Chapter 1

Color Feature Detection

THEO GEVERS

ISLA, Informatics Institute
University of Amsterdam
Amsterdam, The Netherlands
Email: gevers@science.uva.nl

JOOST VAN DE WEIJER

LEAR, GRAVIR laboratory
INRIA
Grenoble, France
Email: Joost.van-de-weijer@inrialpes.fr

HARRO STOKMAN

ISLA, Informatics Institute
University of Amsterdam
Amsterdam, The Netherlands
Email: stokman@science.uva.nl

Invited Chapter to appear in "Color Image Processing: Emerging Applications," CRC Press, Rastislav Lukac and Konstantinos N. Plataniotis, Editors.

1.1 Introduction

The detection and classification of local structures (i.e. edges, corners, and T-Junctions) in color images is important for many applications such as image segmentation, image matching, object recognition, visual tracking in the fields of image processing and computer vision [1], [2], [3]. In general, those local image structures are detected by differential operators which are commonly restricted to luminance information. However, most of the images recorded today are in color. Therefore, in this chapter, the focus is on the use of color information to detect and classify local image features.

The basic approach to compute color image derivatives is to calculate separately the derivatives of the channels and add them to produce the final color gradient. However, the derivatives of a color edge can be in opposing directions for the separate color channels. Therefore, a summation of the derivatives per channel will discard the correlation between color channels [4]. As a solution to the opposing vector problem, DiZenzo [4] proposes the color tensor, derived from the structure tensor, for the computation of the color gradient. Adaptations of the tensor lead to a variety of local image features, such as circle detectors and curvature estimation [5], [6], [7], [8]. In this chapter, we study on methods and techniques to combine derivatives of the different color channels to compute locale image structures.

To better understand the formation of color images, the dichromatic reflection model has been introduced by Shafer [9]. The model describes how photometric changes, such as shadows and specularities, influence the RGB-values in an image. On the basis of this model, algorithms have been proposed which are invariant to different photometric phenomena such as shadows, illumination and specularities [10], [11], [12]. The extension to differential photometric invariance has been proposed by Geusebroek et al. [13]. Van de Weijer et al. [14] propose photometric quasi-invariants which have better noise and stability characteristics compared to existing photometric invariants. Combining photometric quasi-invariants with derivative based feature detectors leads to features which can identify various physical causes, e.g. shadow corners and object corners. In this chapter, the theory and practice is reviewed to obtain color invariance such as shading/shadow and illumination invariance incorporated into the color feature detectors.

Two important criteria for color feature detectors are: 1. (*repeatability*) they should be *invariant* (stable) under varying viewing conditions, such as illumination, shading, and highlights; 2. (*distinctiveness*) they should have high *discriminative power*. It has been shown that there exists a trade-off between color invariant models and their discriminative power [10]. For example, color constant derivatives have been proposed [11] which are invariant to all possible light sources assuming a diagonal model for illumination changes. However, such a strong assumption will significantly reduce the discriminative power. For a particular computer vision task that assumes only a few different light sources, color models should be selected which are invariant (only) to these few light sources resulting in an augmentation of the discriminative power of

the algorithm. Therefore, in this chapter, we outline an approach to the selection and weighting of color (invariant) models for discriminatory and robust image feature detection.

Further, although color is important to express saliency [15], the explicit incorporation of color distinctiveness into the design of salient points detectors has been largely ignored. To this end, in this chapter, we review on how color distinctiveness can be explicitly incorporated in the design of image feature detectors [16], [17]. The method is based upon the analysis of the statistics of color derivatives. It will be shown that isosalient derivatives generate ellipsoids in the color derivative histograms. This fact is exploited to adapt derivatives in such a way that equal saliency implies equal impact on the saliency map.

Classifying image features (e.g. edges, corners and T-junctions) is useful for a large number of applications where corresponding feature types (e.g. material edges) from distinct images are selected for image matching while discounting other accidental feature types (e.g. shadow and highlight edges). Therefore, in this chapter, a classification framework is discussed to combine the local differential structure (i.e. geometrical information such as edges, corners and T-junctions) and color invariance (i.e. photometrical information such as shadows, shading, illumination and highlights) in a multi-dimensional feature space [18]. This feature space is used to yield proper rule-based and training-based classifiers to label salient image structures on the basis of their physical nature [19].

