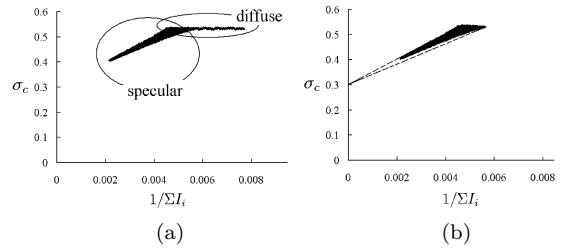


Fig. 5 (a) Diffuse and specular points of a synthetic image (Fig.3(a)) in inverse-intensity chromaticity space, with c representing the green channel. (b) The cluster of specular points which head for illumination chromaticity value in y -axis.

sector of viewing direction and illumination direction, and ϕ is the surface roughness. Thus, if the two specular pixels have the same surface color lit by distant light source and have the same m_d which implies the same \mathbf{p} , then m_s of both pixels will be different if their values of θ_r and α are different.

Hence, in general conditions, specular pixels can be grouped into a number of clusters that have the same values of p_c and different ΣI_i . For every group of pixels that share the same or approximately the same value of m_d , we can consider p_c as a constant, which makes Eq. (13) become a linear equation, with p_c as its constant gradient. These groups of pixels can be clearly observed in inverse-intensity chromaticity space, with x -axis representing $1/\Sigma I_i$ and y -axis representing σ_c , as illustrated in **Fig. 4** (a). Several straight lines in the figure correspond to several groups of different m_d values (several number of different p_c : $p_c^1, \dots, p_c^j, \dots, p_c^n$, where c is identical to the c of σ_c). These lines intersect at a single point on the y -axis, which is identical to the illumination chromaticity (Γ_c). **Figure 5** (a) shows the projection of all pixels

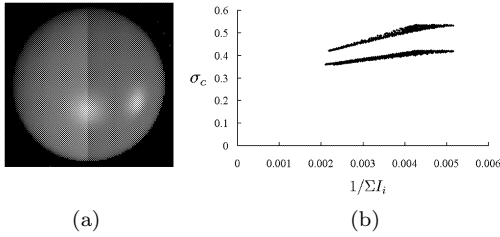


Fig. 6 (a) Synthetic image with multiple surface colors. (b) Specular points in inverse-intensity chromaticity space, with c representing the green channel.

of a synthetic image in Fig. 3 (a) into inverse-intensity chromaticity space. The horizontal line in the figure represents the diffuse points, since the image chromaticity of diffuse pixels will be constant regardless the change of Σ_i . While, the slant cluster represents the specular points. If we focus on this cluster by removing the diffuse points, according to Eq. (13) we will find that a number of straight lines, which compose the cluster, head for the value of illumination chromaticity at y -axis, as shown in Fig. 5 (b).

Now we relax the assumption of uniformly colored surface to handle multicolored surfaces. Figure 4 (b) illustrates the projection of two different surface colors into inverse-intensity chromaticity space. We can observe two specular clusters with different values of diffuse chromaticity head for the same value on the chromaticity axis (Γ_c). Since we only consider points that have the same values of p_c and Γ_c , then even if there are many different clusters with different values of Λ_c , as is the case for multicolored surfaces, we can still safely estimate the illumination chromaticity (Γ_c). This means that, for multicolored surfaces, the estimation process is exactly the same to the case of a uniformly colored surface. **Figure 6** (b) shows the projection of highlighted regions of a synthetic image with two surface colors (Fig. 6 (a)) into inverse-intensity chromaticity space.

3.3 Computational Method

To estimate every value of illumination chromaticity ($\{\Gamma_r, \Gamma_g, \Gamma_b\}$) from inverse-intensity chromaticity space, we use the Hough transform for each color channel. **Figure 7** (a) shows the transformation from inverse-intensity chromaticity space into the Hough space, where its x -axis represents Γ_c with index c representing color channel we want to estimate, and its y -

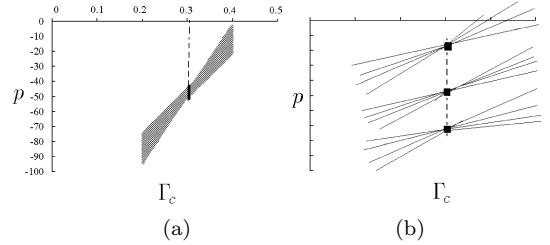


Fig. 7 (a) Projection of points in Fig. 5 (b) into Hough space. (b) Sketch of intersected lines in Hough space.

axis represents p_c , with c equals the c of Γ_c . Since Γ_c is a normalized value, the range of its value is from 0 to 1 ($0 < \Gamma_c < 1$).

Using the Hough transform alone does not yet give any solution, because the values of p_c are not constant throughout the image, which makes the intersection point of lines not located at a single location. Fortunately, even if the values of p_c vary, the values of Γ_c are constant. Thus, in principle, all intersections will be concentrated at a single value of Γ_c , with a small range of p_c 's values. These intersections are indicated by a thick solid line in Fig. 7 (a). If we focus on the intersections in the Hough space as illustrated in Fig. 7 (b), we should find a larger number of intersection at a certain value of Γ_c compared to other values of Γ_c . The reason is, in inverse-intensity chromaticity space, within the range of Γ_c ($0 < \Gamma_c < 1$), the number of groups of points that form a straight line heading for certain value of Γ_c are more dominant than the number of groups of points that form a straight line heading for other values of Γ_c .

In practice, we count the intersections in the Hough space based on the number of points that occupy the same location. The details are as follows. A line in the Hough space is formed by a number of points. If this line is not intersected by other lines, then each point will occupy a certain location uniquely (one point for each location). However, if two lines intersect, a location where the intersection takes place will be shared by two points. The number of points will increase if other lines also intersect with those two lines at the same location. Thus, to count the intersections, we first discard all points that occupy a location uniquely, as it means there are no intersections, and then count the number of points for each value of Γ_c .

As a consequence, by projecting the total number of intersections of each Γ_c into a two-dimensional space, illumination-chromaticity

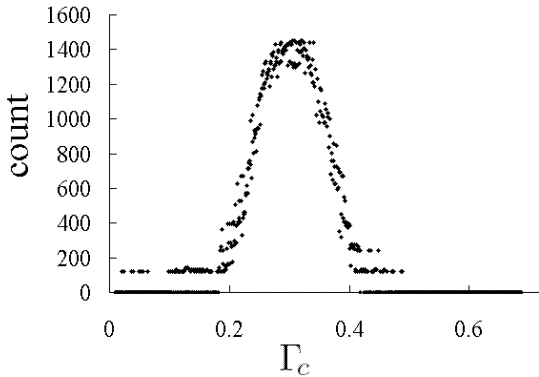


Fig. 8 Intersection-counting distribution of the green channel. The estimated illumination chromaticity is as follows: $\Gamma_r = 0.535$, $\Gamma_b = 0.303$, $\Gamma_g = 0.162$, the ground-truth values are: $\Gamma_r = 0.536$, $\Gamma_b = 0.304$, $\Gamma_g = 0.160$.

count space, with y -axis representing the count of intersections and x -axis representing Γ_c , we can robustly estimate the actual value of Γ_c . **Figure 8(a)** shows the distribution of the count numbers of intersections in the space, where the distribution forms a Gaussian-like distribution. The peak of the distribution lies at the actual value of Γ_c .

3.4 Implementation

Implementation of the proposed method is quite simple. Given an image that has highlights, we first find the highlight regions by using thresholding on image intensity and saturation values. Following the method of Lehmann, et al. ²⁸⁾, we define the thresholding as follows:

$$\begin{aligned} \tilde{I} &= \frac{I_r + I_g + I_b}{3} > T_a \tilde{I}^{\max} \\ \tilde{S} &= 1 - \frac{\min(I_r, I_g, I_b)}{\tilde{I}} < T_b \tilde{S}^{\max} \end{aligned} \quad (15)$$

where \tilde{I}^{\max} and \tilde{S}^{\max} are the largest \tilde{I} and \tilde{S} in the whole input image, respectively. T_a and T_b are the thresholds of image intensity and saturation, respectively. In our implementation, we set T_a and T_b from 0.4 – 0.6.

This thresholding technique cannot always produce precise highlight regions. Fortunately, in practice our estimation method does not need precise highlight region, even if relatively small regions of diffuse pixels are included, the algorithm could work robustly. Of course, more preciseness is better. Then, for each color channel, we project the highlight pixels into inverse-intensity chromaticity space. From this space, we use the conventional Hough transform to project the clusters into Hough space. Dur-

ing the projection, we count all possible intersections at each value of chromaticity. We plot these intersection-counting numbers into the illumination-chromaticity count space. Ideally, from this space, we can choose the tip as the estimated illumination chromaticity. However, as noise always exists in real images, the result can be improved by computing the median of a certain percentage from the highest counts. In our implementation, we use 30% from the highest counted number.

Note that, first, in our current implementation we estimate three color channels of illumination chromaticity independently. In fact, since $(\Gamma_r + \Gamma_g + \Gamma_b) = 1$, we can solely estimate two color channels instead of three color-channels. Second, the problem of determining highlight regions is still an open challenging problem, and our method could fail for specific domains that do not follow our thresholding described in Eq. (15).

4. Reflection Components Separation

4.1 Normalization

In our method, to separate reflection components correctly, the color of the specular component must be pure white ($\Gamma_r = \Gamma_g = \Gamma_b$). Therefore, we have to normalize the input image, since real images rarely have pure white illumination chromaticity. The normalization requires the value of Γ (illumination chromaticity), which can be estimated using color constancy algorithms explained in Section 3, or other methods ^{11),38)}. We express the estimated illumination chromaticity as Γ^{est} , with $\Gamma^{est} = \{\Gamma_r^{est}, \Gamma_g^{est}, \Gamma_b^{est}\}$, and the normalized image as:

$$\mathbf{I}'(\mathbf{x}) = m'_d(\mathbf{x})\mathbf{\Lambda}'(\mathbf{x}) + m'_s(\mathbf{x})\frac{1}{3} \quad (16)$$

where $\mathbf{I}'(\mathbf{x}) = \frac{\mathbf{I}(\mathbf{x})}{\Gamma^{est}} = \{\}$, the normalized image intensity. $m'_d = m_d \left[\frac{\Lambda_r(\mathbf{x})}{\Gamma_r^{est}} + \frac{\Lambda_g(\mathbf{x})}{\Gamma_g^{est}} + \frac{\Lambda_b(\mathbf{x})}{\Gamma_b^{est}} \right]$, $\mathbf{\Lambda}'$

is the chromaticity of $\left(m_d \frac{\mathbf{\Lambda}(\mathbf{x})}{\Gamma^{est}} \right)$, which we call the normalized diffuse chromaticity. We assume $\frac{\Gamma}{\Gamma^{est}} = \{1, 1, 1\}$, as a result the normalized specular chromaticity ($\mathbf{\Gamma}'$) equals $\{1/3, 1/3, 1/3\}$, and $m'_s = 3m_s$. The above normalization makes the specular reflection component become a scalar value.