In summary, in this chapter, we will review on methods and techniques solving the following important issues in the field of color feature detection: (1) to obtain color invariance such as shading/shadow, illumination invariance, (2) to combine derivatives of the different color channels to compute locale image structures such as edges, corners, circles etc., (3) to select and weight color (invariant) models for discriminatory and robust image feature detection, (4) to improve color saliency to arrive at color distinctiveness (focus-of-attention), (5) to classify the physical nature of image structures such as shadow, highlight and material edges/corners.

This chapter is organized as follows. First, in 1.2, a brief review is given on the various color models and their invariant properties based on the dichromatic reflection model. Further, color derivatives are introduced. In section 1.3, color feature detection is proposed based on the color tensor. Color feature detection on its application to color feature learning, color boosting, and color feature classification is given in sections 1.4, 1.5, and 1.6.

1.2 Color Invariance

In this section, the dichromatic reflection model is explained Shafer [9]. The dichromatic reflection model explains the image formation process and therefore models the photometric changes, such as shadows and specularities. On the basis of this model, methods are discussed containing invariance. In section, 1.2.1

the dichromatic reflection model is introduced. Then, in sections 1.2.2 and 1.2.3, color invariants and color (invariant) derivatives will be explained.

1.2.1 Dichromatic Reflection Model

The Dichromatic Reflection Model [9] divides the light that falls upon a surface into two distinct components: specular reflection and body reflection. Specular reflection is when a ray of light hits a smooth surface at some angle. The reflection of that ray will reflect at the same angle as the incident ray of light. This kind of reflection causes highlights. Diffuse reflection is when a ray of light hits the surface which will be reflected back in every direction.

Suppose we have an infinitesimally small surface patch of some object, and three sensors are used for red, green and blue (with spectral sensitivities $f_R(\lambda)$, $f_G(\lambda)$ and $f_B(\lambda)$) to obtain an image of the surface patch, then the sensor values are [9]:

$$C = m_b(\mathbf{n}, \mathbf{s}) \int_\lambda f_C(\lambda) e(\lambda) c_b(\lambda) d\lambda + m_s(\mathbf{n}, \mathbf{s}, \mathbf{v}) \int_\lambda f_C(\lambda) e(\lambda) c_s(\lambda) d\lambda, \tag{1.1}$$

for $C \in \{R, G, B\}$, and where $e(\lambda)$ is the incident light. Further, $c_b(\lambda)$ and $c_s(\lambda)$ are the surface albedo and Fresnel reflectance, respectively. The geometric terms m_b and m_s are the geometric dependencies on the body and surface reflection component. λ is the wavelength, $\mathbf{n}$ is the surface patch normal, $\mathbf{s}$ is the direction of the illumination source, and $\mathbf{v}$ is the direction of the viewer. The first term in the equation is the diffuse reflection term and the second term is the specular reflection term.

Let's assume white illumination, i.e. all wavelengths within the visible spectrum have similar energy: $e(\lambda) = e$. Further assume that the neutral interface reflection model holds, so that $c_s(\lambda)$ has a constant value independent of the wavelength ($c_s(\lambda) = c_s$). First, we construct a variable that depends only on the sensors and the surface albedo:

$$k_C = \int_\lambda f_c(\lambda) c_b(\lambda) d\lambda. \tag{1.2}$$

Finally, we assume that the following holds:

$$\int_\lambda f_R(\lambda) d\lambda = \int_\lambda f_G(\lambda) d\lambda = \int_\lambda f_B(\lambda) d\lambda = f \tag{1.3}$$

With these assumptions, we have the following equation for the sensor values from an object under white light [11]:

$$C_w = e m_b(\mathbf{n}, \mathbf{s}) k_C + e m_s(\mathbf{n}, \mathbf{s}, \mathbf{v}) c_s f, \tag{1.4}$$

with $C_w \in \{R_w, G_w, B_w\}$.