Later, when the separation is done, to obtain the actual reflection components, we need to renormalize the separated components, sim-

ply by multiplying them ($m'_d(\mathbf{x})\mathbf{\Lambda}'(\mathbf{x})$ and $m'_s(\mathbf{x})\frac{1}{3}$) with $\mathbf{\Gamma}^{est}$:

$$m_d(\mathbf{x})\mathbf{\Lambda}(\mathbf{x}) = \left[m'_d(\mathbf{x})\mathbf{\Lambda}'(\mathbf{x}) \right] \mathbf{\Gamma}^{est} \quad (17)$$

$$m_s(\mathbf{x})\mathbf{\Gamma} = \left[m'_s(\mathbf{x})\frac{1}{3} \right] \mathbf{\Gamma}^{est} \quad (18)$$

4.2 Specular to Diffuse mechanism

To separate the reflection components, we basically rely on the *specular-to-diffuse mechanism*. This mechanism is derived from *maximum chromaticity* and intensity values of diffuse and specular pixels. Following the chromaticity definition in Eq.(3) we define maximum chromaticity as:

$$\tilde{\sigma}'(\mathbf{x}) = \frac{\max(I'_r(\mathbf{x}), I'_g(\mathbf{x}), I'_b(\mathbf{x}))}{I'_r(\mathbf{x}) + I'_g(\mathbf{x}) + I'_b(\mathbf{x})} \quad (19)$$

where $\{I'_r(\mathbf{x}), I'_g(\mathbf{x}), I'_b(\mathbf{x})\}$ are obtained from a normalized image (\mathbf{I}' in Eq.(16)). Identically, we can express:

$$\tilde{\sigma}'(\mathbf{x}) = \max(\sigma'_r(\mathbf{x}), \sigma'_g(\mathbf{x}), \sigma'_b(\mathbf{x})) \quad (20)$$

where σ' is the image chromaticity of the normalized image. Unlike chromaticity (σ'), maximum chromaticity ($\tilde{\sigma}'$) is a scalar value.

For a uniformly colored surface that has been normalized, in a two-dimensional space: maximum chromaticity intensity space, where its x -axes representing $\tilde{\sigma}'$ and its y -axes representing \tilde{I}' , with $\tilde{I}' = \max(I'_r, I'_g, I'_b)$, the diffuse points' maximum chromaticities of the image are always larger than the specular points' maximum chromaticity, due to the maximum chromaticity definition (19). Mathematically, it can be proved by comparing the values of maximum chromaticity ($\tilde{\sigma}'$) of diffuse and specular pixels defined in Eq.(16):

$$\tilde{\sigma}'_{diff} > \tilde{\sigma}'_{spec} \quad (21)$$

$$\frac{\tilde{\Lambda}'}{\Lambda'_r + \Lambda'_g + \Lambda'_b} > \frac{m'_d \tilde{\Lambda}' + \frac{1}{3} m'_s}{m'_d(\Lambda'_r + \Lambda'_g + \Lambda'_b) + m'_s} \quad (22)$$

$$\tilde{\Lambda}' > \frac{1}{3} \quad (23)$$

where $\tilde{\Lambda}' = \max(\Lambda'_r, \Lambda'_g, \Lambda'_b)$, the Λ'_c of \tilde{I}' (with index c is identical to the color channel of \tilde{I}'), and $(\Lambda'_r + \Lambda'_g + \Lambda'_b) = 1$. Thus, since the values of $\tilde{\Lambda}'$ for chromatic pixels are always larger than $1/3$, the last equation holds true.

In addition, using either the chromaticity or

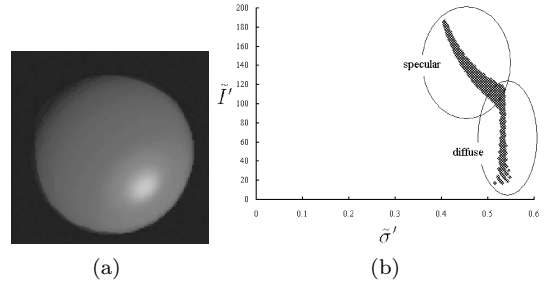


Fig. 9 (a) Synthetic image. (b) Projection of the synthetic image pixels into the maximum chromaticity intensity space.

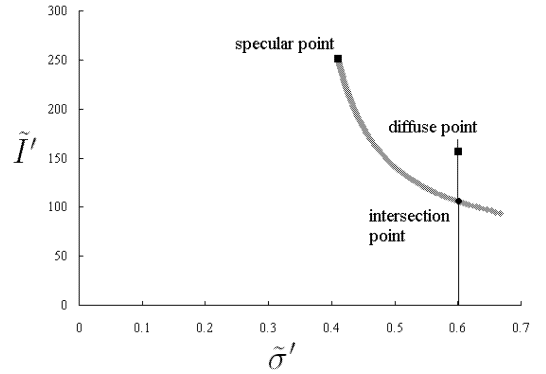


Fig. 10 Specular-to-diffuse mechanism. The intersection point is equal to the diffuse component of the specular pixel. By knowing diffuse chromaticity from the diffuse pixel, the intersection point can be obtained.

the maximum chromaticity definition, the chromaticity values of the diffuse points will be constant, regardless of the variance of $m'_d(\mathbf{x})$. In contrast, the chromaticity values of specular points will vary with regard to the variance of $m'_s(\mathbf{x})$, as shown in **Fig. 9** (b). From these different characteristics of specular and diffuse points in the maximum chromaticity intensity space, we devised specular-to-diffuse mechanism. The details are as follows.

When two normalized pixels, a specular pixel $\mathbf{I}'(\mathbf{x}_1)$ and a diffuse pixel $\mathbf{I}'(\mathbf{x}_2)$, with the same $\mathbf{\Lambda}'$ are projected into the maximum chromaticity intensity space, the location of the diffuse point will be at the right side of the specular point since, diffuse's maximum chromaticity is larger than specular's maximum chromaticity. Then, by subtracting every color channel of the specular pixel's intensity using a small scalar number iteratively, and projecting the subtracted values into the maximum chromaticity intensity space, we will find that the projected points form a curved line in the space, as

shown in **Fig. 10**. This curved line follows the following equation (see Appendix A for complete derivation):

$$\tilde{I}'(\mathbf{x}) = m'_d(\mathbf{x})(\tilde{\Lambda}'(\mathbf{x}) - 1/3) \left(\frac{\tilde{\sigma}'(\mathbf{x})}{\tilde{\sigma}'(\mathbf{x}) - 1/3} \right) \quad (24)$$

The last equation proves that the distribution of specular points in maximum chromaticity intensity space forms a curved cluster if the values of m'_d vary (Fig. 9 (b)).

In Fig. 10, we can observe that a certain point in the curved line intersects with a vertical line representing the maximum chromaticity of the diffuse point. At this intersection, m'_s of the specular pixel equals zero, since the maximum chromaticity of the subtracted specular pixel becomes identical to that of the diffuse pixel. As a consequence, the intersection point becomes crucial, since it indicates the diffuse component of the specular pixel ($m'_d(\mathbf{x}_1)\tilde{\Lambda}'$). To obtain this value, we first compute $m'_d(\mathbf{x}_1)$, which can be derived from Eq. (24):

$$m'_d(\mathbf{x}_1) = \frac{\tilde{I}'(\mathbf{x}_1)[3\tilde{\sigma}'(\mathbf{x}_1) - 1]}{\tilde{\sigma}'(\mathbf{x}_1)[3\tilde{\Lambda}'(\mathbf{x}_1) - 1]} \quad (25)$$

To compute m'_d we need to know the value of $\tilde{\Lambda}'(\mathbf{x}_1)$. This value can be obtained from the diffuse pixel since, if the two pixels have the same diffuse chromaticity, then $\tilde{\Lambda}'(\mathbf{x}_1) = \tilde{\Lambda}'(\mathbf{x}_2) = \tilde{\sigma}'(\mathbf{x}_2)$. Upon knowing the value of $m'_d(\mathbf{x}_1)$, we can directly obtain the value of $m'_s(\mathbf{x}_1)$, since $m'_s(\mathbf{x}_1) = (I'_r(\mathbf{x}_1) + I'_b(\mathbf{x}_1) + I'_g(\mathbf{x}_1)) - m'_d(\mathbf{x}_1)$. As a result, the normalized diffuse reflection component of the specular pixel becomes able to obtain: $m'_d(\mathbf{x}_1)\tilde{\Lambda}'(\mathbf{x}_1) = \mathbf{I}'(\mathbf{x}_1) - \frac{m'_s(\mathbf{x}_1)}{3}$.

To correctly compute the diffuse component ($m'_d(\mathbf{x}_1)\tilde{\Lambda}'$), the mechanism needs a linearity between the camera output and the flux of incoming light intensity. Moreover, in the case of the above two pixels, the mechanism can successfully obtain the reflection components because the diffuse chromaticity is known. Unfortunately, given a multicolored image as shown in **Fig. 11**, the diffuse chromaticity for each color is unknown; this, in fact, is the main problem of separating reflection components by using a single multicolored image.

Although we cannot directly use specular-to-diffuse mechanism to separate the reflection components, the mechanism is still useful, since it tells us that the diffuse component of a

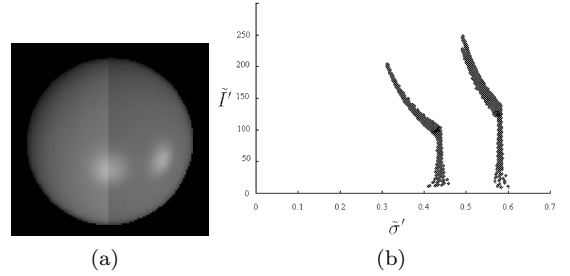


Fig. 11 (a) Synthetic image with multicolored surface. (b) Projection of the synthetic image pixels into the maximum chromaticity intensity space.

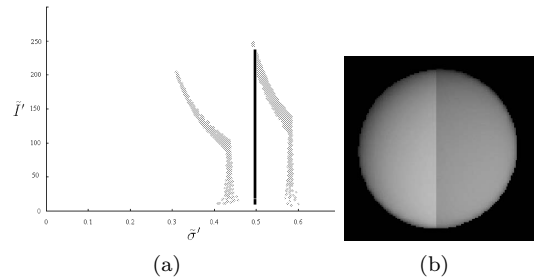


Fig. 12 (a) Shifting all pixels into arbitrary $\tilde{\Lambda}'$. (b) Specular-free image.

specular pixel lies somewhere in the curved line (Eq. (24)). Furthermore, by using the mechanism, we are also able to generate a specular-free image, which is one of the crucial components in our proposed method.