1.2.2 Color Invariants

To derive that normalized color, given by,

$$r = \frac{R}{R+G+B}, \tag{1.5}$$

$$g = \frac{G}{R+G+B}, \tag{1.6}$$

$$b = \frac{B}{R+G+B}, \tag{1.7}$$

is insensitive to surface orientation, illumination direction and illumination intensity, the diffuse reflection term

$$C_b = em_b(\mathbf{n}, \mathbf{s})k_C \tag{1.8}$$

is used.

By substituting eq. 1.8 in the equations of r, g and b we obtain:

$$r(R_b, G_b, B_b) = \frac{k_R}{k_R + k_G + k_B}, \tag{1.9}$$

$$g(R_b, G_b, B_b) = \frac{k_G}{k_R + k_G + k_B}, \tag{1.10}$$

$$b(R_b, G_b, B_b) = \frac{k_B}{k_R + k_G + k_B}, \tag{1.11}$$

and hence rgb is only dependent on the sensor characteristics and surface albedo. Note that rgb is dependent on highlights (i.e. dependent on specular reflection term of eq. 1.4).

The same argument holds for the $c_1c_2c_3$ color space:

$$c_1(R_b, G_b, B_b) = \arctan(\frac{k_R}{\max\{k_G, k_B\})}), \tag{1.12}$$

$$c_2(R_b, G_b, B_b) = \arctan(\frac{k_G}{\max\{k_R, k_B\})}), \tag{1.13}$$

$$c_1(R_b, G_b, B_b) = \arctan(\frac{k_B}{\max\{k_G, k_R\})}). \tag{1.14}$$

Invariant properties for saturation

$$S(R, G, B) = 1 - \frac{\min\{R, G, B)}{R+G+B}, \tag{1.15}$$

are obtained by substituting the diffuse reflection term into the equation of saturation:

$$S(R_b, G_b, B_b) = 1 - \frac{\min\{k_R, k_G, k_B\}}{(k_R + k_G + k_B)}. \tag{1.16}$$

S is only dependent on the sensors and the surface albedo.

Further, hue

$$H(R, G, B) = \arctan(\frac{\sqrt{3}(G-B)}{((R-G)+(R-B))}), \tag{1.17}$$

is also invariant to surface orientation, illumination direction and intensity:

$$H(R_b, G_b, B_b) = \arctan(\frac{\sqrt{3}em_b(\mathbf{n}, \mathbf{s})(k_G - k_B)}{em_b(\mathbf{n}, \mathbf{s})((k_R - k_g) + (k_R - k_B))}) = \arctan(\frac{\sqrt{3}(k_G - k_B)}{(k_R - k_G) + (k_R - k_B)}). \tag{1.18}$$

In addition, hue is invariant to highlights

$$H(R_w, G_w, B_w) = \arctan(\frac{\sqrt{3}(G_w - B_w)}{(R_w - G_w) + (R_w - B_w)}) =$$

$$= \arctan(\frac{\sqrt{3}em_b(\mathbf{n}, \mathbf{s})(k_G - k_B)}{em_b(\mathbf{n}, \mathbf{s})((k_R - k_G) + (k_R - k_B))}) = \arctan(\frac{\sqrt{3}(k_G - k_B)}{(k_R - k_G) + (k_R - k_B)}). \tag{1.19}$$

System	*Viewpoint*	*Geometry*	*Ill. Color*	*Ill. Int.*	*Highlights*
RGB	-	-	-	-	-
rgb	+	+	-	+	-
Hue	+	+	-	+	+
S	+	+	-	+	-
I	-	-	-	-	-
$c_1c_2c_3$	+	+	-	+	-

Table 1.1: Invariance for different color spaces for varying image properties. A '+' means that the color space is not sensitive to the property, a '-' means that it is.

A taxonomy of color invariant models is shown in table 1.1.

1.2.3 Color Derivatives

Here we describe three color coordinate transformations from which derivatives are taken [20]. The transformation are derived from photometric invariance theory as discussed in the previous section.