4.3 Specular-Free Image

To generate a *specular-free image*, we simply set the diffuse maximum chromaticity ($\tilde{\Lambda}'$ in Eq. (25)) equal to an arbitrary scalar value ($1/3 < \tilde{\Lambda}' \leq 1$), for all pixels regardless of their color. For instance, we set $\tilde{\Lambda}'$ equal to 0.5 for image in Fig. 9 (a), which implies that the distribution of the points in maximum chromaticity-intensity space becomes a vertical line as shown in **Fig. 12** (a). As a result, we can obtain an image that does not have specular reflections (Fig. 12 (b)). **Figure 13** (a) shows a real image of a multicolored scene. By setting $\tilde{\Lambda}' = 0.5$ for all pixels, we can obtain an image that is geometrically identical to the diffuse component of the input image (Fig. 13 (b)). The difference of both is solely in their surface colors.

This technique can successfully remove highlights mainly because the saturation values of all pixels are made constant regarding to the

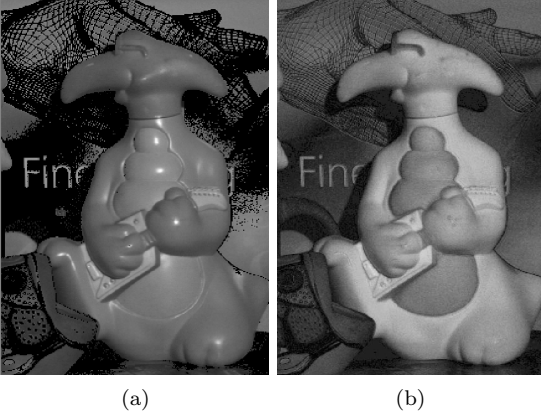


Fig. 13 (a) Normalized input image. (b) Specular-free image by setting $\tilde{\Lambda}' = 0.5$. The specular components are perfectly removed, but the surface color is different.

maximum chromaticity, while retaining their hue^{2),6)}. It is well known that, if the specular component's color is pure white, then diffuse and specular pixels that have the same surface color will have identical values of hue, with the hue defined as¹⁸⁾:

$$H = \cos^{-1} \left[\frac{\frac{1}{2} [(I'_r - I'_g) + (I'_r - I'_b)]}{[(I'_r - I'_g)^2 + (I'_r - I'_b)(I'_g - I'_b)]^{\frac{1}{2}}} \right] \quad (26)$$

and difference saturation values, with saturation is defined as¹⁸⁾:

$$S = 1 - \left[\frac{3}{I'_r + I'_g + I'_b} \min(I'_r, I'_g, I'_b) \right] \quad (27)$$

In our dichromatic reflection model (Eq. (16)), different saturation means different value of m'_s (the weighting factor of specular component), and the same hue means the same value of Λ' (the normalized diffuse chromaticity). As consequences, in maximum chromaticity intensity space, for diffuse points with the same Λ' , both saturation and hue values will be constant (since their m'_s values equal zero) while, for specular points with the same Λ' , their saturation values will vary (since their m'_s values vary), and the hue values will be constant. Thus, shifting all points in maximum chromaticity intensity space into a certain arbitrary value using a specular-to-diffuse mechanism is identical to making all points' saturation values constant, but retaining their hue values intact. These constant-saturation values can make the

highlights disappear from the image.

Formally, we describe the specular-free image as:

$$\hat{\mathbf{I}}(\mathbf{x}) = \hat{m}_d(\mathbf{x})\hat{\Lambda}(\mathbf{x}) \quad (28)$$

where $\hat{\mathbf{I}} = \{\hat{I}_r, \hat{I}_g, \hat{I}_b\}$ is the image intensity of the specular-free image, $\hat{\Lambda} = \{\hat{\Lambda}_r, \hat{\Lambda}_g, \hat{\Lambda}_b\}$ is the diffuse chromaticity, and \hat{m}_d is the diffuse weighting factor. In the following, we will prove that \hat{m}_d has the same geometrical profile to m'_d (the diffuse weighting factor of normalized image).

According to Eq. (16) a normalized diffuse pixel is described as $\mathbf{I}'(\mathbf{x}) = m'_d(\mathbf{x})\Lambda'(\mathbf{x})$. If we apply the specular-to-diffuse mechanism to the pixel by substituting the value of $\tilde{\Lambda}'$ in Eq. (25) where $\tilde{\Lambda}' = \max(\Lambda'_r, \Lambda'_g, \Lambda'_b)$ with an arbitrary maximum chromaticity whose value equals $\max(\hat{\Lambda}_r, \hat{\Lambda}_g, \hat{\Lambda}_b)$, then the equation becomes:

$$\hat{m}_d(\mathbf{x}) = \frac{\tilde{I}'(\mathbf{x})[3\tilde{\sigma}'(\mathbf{x}) - 1]}{\tilde{\sigma}'(\mathbf{x})[3\max(\hat{\Lambda}_r, \hat{\Lambda}_g, \hat{\Lambda}_b) - 1]} \quad (29)$$

Since $\tilde{I}'(\mathbf{x}) = m'_d(\mathbf{x})\tilde{\Lambda}'(\mathbf{x})$, and for diffuse pixels $\tilde{\Lambda}'(\mathbf{x}) = \tilde{\sigma}'(\mathbf{x})$, by defining $\tilde{\Lambda}^{new} = \max(\hat{\Lambda}_r, \hat{\Lambda}_g, \hat{\Lambda}_b)$, we can obtain:

$$\hat{m}_d(\mathbf{x}) = m'_d(\mathbf{x}) \frac{3\tilde{\Lambda}'(\mathbf{x}) - 1}{3\tilde{\Lambda}^{new} - 1} \quad (30)$$

$\tilde{\Lambda}^{new}$ is independent of the spatial parameter (\mathbf{x}) , since we use the same value $\tilde{\Lambda}^{new}$ for all pixels regardless of their colors. Note that the same value of $\tilde{\Lambda}^{new}$ does not necessarily imply the same value $\hat{\Lambda}$. As a result, for diffuse pixels with the same diffuse chromaticity (the same surface color), $\frac{3\tilde{\Lambda}'(\mathbf{x})-1}{3\tilde{\Lambda}^{new}-1}$ will be constant, thereby enabling us to describe the image intensity of specular-free image as:

$$\hat{\mathbf{I}}(\mathbf{x}) = m'_d(\mathbf{x})k(\mathbf{x})\hat{\Lambda}(\mathbf{x}) \quad (31)$$

where $k(\mathbf{x}) = \frac{3\tilde{\Lambda}'(\mathbf{x})-1}{3\tilde{\Lambda}^{new}-1}$. For pixels with the same diffuse chromaticity (Λ'), k is a constant scalar value. For the proof for specular pixels, see Appendix B. Therefore, since $\hat{m}_d(\mathbf{x}) = m'_d(\mathbf{x})k$, the diffuse geometrical profile of the specular-free image is identical to the geometrical profile of both the normalized image (16) and the input image (6).

Generating a specular-free image using specular-to-diffuse mechanism is a one-pixel-based operation that requires only a single col-

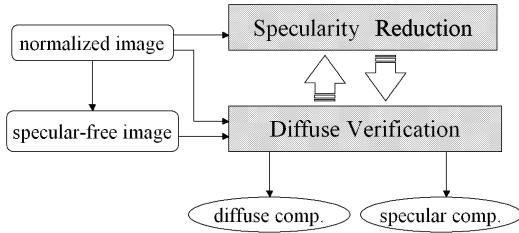


Fig. 14 Basic Flow of the proposed method.

ored image without any segmentation process. As a result, it is simple and could be useful for many applications in computer vision that do not need actual surface color but suffer from highlights. Note that caution should be taken in using a specular-free image, particularly for applications that require evaluating color discontinuities since, in the case of two adjacent colors that have the same hue but different saturation, color discontinuities of the two colors will disappear.

5. Separation Method

Flowchart in **Fig. 14** illustrates the basic idea of our proposed method. First, given a normalized image, a specular-free image is generated. Based on these two images (the normalized image and the specular free image), the ‘diffuse verification’ verifies whether the normalized image has diffuse only pixels. If it has diffuse only, then the processes terminate. Otherwise, the ‘specularity reduction’ will decrease the intensity of the specular pixels of the normalized image. After that, the diffuse verification verifies once again whether the normalized image has diffuse-only pixels. These two processes are done iteratively until there is no specularity in the normalized image. All processes require only two adjacent pixels to accomplish their task; and, this local operation is indispensable in dealing with highly textured surfaces. The following subsections will show the detail of the two processes.

5.1 Diffuse Pixels Verification

5.1.1 Intensity Logarithmic Differentiation

Given one colored pixel, to determine whether it is diffuse or specular pixel is completely an ill posed problem. Since in a linear equation such as Eq. (16), only from a single \mathbf{I}' , whether m'_s is equal to zero is undeterminable. In this section, instead of a single pixel, we will show that two-neighboring pixels can be the minimum requirement to determine

whether both of them are diffuse pixels.

We base our technique on intensity logarithmic differentiation of the normalized image and the specular free image. Considering a diffuse pixel which is not located at color discontinuities in Fig.13 (a), we can describe it as: $\mathbf{I}'(\mathbf{x}_1) = m'_d(\mathbf{x}_1)\mathbf{\Lambda}'$. The spatial parameter (\mathbf{x}_1) is removed from $\mathbf{\Lambda}'$, since the pixel is not located at color discontinuities. If we apply logarithmic and then differentiation operation on this pixel, the equation becomes:

$$\log(\mathbf{I}'(\mathbf{x}_1)) = \log(m'_d(\mathbf{x}_1)) + \log(\mathbf{\Lambda}') \quad (32)$$

$$\frac{d}{d\mathbf{x}} \log(\mathbf{I}'(\mathbf{x}_1)) = \frac{d}{d\mathbf{x}} \log(m'_d(\mathbf{x}_1)) \quad (33)$$

For the same pixel’s location (\mathbf{x}_1) , we can obtain a corresponding pixel in the specular-free image. We describe it as: $\dot{\mathbf{I}}(\mathbf{x}_1) = m'_d(\mathbf{x}_1)k\dot{\mathbf{\Lambda}}$, where k and $\dot{\mathbf{\Lambda}}$ are independent from spatial parameter. Thus, using the same operations, logarithmic and differentiation, we can obtain:

$$\log(\dot{\mathbf{I}}(\mathbf{x}_1)) = \log(m'_d(\mathbf{x}_1)) + \log(k) + \log(\dot{\mathbf{\Lambda}}) \quad (34)$$

$$\frac{d}{d\mathbf{x}} \log(\dot{\mathbf{I}}(\mathbf{x}_1)) = \frac{d}{d\mathbf{x}} \log(m'_d(\mathbf{x}_1)) \quad (35)$$

The last equation has the same result to Eq.(33). It means that the differential logarithmic of the diffuse pixels of the normalized image (Eq. (33)) and the differential logarithmic of the corresponding pixels in the specular free image (Eq. (35)) are exactly identical.