For an image $\mathbf{f} = (R, G, B)^T$ the spherical color transformation is given by:

$$\begin{pmatrix} \theta \\ \varphi \\ r \end{pmatrix} = \begin{pmatrix} \arctan(\frac{G}{R}) \\ \arcsin\left(\frac{\sqrt{R^2+G^2}}{\sqrt{R^2+G^2+B^2}}\right) \\ r = \sqrt{R^2 + G^2 + B^2} \end{pmatrix}. \tag{1.20}$$

The spatial derivatives are transformed to the spherical coordinate system by:

$$S(\mathbf{f}_x) = \mathbf{f}_x^s = \begin{pmatrix} r \sin\varphi\, \theta_x \\ r\varphi_x \\ r_x \end{pmatrix} = \begin{pmatrix} \frac{G_x R - R_x G}{\sqrt{R^2+G^2}} \\ \frac{R_x RB + G_x GB - B_x(R^2+G^2)}{\sqrt{R^2+G^2}\sqrt{R^2+G^2+B^2}} \\ \frac{R_x R + G_x G + B_x B}{\sqrt{R^2+G^2+B^2}} \end{pmatrix}. \tag{1.21}$$

The scale factors follow from the Jacobian of the transformation. They ensure that the norm of the derivative remains constant under the transformation, hence $|\mathbf{f}_x| = |\mathbf{f}_x^s|$. In the spherical coordinate system the

derivative vector is a summation of a shadow-shading variant part, $\mathbf{S}_x = (0, 0, r_x)^T$ and a shadow-shading quasi-invariant part, given by $\mathbf{S}_x^c = (r \sin\varphi\theta_x, r\varphi_x, 0)^T$ [20].

The opponent color space is given by:

$$\begin{pmatrix} o1 \\ o2 \\ o3 \end{pmatrix} = \begin{pmatrix} \frac{R-G}{\sqrt{2}} \\ \frac{R+G-2B}{\sqrt{6}} \\ \frac{R+G+B}{\sqrt{3}} \end{pmatrix}. \tag{1.22}$$

For this, the following transformation of the derivatives is obtained:

$$O(\mathbf{f}_x) = \mathbf{f}_x^o = \begin{pmatrix} o1_x \\ o2_x \\ o3_x \end{pmatrix} = \begin{pmatrix} \frac{1}{\sqrt{2}}(R_x - G_x) \\ \frac{1}{\sqrt{6}}(R_x + G_x - 2B_x) \\ \frac{1}{\sqrt{3}}(R_x + G_x + B_x) \end{pmatrix}. \tag{1.23}$$

The opponent color space decorrelates the derivative with respect to specular changes. The derivative is divided into a specular variant part, $\mathbf{O}_x = (0, 0, o3_x)^T$, and a specular quasi-invariant part $\mathbf{O}_x^c = (o1_x, o2_x, 0)^T$.

The hue-saturation-intensity is given by

$$\begin{pmatrix} h \\ s \\ i \end{pmatrix} = \begin{pmatrix} \arctan\left(\frac{o1}{o2}\right) \\ \sqrt{o1^2 + o2^2} \\ o3 \end{pmatrix}. \tag{1.24}$$

The transformation of the spatial derivatives into the hsi-space decorrelates the derivative with respect to specular, shadow and shading variations,

$$H(\mathbf{f}_x) = \mathbf{f}_x^h = \begin{pmatrix} s\ h_x \\ s_x \\ i_x \end{pmatrix} = \begin{pmatrix} \frac{(R(B_x-G_x)+G(R_x-B_x)+B(G_x-R_x))}{\sqrt{2(R^2+G^2+B^2-RG-RB-GB)}} \\ \frac{R(2R_x-G_x-B_x)+G(2G_x-R_x-B_x)+B(2B_x-R_x-G_x)}{\sqrt{6(R^2+G^2+B^2-RG-RB-GB)}} \\ \frac{(R_x+G_x+B_x)}{\sqrt{3}} \end{pmatrix}. \tag{1.25}$$

The shadow-shading-specular variant is given by $\mathbf{H}_x = (0, 0, i_x)^T$ and the shadow-shading-specular quasi-invariant by $\mathbf{H}_x^c = (sh_x, s_x, 0)^T$.