As a result, based on the intensity logarithmic differentiation operation, we become able to determine whether two-neighboring pixels are diffuse pixels:

$$\Delta(\mathbf{x}) = d \log(\mathbf{I}'(\mathbf{x})) - d \log(\dot{\mathbf{I}}(\mathbf{x})) \quad (36)$$

$$\Delta(\mathbf{x}) \begin{cases} = 0 & : \text{diffuse} \\ \neq 0 & : \text{specular} \\ & : \text{or color discontinuity} \end{cases} \quad (37)$$

As shown in Eq.(37), for pixels located at color discontinuities, there is still an ambiguity between specular and color discontinuity pixels. Since using two neighboring pixels that have different surface color, the difference of the logarithmic differentiation does not equal zero, although the pixels are diffuse pixels. Theoretically, by extending the number of pixels into at least four neighboring pixels, it is possible to distinguish them. However, in real images,

camera noise and surface noise (surface variance)^{19),39)} make such identification become error-prone; consequently, to deal with the color discontinuity problem, we need another more robust analysis which will be described in the next subsection.

5.1.2 Color Discontinuity

A number of methods have been proposed to solve the color discontinuity problem, which is also known as the problem of material changes^{15),20)}. Unlike those methods, we use a simple chromaticity-based method to handle the problem. We use the below decision rule:

$$\left[\begin{array}{l} \Delta r > thR \text{ and} \\ \Delta g > thG \end{array} \right] \begin{cases} 1 & : \text{color discontinuity} \\ 0 & : \text{otherwise} \end{cases} \quad (38)$$

where thR and thG are the small scalar numbers. $\Delta r(\mathbf{x}) = \sigma'_r(\mathbf{x}) - \sigma'_r(\mathbf{x} - \mathbf{1})$ and $\Delta g(\mathbf{x}) = \sigma'_g(\mathbf{x}) - \sigma'_g(\mathbf{x} - \mathbf{1})$, with $\sigma'_r = \frac{I'_r}{I'_r + I'_g + I'_b}$ and $\sigma'_g = \frac{I'_g}{I'_r + I'_g + I'_b}$. This simple technique is similar to the method proposed by Funt, et al.¹³⁾.

For two neighboring pixels, this simple chromaticity thresholding is sufficient since when two neighboring pixels have the same surface color, their chromaticity difference is small, even for specular pixels. This is one of the advantages of our local, two-neighboring-pixels operation. Moreover, the above thresholding can also solve the problem of two adjacent objects that have the same hue but different saturation, as long as the saturation difference is not less than that of the thresholds. Fortunately, in practice, even if the saturation difference is less than the thresholds, it does not affect the result much; since it implies that the objects have almost the same color, so that it is unnecessary to distinguish them. In addition, we have no problem when the above thresholding wrongly deems the shadow boundary to be a color discontinuity, since we have nothing to do with shadow.

5.2 Specularity Reduction

Specularity reduction is the second process of the two main processes we have proposed. The purpose of this process is to decrease the intensity of the specular pixels until we obtain diffuse only reflections. All operations in this process are still based only on two-neighboring pixels. **Figure 15** (a) shows three pixels: a ,

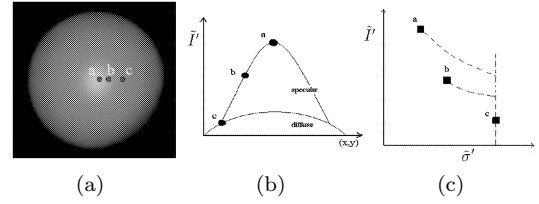


Fig. 15 (a) Three points in an image. (b) The three points in spatial-intensity space. (c) The three points in maximum chromaticity intensity space.

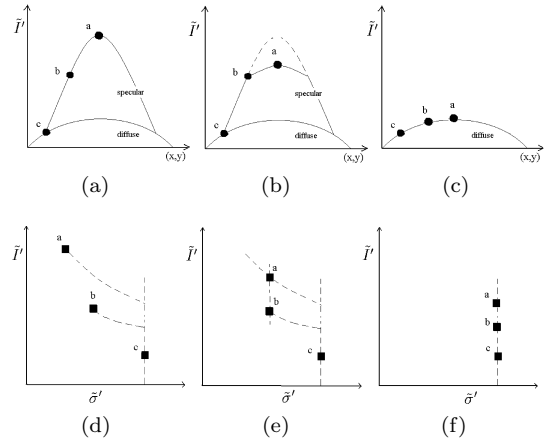


Fig. 16 Basic idea of the iterative framework using local two-pixels operation. Top row, spatial-intensity space: (a) Initial condition. (b) First looping. (c) Final condition; Bottom row, chromaticity intensity space: (d) Initial condition. (e) First looping. (f) Final condition.

b , and c . For the sake of simplicity, for the moment we assume a uniformly colored surface and those the three pixels are adjacent spatially to each other. Pixel a is the highlight's brightest pixels, and pixel c is a diffuse pixel, and pixel b is a specular pixels located between pixels a and c . In spatial-image intensity space, the image intensity of pixel a will be the largest value followed by pixels b and c , as shown in Fig. 15 (b). If we transform the pixels into maximum chromaticity-intensity space, we will obtain a point distribution illustrated in Fig. 15 (c).

Figure 16 illustrates the basic idea of our specularity reduction. In considering a two-pixel operation, the iteration begins with comparing the maximum chromaticity of point a and point b in Fig. 16 (d). From the maximum chromaticity definition in Eq. (19), we know that the smaller the m'_s is, the bigger the maximum chromaticity value. In other words, point

b is more diffuse than point a . Thus, by shifting point a using the specular-to-diffuse mechanism w.r.t the maximum chromaticity of point b , the more diffuse pixel a can be obtained, i.e., the intensity of pixel a becomes decreased and its chromaticity becomes identical to point b 's, as illustrated in Figs. 16 (b) and (e), respectively. Using the same process in the second iteration, the maximum chromaticities of point b and point c are compared and then shifted. When the maximum chromaticity of point b equals the maximum chromaticity of point c , the intensity of pixel b becomes equal to its diffuse component. The same operation is done for all pixels iteratively until their maximum chromaticity becomes the same (Fig. 16 (f)), which as a result, produces the diffuse components of the three pixels (Fig. 16 (c)).

However, the above termination condition, looping until the maximum chromaticities of all pixels are the same, is feasible only for a uniformly colored surface. In multicolored surfaces, such a termination condition will produce incorrect separation results. Therefore, to determine the termination we use the diffuse verification process explained in Subsection 5.1. We have learned that the process can identify whether an image has diffuse only pixels, even for a multicolored image. Algorithm 6.1 shows the pseudo-code of the iteration method for both uniform and multicolored surfaces.

6. Implementation

Algorithm 6.1 shows the pseudo-code of the iterative algorithm. It begins with executing function $\text{delta}(N, S, \epsilon)$, which computes the difference of the intensity logarithmic differentiation of the normalized image (N) and the specular-free image (S). In discrete operations, the logarithmic differentiation is done using: $d \log(I'_{tot}(\mathbf{x})) = \log(\Sigma I'_i(\mathbf{x} + 1)) - \log(\Sigma I'_i(\mathbf{x}))$, where $\Sigma I'_i = (I'_r + I'_g + I'_b)$. Then, the function computes $\Delta = d \log(I'_{tot}(\mathbf{x})) - d \log(\tilde{I}'_{tot}(\mathbf{x}))$, and labels the pixels of the normalized image: for pixels that have Δ more than ϵ (≈ 0), they are labeled 'specular', otherwise, they are labeled 'diffuse'.

Algorithm 6.1: ITERATION(N, S, ϵ)

comment: N = the normalized image

comment: S = the specular-free image

```

(1)  $\Delta = \text{delta}(N, S, \epsilon)$ ;
(2) while  $\text{any}(\Delta(\mathbf{x}) > \epsilon)$ 
    {
    (3) for  $\mathbf{x} \leftarrow 0$  to  $\text{sizeof}(N)-1$ 
        {
        (4) if  $\text{IsDiscontinuity}(\mathbf{x}, \mathbf{x} + 1) == \text{true}$ 
            then  $\text{next}(\mathbf{x})$ ;
        (5) if  $\text{IsDiscontinuity}(\mathbf{x}, \mathbf{x} + 1) == \text{true}$ 
            then  $\begin{cases} \mathbf{x}.flag = \text{discontinuity}; \\ (\mathbf{x} + 1).flag = \text{discontinuity}; \\ \text{next}(\mathbf{x}); \end{cases}$ 
        (6) if  $\tilde{\sigma}'(\mathbf{x}) == \tilde{\sigma}'(\mathbf{x} + 1)$ 
            then  $\begin{cases} \mathbf{x}.flag = \text{noise}; \\ (\mathbf{x} + 1).flag = \text{noise}; \\ \text{next}(\mathbf{x}); \end{cases}$ 
        (7)  $M(\mathbf{x}) = \text{Specular2Diffuse}(\mathbf{I}'(\mathbf{x}), \mathbf{I}'(\mathbf{x} + 1))$ ;
         $\text{next}(\mathbf{x})$ ;
        }
    }
     $N = M$ ;
(8)  $\Delta = \text{delta}(N, S, \epsilon)$ ;
return ( $N$ )

```

comment: N = normalized diffuse component

In Step 2 until Step 4, if there are any pixels labeled 'specular', for each of them, the algorithm examines whether the pixel and its neighbor are color discontinuity pixels. If so, then they are labeled 'discontinuity'; otherwise, then at least one of them must be a specular pixel. In Step 5, before we apply the specular-to-diffuse mechanism to both pixels, additional checking is necessary, i.e., whether both pixels' maximum chromaticity is the same. If they are the same, then the pixels are labeled 'noise'. The reason that they are noise and not specular pixels is because two-neighboring specular pixels never have the same maximum chromaticity.

In Step 6, using the specular-to-diffuse mechanism the intensity and maximum chromaticity value of the pixel that have smaller $\tilde{\sigma}'$ is shifted w.r.t. the pixel with bigger $\tilde{\sigma}'$. This is applied to all pixels, and produces a more diffuse normalized image. By setting N equal to this image (M), function $\text{delta}(N, S, \epsilon)$ is executed once again in Step 7. This time, pixels labeled 'discontinuity' and 'noise' are ignored (not included in the process). Finally, if there is still any Δ larger than ϵ , then the iteration continues; if not, the separation terminates, which consequently yields a diffuse component of the normalized image.

In our implementation, we define $\epsilon = 0$. For color discontinuity thresholds (thR and thG), we set them with the same number ranging from 0.05 to 0.1. The numbers are chosen by considering camera noise, illumination color variance, ambient light (some considerably small interreflections) and surface color variance (although human perception deems that the color surface is uniform, there is, in fact, still color variance due to dust, imperfect painting, etc.³⁹).

Algorithm 6.2: CONTROLLEDTHRESHOLD
(N, S)

comment: N = the normalized-image

comment: S = the specular-free image

RemoveAchromaticPixels(N);

stepTH = *InitialThreshold*;

while *stepTH* > ϵ

$\Delta = \text{delta}(N, S, \epsilon)$;
 if *any*($\Delta(\mathbf{x}) > \text{stepTH}$)
 then *Iteration*(N, S, stepTH);
 stepTH = *stepTH* - δ ;
 ResetAllLabels();

Renormalization(N);

return (N);

comment: N =actual diffuse component

For a more stable and robust algorithm we add an algorithm that controls the decrease of the threshold of Δ step-by-step, as described in Algorithm 6.2. In function *Iteration*(N, S, ϵ), *stepTh* will replace ϵ , which in our implementation its initial value is equal to 0.5. Ideally, the initial value should be set as large as possible; yet, by considering the time computation the number is chosen. To obtain more accurate results, the smaller subtracting number (δ) is preferable and, in our implementation, we set it equal to 0.01. To anticipate regions having achromatic pixels ($I'_r = I'_g = I'_b$), which are inevitable in the real images, we remove them by using simple thresholding in maximum chromaticity; achromatic pixels of normalized image have maximum chromaticity near 1/3.

7. Experimental Results

7.1 Experimental Conditions

We have conducted several experiments on real images, which were taken using a SONY DXC-9000, a progressive 3 CCD digital camera, by setting its gamma correction off. To ensure

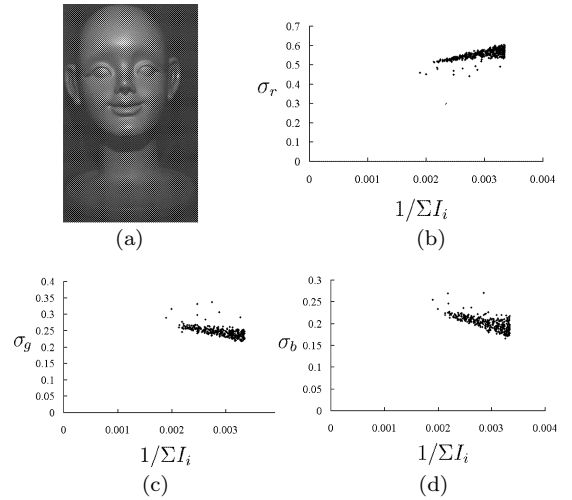


Fig. 17 (a) Real input image with a single surface color. (b) Projection of the red channel of the specular pixels into inverse-intensity chromaticity space. (c) Projection of the green channel of the specular pixels into inverse-intensity chromaticity space. (d) Projection of the blue channel of the specular pixels into inverse-intensity chromaticity space.

that the outputs of the camera are linear to the flux of incident light, we used a spectrometer: Photo Research PR-650. We examined the algorithm using four types of input, i.e., uniform colored surfaces, multicolored surfaces, highly textured surfaces, and a scene multiple objects. We used convex objects to avoid interreflection, and excluded saturated pixels from the computation. For evaluation, we compared the results with the average values of image chromaticity of a white reference image (Photo Research Reflectance Standard model SRS-3), captured by the same camera. The standard deviations of these average values under various illuminant positions and colors were approximately 0.01 ~ 0.03.

7.2 Color Constancy

7.2.1 Result on a Uniformly Colored Surface

Figure 17 (a) shows a real image of a head model that has a uniformly colored surface and relatively low specularity, illuminated by Solux Halogen with temperature 4,700 K. Under the illumination, the image chromaticity of the white reference taken by our camera has chromaticity value: $\Gamma_r = 0.371, \Gamma_g = 0.318, \Gamma_b = 0.310$.

Figure 17 (b) shows the specular points of the red channel of chromaticity in inverse-intensity

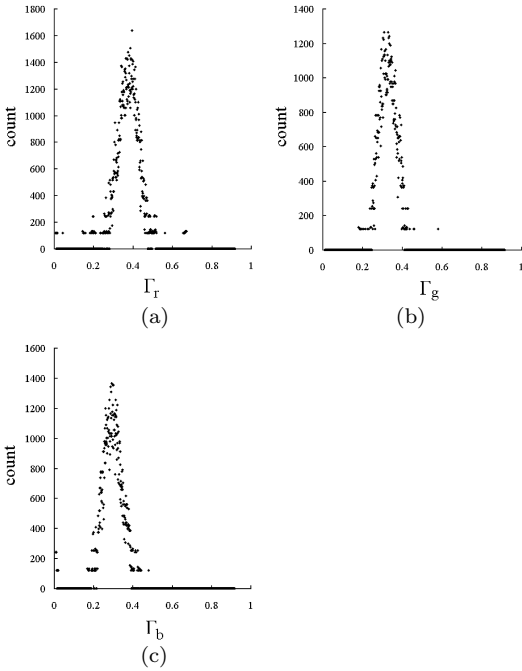


Fig. 18 (a) Intersection-counting distribution for red channel of illumination chromaticity for image in Fig. 17. (b) Intersection-counting distribution for green-channel. (c) Intersection-counting distribution for blue channel.

chromaticity space. Even though there is some noise, generally, all points form several straight lines heading for a certain point in the chromaticity axis. The same phenomenon can also be observed in Figs. 17 (c) and (d). **Figure 18** shows the intersection-counting distribution in the illumination-chromaticity count space. The peaks of the distribution denote the illumination chromaticity. The result of the estimation was: $\Gamma_r = 0.378, \Gamma_g = 0.324, \Gamma_b = 0.287$.

7.2.2 Result on a Multi-colored Surface

Figure 19 (a) shows a plastic toy with a multi-colored surface. The illumination is Solux Halogen covered with a green filter. The image chromaticity of the white reference under this illuminant taken by our camera was $\Gamma_r = 0.298, \Gamma_g = 0.458, \Gamma_b = 0.244$.

Figures 19 (b), (c), (d) show the specular points of multiple surface colors in inverse-intensity chromaticity space. From **Fig. 20**, we can observe that, even for several surface colors, the peak of intersection counts was still at a single value of Γ_c . The result of the estimation was $\Gamma_r = 0.319, \Gamma_g = 0.439, \Gamma_b = 0.212$.

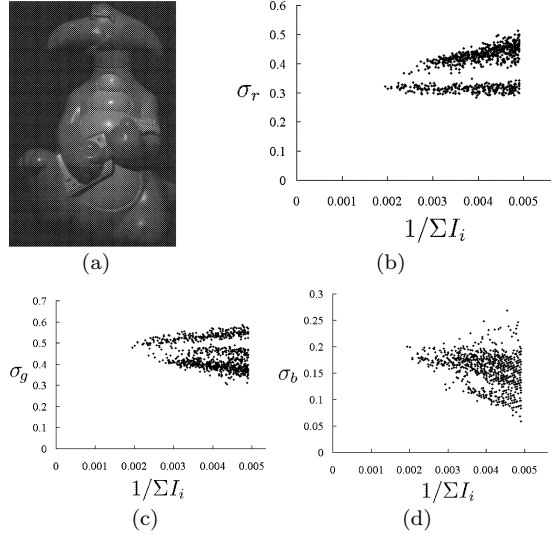


Fig. 19 (a) Real input image with multiple surface colors. (b) Projection of the red channel of the specular pixels into inverse-intensity chromaticity space. (c) Projection of the green channel of the specular pixels into inverse-intensity chromaticity space. (d) Projection of the blue channel of the specular pixels into inverse-intensity chromaticity space.

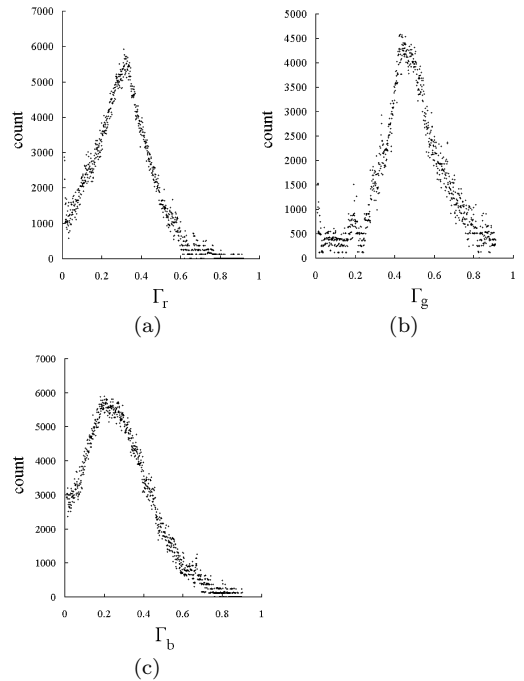


Fig. 20 (a) Intersection-counting distribution for the red channel of illumination chromaticity for image in Fig. 19. (b) Intersection-counting distribution for the green channel. (c) Intersection-counting distribution for the blue channel.

7.2.3 Result on Highly Textured Surface

Figure 21 (a) shows a magazine cover with a complex multicolored surface, which was lit by a fluorescent light covered with a green filter. The image chromaticity of the white reference under this illuminant taken by our camera has a chromaticity value of $\Gamma_r = 0.283, \Gamma_g = 0.481, \Gamma_b = 0.236$. The result of the estimation was $\Gamma_r = 0.315, \Gamma_g = 0.515, \Gamma_b = 0.207$, as shown in **Fig. 22**.

7.2.4 Result on Multiple Objects

Figure 23 (a) shows a scene with multiple objects, which was lit by a fluorescent light taken in uncontrolled environment. The image chromaticity of the white reference under this illuminant taken by our camera has a chromaticity value of $\Gamma_r = 0.337, \Gamma_g = 0.341, \Gamma_b = 0.312$. The result of the estimation was $\Gamma_r = 0.321, \Gamma_g = 0.346, \Gamma_b = 0.309$, as shown in **Fig. 24**.

7.2.5 Evaluation

To evaluate the robustness of our method, we have also conducted experiments on 6 different objects: 2 objects with a single surface color, 1 object with multiple surface colors, and 3 objects with highly textured surfaces. The colors of illuminants were grouped into 5 different colors: Solux Halogen lamp with temperature 4,700 K, incandescent lamp with temperature around 2,800 K, Solux Halogen lamp covered with green, blue and purple filters. The illuminants were arranged at various positions. The total of images in our experiment was 43 images. From these images, we calculated the errors of the estimation by comparing them with the image chromaticity of the white reference, which are shown in **Table 1**. The errors are considerably small, as the standard deviations of the reference image chromaticity are around $0.01 \sim 0.03$.

7.3 Reflection Components Separation

We evaluate the separation results by comparing the results of two polarizing filters. We place one of the two filters in front of camera and the other in front of the light source. Theoretically, if we change the polarization angle of one of the two filters into a certain angle, we can obtain diffuse only reflection. In our experiment, we changed the polarization angle of the filter placed in front of the camera. **Figures 25** (a), (b) and (c) show, respectively, the input image, the diffuse reflection compo-

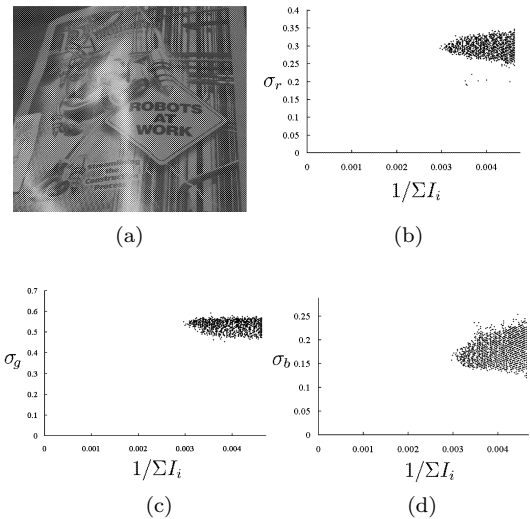


Fig. 21 (a) Real input image of complex textured surface. (b) Projection of the red channel of the specular pixels into inverse-intensity chromaticity space. (c) Projection of the green channel of the specular pixels into inverse-intensity chromaticity space. (d) Projection of the blue channel of the specular pixels into inverse-intensity chromaticity space.

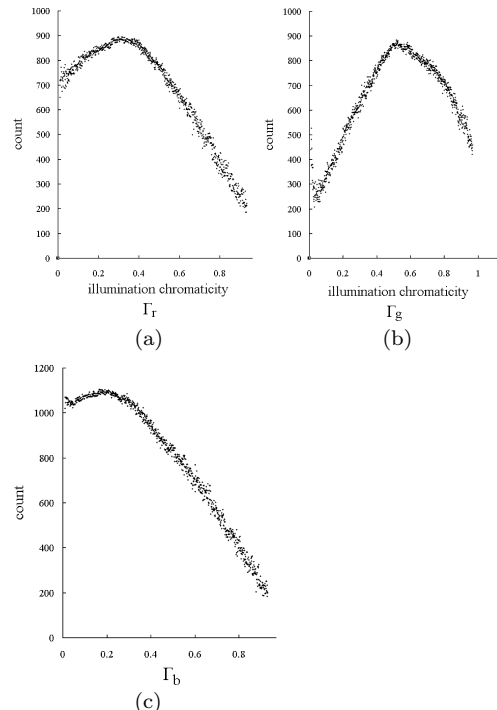


Fig. 22 (a) Intersection-counting distribution for the red channel of illumination chromaticity for image in Fig. 21. (b) Intersection-counting distribution for the green channel. (c) Intersection-counting distribution for the blue channel.

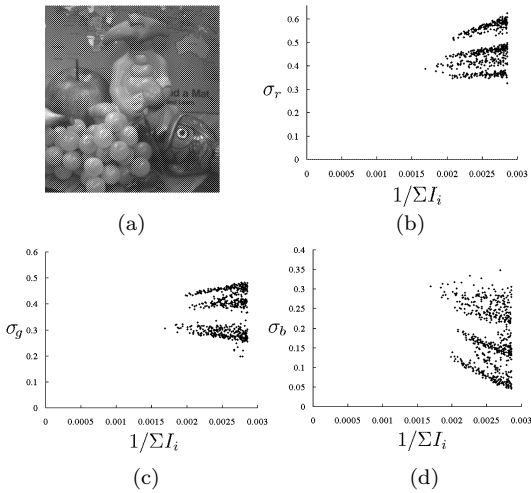


Fig. 23 (a) Real input image of a scene with multiple objects. (b) Result of projecting the specular pixels into inverse-intensity chromaticity space, with c representing the red channel. (c) Result of projecting the specular pixels, with c representing the green channel. (d) Result of projecting the specular pixels, with c representing the blue channel.

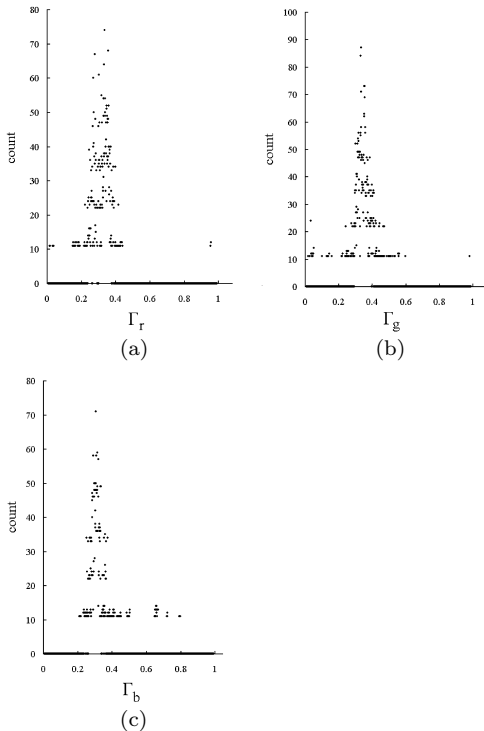


Fig. 24 (a) Intersection-counting distribution for the red channel of illumination chromaticity for image in Fig. 21. (b) Intersection-counting distribution for the green channel. (c) Intersection-counting distribution for the blue channel.

Table 1 The performance of the estimation method with regard to the image chromaticity of the white reference.

	red	green	blue
average of error	0.0172	0.0141	0.0201
std. dev. of error	0.01	0.01	0.01

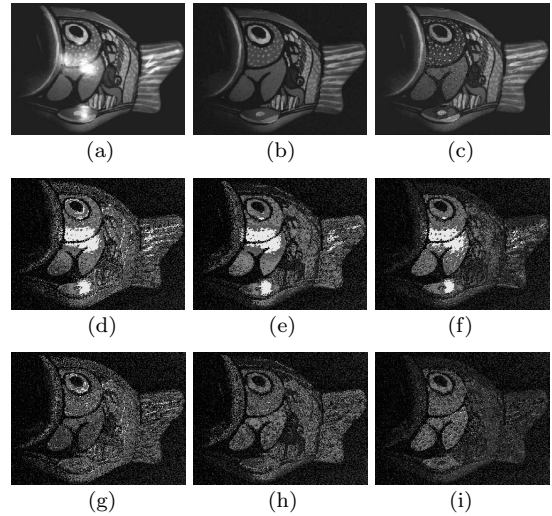


Fig. 25 Top: (a) Textured input image (b) ground truth (c) estimation. Middle: specular-ground truth comparison: (d) R-channel (e) G-channel (f) B-channel. Bottom: error: (d) R-channel (e) G-channel (f) B-channel.

ment obtained using the two polarizing filters (ground truth) and reflection components estimated using our method. Figures 25 (d), (e) and (f) show the difference of image intensity values of the input image (Fig. 25 (a)) and the ground truth (Fig. 25 (b)), in red, green and blue channels, respectively. The ranges of blue pixels in the figures are 0 ~ 5. Green pixels are 6 ~ 15, red pixels are 16 ~ 35, while yellow pixels represent larger than 35. In highlighted regions, we can observe a large difference of the intensity values in all color channels. Also, in certain places near occluding boundaries, yellow and red pixels also appear; this is caused by the difference of intensity distribution when the polarization angle is changed. Figures 25 (g), (h) and (i) show the difference of image intensity values of the estimated reflection component (Fig. 25 (c)) and the ground truth (Fig. 25 (b)) in red, green and blue, respectively. In former highlighted regions, the colors became blue, indicating that the estimation result was considerably accurate. Red and green pixels occurring in many places in the

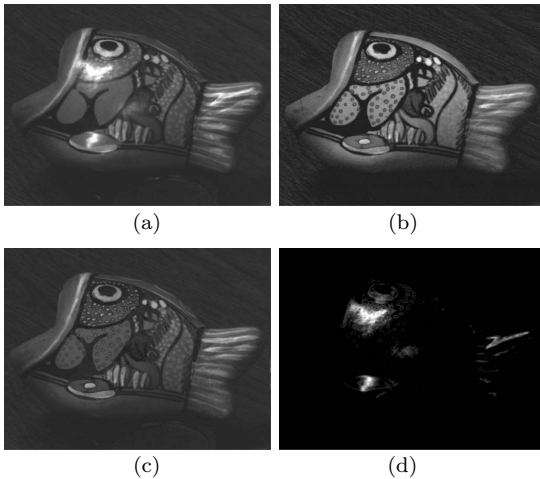


Fig. 26 (a) A complex textured surface lit with fluorescent lights. (b) The specular-free image was created by setting $\hat{\Lambda}' = 0.5$. (c) Diffuse reflection component. (d) Specular reflection component.

comparison are due to two main factors: inaccurate illumination chromaticity estimation, and the second type of noise (dark noise) that occurs as the result of using polarizing filters. Despite these factors, the estimation results are considerably accurate, since the maximum value of second type of noise of the camera (Sony DXC-9000) is around 10. Note that, in this evaluation, we do not evaluate pixels whose image intensity below camera is dark (black pixels in the evaluation represent unevaluated parts).

For a complex textured surface, **Fig. 26** (a) shows an image of a textured surface under fluorescent lights in uncontrolled environment. The specular-free image, which was generated by setting $\hat{\Lambda}^{new}$ equal to 0.5 is shown in Fig. 26 (b). Figures 26 (c) and (d) show the separated components of the object. The top part of **Fig. 27** shows a complex scene lit with fluorescent lights in an uncontrolled environment. The specular-free image result is shown in the bottom part of Fig. 27. **Figure 28** show the diffuse and specular reflections. In the estimated diffuse component (the top part of Fig. 28) and the specular-free image (the bottom part of Fig. 27), regions which are originally white become dark. The reason is that the specular-to-diffuse mechanism fails to handle achromatic pixels.

8. Conclusion

We have introduced a novel method for illumination chromaticity estimation. The pro-

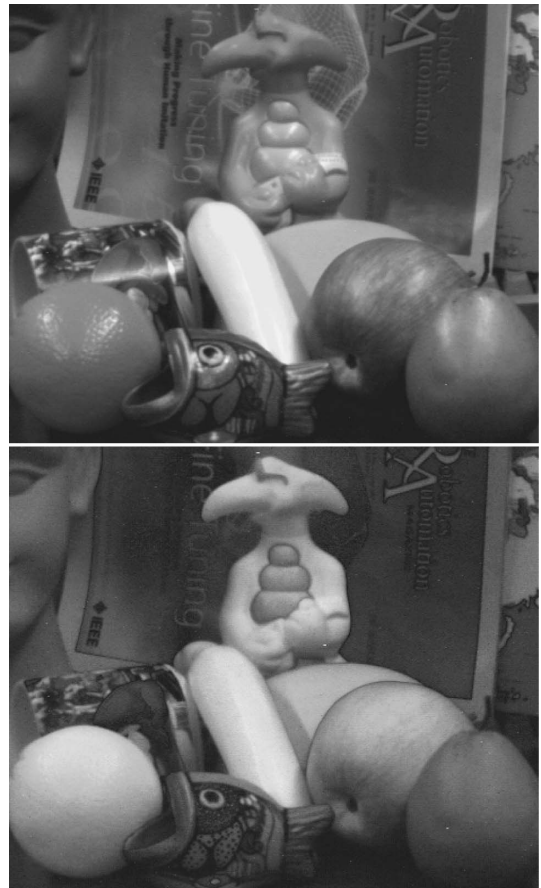


Fig. 27 Top: a complex multicolored scene lit with fluorescent lights. Bottom: the specular-free image by setting $\hat{\Lambda} = 0.5$.

posed method can handle both uniform and non-uniform surface color objects. Given crude highlight regions, the method can estimate illumination color without requiring color segmentation. It is also applicable for multiple objects with various colored surfaces, as long as there are no interreflections. In this paper, we also introduced *inverse-intensity chromaticity space* to analyze the relationship between illumination chromaticity and image chromaticity. There are a few advantages of the method. First, the capability to cope with either single surface color or multiple surface colors. Second, color segmentation inside highlight regions and intrinsic camera characteristics are not required. Third, the method does not use strong constraints on illumination, which several existing color constancy methods use, such as black-body radiator.

We also have proposed a novel method to separate diffuse and specular reflection compo-

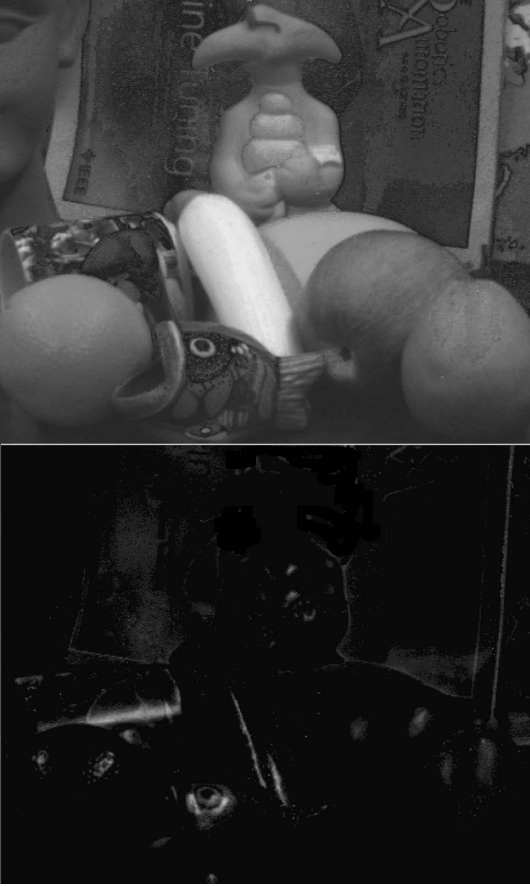


Fig. 28 Top: the diffuse reflection component.
Bottom: the specular reflection component.

nents. The main insight of the method is on the chromaticity-based iteration with regard to the logarithmic differentiation of the specular-free image. Using the method, the separation problem in textured surfaces with complex multicolored scene can be resolved without requiring explicit color segmentation. It is possible because we base our method on local operation by utilizing the specular-free image. There are three crucial factors, and thus the main contributions of our method, i.e., the specular-to-diffuse mechanism, the specular-free image, and the logarithmic differentiation-based iteration framework.

The experimental results of our color constancy and reflection components separation on complex textured images show that the proposed methods are accurate and robust.

Acknowledgments This research is supported, in part, by Ministry of Education, Culture, Sports, Science and Technology under

the Leading Project, “Development of High Fidelity Digitization Software for Large-Scale and Intangible Cultural Assets,” and, in part, by Japan Science and Technology Agency, under the CREST program, “Automatic generation of virtual models of cultural heritage.” Special thanks to Dr. Ko Nishino, who gave valuable comments and ideas to the first author during his study at the University of Tokyo, and Dr. Daisuke Miyazaki, who helps the preparation of the paper.

Appendix A

Derivation of the correlation between illumination chromaticity and image chromaticity.

$$\tilde{\sigma}'(\mathbf{x}) = \frac{m'_d(\mathbf{x})\tilde{\Lambda}'(\mathbf{x}) + \frac{1}{3}m'_s(\mathbf{x})}{m'_d(\mathbf{x})[\Lambda'_r(\mathbf{x}) + \Lambda'_g(\mathbf{x}) + \Lambda'_b(\mathbf{x})] + m'_s(\mathbf{x})} \quad (39)$$

where $[\Lambda'_r + \Lambda'_g + \Lambda'_b] = 1$. For local (pixel based) operation the location (\mathbf{x}) can be removed. Then:

$$m'_s = m'_d \frac{(\tilde{\Lambda}' - \tilde{\sigma}')}{(\tilde{\sigma}' - 1/3)} \quad (40)$$

Substituting m'_s in the definition of \tilde{I} (Eq. (16)) with m'_s in the last equation:

$$\tilde{I}' = m'_d(\tilde{\Lambda}' - 1/3) \left(\frac{\tilde{\sigma}'}{\tilde{\sigma}' - 1/3} \right) \quad (41)$$

Appendix B

A diffuse pixel from a normalized image can be described as: $\mathbf{I}'(\mathbf{x}) = m'_d(\mathbf{x})\Lambda'(\mathbf{x})$. In Section 4.3, we have shown that using specular-to-diffuse mechanism by substituting Λ' with an arbitrary value ($\tilde{\Lambda}^{new}$) whose value is between $1/3 \sim 1$, we can obtain:

$$\mathring{\mathbf{I}}(\mathbf{x}) = \mathring{m}_d(\mathbf{x})\mathring{\Lambda}(\mathbf{x}) = m'_d(\mathbf{x})k(\mathbf{x})\mathring{\Lambda}(\mathbf{x}) \quad (42)$$

where, for pixels with the same diffuse chromaticity, k is a constant scalar value. Thus, we can obtain that the geometrical profile of specular-free image is identical to that of diffuse reflection component. The proof for specular pixels is as follows:

A specular pixel with identical diffuse geometrical profile to the above diffuse pixel is described as: $\mathbf{I}'(\mathbf{x}) = m'_d(\mathbf{x})\Lambda'(\mathbf{x}) + \frac{m'_s(\mathbf{x})}{3}$. By applying specular-to-diffuse mechanism (Eq. (30)) to the specular pixel with the same value of $\tilde{\Lambda}^{new}$, we can obtain:

$$\hat{m}_d(\mathbf{x}) = \frac{\tilde{I}'(\mathbf{x})[3\tilde{\sigma}'(\mathbf{x}) - 1]}{\tilde{\sigma}'(\mathbf{x})[3\tilde{\Lambda}^{new} - 1]} \quad (43)$$

where $\tilde{I}'(\mathbf{x}) = m'_d(\mathbf{x})\tilde{\Lambda}'(\mathbf{x}) + \frac{m'_s(\mathbf{x})}{3}$, and $\tilde{\Lambda}^{new}$ is the arbitrary maximum chromaticity. Unlike diffuse pixels, for specular pixels, $\tilde{\sigma}' \neq \tilde{\Lambda}'$. Then, the last equation becomes:

$$\hat{m}_d(\mathbf{x}) = \frac{\left[m'_d(\mathbf{x})\tilde{\Lambda}'(\mathbf{x}) + \frac{m'_s(\mathbf{x})}{3} \right] [3\tilde{\sigma}'(\mathbf{x}) - 1]}{\tilde{\sigma}'(\mathbf{x})[3\tilde{\Lambda}^{new} - 1]} \quad (44)$$

Since we argued that in specular-free image specular reflection disappear ($\hat{m}_s = 0$), then \hat{m}_d of the specular pixel should equal to \hat{m}_d of the diffuse pixel:

$$\hat{m}_d^{diff} = \hat{m}_d^{spec} \quad (45)$$

$$m'_d \left[\frac{3\tilde{\Lambda}'(\mathbf{x}) - 1}{3\tilde{\Lambda}^{new} - 1} \right] = \left[m'_d(\mathbf{x})\tilde{\Lambda}'(\mathbf{x}) + \frac{m'_s(\mathbf{x})}{3} \right] \frac{[3\tilde{\sigma}'(\mathbf{x}) - 1]}{\tilde{\sigma}'(\mathbf{x})[3\tilde{\Lambda}^{new} - 1]} \quad (46)$$

$$\begin{aligned} m'_d(\mathbf{x}) \left[\frac{3\tilde{\Lambda}'(\mathbf{x}) - 1}{3\tilde{\Lambda}^{new} - 1} \right] \tilde{\sigma}'(\mathbf{x}) \\ = m'_d(\mathbf{x})\tilde{\Lambda}'(\mathbf{x}) \left[3\tilde{\sigma}'(\mathbf{x}) - 1 \right] \\ + \frac{m'_s(\mathbf{x})}{3} \left[3\tilde{\sigma}'(\mathbf{x}) - 1 \right] \end{aligned} \quad (47)$$

$$m'_d(\mathbf{x}) \left[\tilde{\Lambda}'(\mathbf{x}) - \tilde{\sigma}'(\mathbf{x}) \right] = m'_s(\mathbf{x}) \left[\tilde{\sigma}'(\mathbf{x}) - 1/3 \right] \quad (48)$$

$$m'_s(\mathbf{x}) = m'_d(\mathbf{x}) \frac{(\tilde{\Lambda}'(\mathbf{x}) - \tilde{\sigma}'(\mathbf{x}))}{(\tilde{\sigma}'(\mathbf{x}) - 1/3)} \quad (49)$$

the last equation is identical to Eq. (40) in Appendix A, which proves that $\hat{m}_d^{diff} = \hat{m}_d^{spec}$ holds true. Therefore, all pixels in a specular-free image have no specular reflection component and its geometrical profile is identical to the diffuse component of the input image.

References

- 1) Andersen, H.J. and Granum, E.: Classifying illumination conditions from two light sources by colour histogram assessment, *Journal of Optics Society of America A.*, Vol.17, No.4, pp.667–676 (2000).
- 2) Bajscy, R., Lee, S.W. and Leonardis, A.: Detection of diffuse and specular interface reflections by color image segmentation, *International Journal of Computer Vision*, Vol.17, No.3, pp.249–272 (1996).
- 3) Beckmann, P. and Spizzochino, A.: *The scattering of electromagnetic waves from rough surfaces*, Pergamon, New York (1963).
- 4) Brainard, D.H. and Freeman, W.T.: Bayesian color constancy, *Journal of Optics Society of America A.*, Vol.14, No.7, pp.1393–1411 (1997).
- 5) Criminisi, A., Kang, S.B., Swaminathan, R., Szeliski, S. and Anandan, P.: Extracting layers and analysis their specular properties using epipolar plane image analysis, *Microsoft Research Technical Report MSR-TR-2002-19* (2002).
- 6) D'Zmura, M. and Lennie, P.: Mechanism of color constancy, *Journal of Optics Society of America A.*, Vol.3, No.10, pp.1162–1672 (1986).
- 7) Finlayson, G.D.: Color in perspective, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol.18, No.10, pp.1034–1038 (1996).
- 8) Finlayson, G.D. and Funt, B.V.: Color constancy using shadows, *Perception*, Vol.23, pp.89–90 (1994).
- 9) Finlayson, G.D., Hordley, S.D. and Hubel, P.M.: Color by correlation: A simple, unifying, framework for color constancy, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol.23, No.11, pp.1209–1221 (2001).
- 10) Finlayson, G.D. and Schaefer, G.: Convex and non-convex illumination constraints for dichromatic color constancy, *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)*, Vol.I, p.598 (2001).
- 11) Finlayson, G.D. and Schaefer, G.: Solving for color constancy using a constrained dichromatic reflection model, *International Journal of Computer Vision*, Vol.42, No.3, pp.127–144 (2001).
- 12) Finlayson, G.D. and Hordley, S.D.: Color constancy at a pixel, *Journal of Optics Society of America A.*, Vol.18, No.2, pp.253–264 (2001).
- 13) Funt, B.V., Drew, M. and Brockington, M.: Recovering shading from color images, *Proc. European Conference on Computer Vision (ECCV)*, pp.124–132 (1992).
- 14) Funt, B.V., Drew, M. and Ho, J.: Color constancy from mutual reflection, *International Journal of Computer Vision*, Vol.6, No.1, pp.5–24 (1991).
- 15) Gershon, R., Jepson, A.D. and Tsotsos, J.K.: Ambient illumination and the determination of material changes, *Journal of Optics Society of America A.*, Vol.3, No.10, pp.1700–1707 (1986).
- 16) Geusebroek, J.M., Boomgaard, R., Smeulders, S. and Geert, H.: Color invariance, *IEEE Trans. on Pattern Analysis and Machine In-*

- telligence*, Vol.23, No.12, pp.1338–1350 (2001).
- 17) Geusebroek, J.M., Boomgaard, R., Smeulders, S. and Gevers, T.: A physical basis for color constancy, *The First European Conference on Colour in Graphics, Image and Vision*, pp.3–6 (2002).
 - 18) Gonzales, R.C. and Woods, R.E.: *Digital Image Processing*, Addison-Wesley (1993).
 - 19) Healey, G. and Kondepudy, R.: Radiometric ccd camera calibration and noise estimation, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol.16, No.3, pp.267–276 (1994).
 - 20) Rubin, J.M. and Richard, W.A.: Color vision: representing material changes, *AI Memo 764*, MIT Artificial Intelligence Lab. Cambridge, Mass. (1984).
 - 21) Klinker, G.J., Shafer, S.A. and Kanade, T.: The measurement of highlights in color images, *International Journal of Computer Vision*, Vol.2, pp.7–32 (1990).
 - 22) Lambert, J.H.: *Photometria sive de mensura de gratibus luminis, colorum et umbrae*, Eberhard Klett, Augsburg, Germany (1760).
 - 23) Lee, H.C.: Method for computing the scene-illuminant from specular highlights, *Journal of Optics Society of America A.*, Vol.3, No.10, pp.1694–1699 (1986).
 - 24) Lee, H.C.: Illuminant color from shading, *Perceiving, Measuring and Using Color*, p.1250 (1990).
 - 25) Lee, H.C., Breneman, E.J. and Schulte, C.P.: Modeling light reflection for computer color vision, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol.12, pp.402–409 (1990).
 - 26) Lee, S.W.: *Understanding of surface reflections in Computer vision by color and multiple views*, PhD thesis, University of Pennsylvania (1991).
 - 27) Lee, S.W. and Bajcsy, R.: Detection of specularly using color and multiple views, *Image and Vision Computing*, Vol.10, pp.643–653 (1992).
 - 28) Lehmann, T.M. and Palm, C.: Color line search for illuminant estimation in real-world scene, *Journal of Optics Society of America A.*, Vol.18, No.11, pp.2679–2691 (2001).
 - 29) Lin, S., Li, Y., Kang, S.B., Tong, X. and Shum, H.Y.: Diffuse-specular separation and depth recovery from image sequences, *Proc. European Conference on Computer Vision (ECCV)*, pp.210–224 (2002).
 - 30) Lin, S. and Shum, H.Y.: Separation of diffuse and specular reflection in color images, *Proc. IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR)* (2001).
 - 31) Nayar, S.K., Fang, X.S. and Boulton, T.: Separation of reflection components using color and polarization, *International Journal of Computer Vision*, Vol.21, No.3 (1996).
 - 32) Nayar, S.K., Ikeuchi, K. and Kanade, T.: Surface reflection: Physical and geometrical perspectives, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol.13, No.7, pp.611–634 (1991).
 - 33) Parkkinen, J.P.S., Hallikainen, J. and Jasskelainen, T.: Characteristic spectra of munsell colors, *Journal of Optics Society of America A.*, Vol.6 (1989).
 - 34) Rosenberg, C., Hebert, M. and Thrun, S.: Color constancy using kl-divergence, *Proc. IEEE International Conference on Computer Vision (ICCV)*, Vol.1, p.239 (2001).
 - 35) Sato, Y. and Ikeuchi, K.: Temporal-color space analysis of reflection, *Journal of Optics Society of America A.*, Vol.11 (1994).
 - 36) Shafer, S.: Using color to separate reflection components, *Color Research and Applications*, Vol.10, pp.210–218 (1985).
 - 37) Tan, R.T. and Ikeuchi, K.: Separating reflection components of textured surface using a single image, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, to appear (2004).
 - 38) Tan, R.T., Nishino, K. and Ikeuchi, K.: Color constancy through inverse intensity chromaticity space, *Journal of Optics Society of America A.*, Vol.21, No.3, pp.321–334 (2004).
 - 39) Tan, R.T., Nishino, K. and Ikeuchi, K.: Separating reflection components based on chromaticity and noise analysis, *IEEE Trans. on Pattern Analysis and Machine Intelligence*, Vol.26, No.10, pp.1373–1379 (2004).
 - 40) Tominaga, S.: A multi-channel vision system for estimating surface and illumination functions, *Journal of Optics Society of America A.*, Vol.13, No.11, pp.2163–2173 (1996).
 - 41) Tominaga, S., Ebisui, S. and Wandell, B.A.: Scene illuminant classification: brighter is better, *Journal of Optics Society of America A.*, Vol.18, No.1, pp.55–64 (2001).
 - 42) Tominaga, S. and Wandell, B.A.: Standard surface-reflectance model and illumination estimation, *Journal of Optics Society of America A.*, Vol.6, No.4, pp.576–584 (1989).
 - 43) Tominaga, S. and Wandell, B.A.: Natural scene-illuminant estimation using the sensor correlation, *Proc. IEEE*, Vol.90, No.1, pp.42–56 (2002).
 - 44) Torrance, K.E. and Sparrow, E.M.: Theory for off-specular reflection from roughened surfaces, *Journal of Optics Society of America*, Vol.57, pp.1105–1114 (1966).
 - 45) Wolff, L.B. and Boulton, T.: Constraining object features using polarization reflectance model,

IEEE Trans. on Pattern Analysis and Machine Intelligence, Vol.13, No.7, pp.635–657 (1991).
 (Received August 31, 2004)
 (Accepted March 4, 2005)

(Editor in Charge: *Haiyuan Wu*)



Robby T. Tan received the M.S. and Ph.D. degrees in Computer Science from The University of Tokyo, Japan, in 2001 and 2004 respectively. From May 2004 until April 2005, he was a postdoctoral fellow in Computer Vision (Ikeuchi) Laboratory, and currently he joins National ICT Australia as a research scientist and the Australian National University as an adjunct research fellow. His research interests are in color constancy, intrinsic properties of images, spectral-based analysis and the applications of optics in computer vision.



Katsushi Ikeuchi is a Professor at the Institute of Industrial Science, the University of Tokyo, Tokyo, Japan. He received the Ph.D. degree in Information Engineering from the University of Tokyo, Tokyo, Japan, in 1978. After working at the Artificial Intelligence Laboratory, Massachusetts Institute of Technology for three years, Electrotechnical Laboratory of Japanese Ministry of International Trade and Industry for five years, and the School of Computer Science, Carnegie Mellon University for ten years, he joined the university in 1996. He has received several awards, including the David Marr Prize in computational vision, and IEEE RA K-S Fu memorial best transaction paper award. In addition, in 1992, his paper, “Numerical Shape from Shading and Occluding Boundaries,” was selected as one of the most influential papers to have appeared in Artificial Intelligence Journal within the past ten years. He has served as the program/general chairman of several international conferences, including 1995 IEEE-IROS, 1996 IEEE-CVPR, 1999 IEEE-ITSC, 2003 IEEE-ICCV. He is Editor-in-Chief of the International Journal of Computer Vision. He is/was on the editorial board of IEEE Trans PAMI, IEEE Trans RA, Journal of Optical Society of America, and the Journal of Computer Vision, Graphics. He has been a fellow of IEEE since 1998. He is selected as a distinguished lecture of IEEE SP society for the period of 2000–2001, and IEEE CS society for the period of 2004–2006.