Since the length of a vector is not changed by orthonormal coordinate transformations, the norm of the derivative remains the same in all three representations $|\mathbf{f}_x| = |\mathbf{f}_x^c| = |\mathbf{f}_x^o| = \left|\mathbf{f}_x^h\right|$. For both the opponent color space and the hue-saturation-intensity color space, the photometrically variant direction is given by the $L1$ norm of the intensity. For the spherical coordinate system the variant is equal to the $L2$ norm of the intensity.

1.3 Combining Derivatives

In the previous section, color (invariant) derivatives have been discussed. The question is how to combine these derivatives into a single outcome. A default method to combine edges is to use equal weights for the different color features. This naive approach is used by many feature detectors. For example, to achieve color edge detection, intensity-based edge detection techniques are extended by taking the sum or Euclidean distance from the individual gradient maps [21], [22]. However, the summation of the derivatives computed for the different color channels may result in the cancellation of local image structures [4]. A more principled way is to sum the orientation information (defined on $[0, \pi\rangle$) of the channels instead of adding the direction information (defined on $[0, 2\pi\rangle$). Tensor mathematics provide a convenient representation in which vectors in opposite directions will reinforce one another. Tensors describe the local orientation rather than the direction i.e. the tensor of a vector and its 180° rotated counterpart vector are equal. Therefore, tensors are convenient to describe color derivative vectors and will be used as a basis for color feature detection.

1.3.1 The Color Tensor

Given a luminance image f, the structure tensor is given by [6]

$$\mathbf{G} = \begin{pmatrix} \overline{f_x^2} & \overline{f_x f_y} \\ \overline{f_x f_y} & \overline{f_y^2} \end{pmatrix}, \tag{1.26}$$

where the subscripts indicate spatial derivatives and the bar $\bar{(.)}$ indicates convolution with a Gaussian filter. The structure tensor describes the local differential structure of images, and is suited to find features such as edges and corners [4], [5], [7]. For a multichannel image $\mathbf{f} = \left(f^1, f^2, ..., f^n\right)^T$, the structure tensor is given by

$$\mathbf{G} = \begin{pmatrix} \overline{\mathbf{f}_x \cdot \mathbf{f}_x} & \overline{\mathbf{f}_x \cdot \mathbf{f}_y} \\ \overline{\mathbf{f}_y \cdot \mathbf{f}_x} & \overline{\mathbf{f}_y \cdot \mathbf{f}_y} \end{pmatrix}. \tag{1.27}$$

In the case that $\mathbf{f} = (R, G, B)$, Eq. 1.27 is the color tensor. For derivatives which are accompanied with a weighting function, w_x and w_y, which appoint a weight to every measurement in $\mathbf{f_x}$ and $\mathbf{f_y}$, the structure tensor is defined by

$$\mathbf{G} = \begin{pmatrix} \frac{\overline{w_x^2 \mathbf{f}_x \cdot \mathbf{f}_x}}{\overline{w_x^2}} & \frac{\overline{w_x w_y \mathbf{f}_x \cdot \mathbf{f}_y}}{\overline{w_x w_y}} \\ \frac{\overline{w_y w_x \mathbf{f}_y \cdot \mathbf{f}_x}}{\overline{w_y w_x}} & \frac{\overline{w_y^2 \mathbf{f}_y \cdot \mathbf{f}_y}}{\overline{w_y^2}} \end{pmatrix}. \tag{1.28}$$

1.3.2 Color Tensor-Based Features

In this section, a number of detectors are discussed which can be derived from the weighted color tensor. In the previous section, we described how to compute (quasi) invariant derivatives. In fact, dependent on the task at hand, either quasi-invariants are selected for detection or full invariants. For feature detection

tasks quasi-invariant have been shown to perform best, while for feature description and extraction tasks full-invariants are required [20]. The features in this chapter will be derived for a general derivative $\mathbf{g}_x$. To obtain the desired photometric invariance for the color feature detector, the inner product of $\mathbf{g}_x$ (see Eq. 1.27) is replaced by one of the following:

$$\overline{\mathbf{g_x} \cdot \mathbf{g_x}} = \begin{cases} \overline{\mathbf{f_x} \cdot \mathbf{f_x}} & \text{if no invariance is required} \\ \overline{\mathbf{S}^c_\mathbf{x} \cdot \mathbf{S}^c_\mathbf{x}} \text{ or } \overline{\mathbf{H}^c_\mathbf{x} \cdot \mathbf{H}^c_\mathbf{x}} & \text{for invariant feature detection} \\ \frac{\overline{\mathbf{S}^c_\mathbf{x} \cdot \mathbf{S}^c_\mathbf{x}}}{|\mathbf{f}|^2} \text{ or } \frac{\overline{\mathbf{H}^c_\mathbf{x} \cdot \mathbf{H}^c_\mathbf{x}}}{|\mathbf{s}|^2} & \text{for invariant feature extraction} \end{cases} \tag{1.29}$$

where $\mathbf{s}$ is the saturation.

In section 1.3.2, we describe features derived from the eigenvalues of the tensor. Further, more features are derived from an adapted version of the structure tensor such as the Canny edge detection, in section 1.3.2, and the detection of circular objects in section 1.3.2.

Eigenvalue-Based Features

Two eigenvalues are derived from the eigenvalue analysis defined by

$$\begin{aligned} \lambda_1 &= \tfrac{1}{2}\left(\overline{\mathbf{g}_x \cdot \mathbf{g}_x} + \overline{\mathbf{g}_y \cdot \mathbf{g}_y} + \sqrt{\left(\overline{\mathbf{g}_x \cdot \mathbf{g}_x} - \overline{\mathbf{g}_y \cdot \mathbf{g}_y}\right)^2 + \left(2\overline{\mathbf{g}_x \cdot \mathbf{g}_y}\right)^2}\right) \\ \lambda_2 &= \tfrac{1}{2}\left(\overline{\mathbf{g}_x \cdot \mathbf{g}_x} + \overline{\mathbf{g}_y \cdot \mathbf{g}_y} - \sqrt{\left(\overline{\mathbf{g}_x \cdot \mathbf{g}_x} - \overline{\mathbf{g}_y \cdot \mathbf{g}_y}\right)^2 + \left(2\overline{\mathbf{g}_x \cdot \mathbf{g}_y}\right)^2}\right). \end{aligned} \tag{1.30}$$

The direction of λ_1 points in the direction of the most prominent local orientation

$$\theta = \tfrac{1}{2}\arctan\left(\frac{2\overline{\mathbf{g}_x \cdot \mathbf{g}_y}}{\overline{\mathbf{g}_x \cdot \mathbf{g}_x} - \overline{\mathbf{g}_y \cdot \mathbf{g}_y}}\right). \tag{1.31}$$

The λ's can be combined to give the following local descriptors:

- $\lambda_1 + \lambda_2$ describes the total local derivative energy.
- λ_1 is the derivative energy in the most prominent direction.
- $\lambda_1 - \lambda_2$ describes the line-energy (see [23]). The derivative energy in the prominent orientation is corrected for the energy contributed by the noise λ_2.
- λ_2 describes the amount of derivative energy perpendicular to the prominent local orientation.

The Harris corner detector [24] is often used in the literature. In fact, the color Harris operator H can easily be written as a function of the eigenvalues of the structure tensor

$$\begin{aligned} H\mathbf{f} &= \overline{\mathbf{g}_x \cdot \mathbf{g}_x}\,\overline{\mathbf{g}_y \cdot \mathbf{g}_y} - \overline{\mathbf{g}_x \cdot \mathbf{g}_y}^2 - k\left(\overline{\mathbf{g}_x \cdot \mathbf{g}_x} + \overline{\mathbf{g}_y \cdot \mathbf{g}_y}\right)^2 \\ &= \lambda_1\lambda_2 - k\left(\lambda_1 + \lambda_2\right)^2. \end{aligned} \tag{1.32}$$

Further, the structure tensor of Eq. 1.27 can also be seen as a local projection of the derivative energy on two perpendicular axes [5], [7], [8], namely $\mathbf{u}_1 = \begin{pmatrix} 1 & 0 \end{pmatrix}^T$ and $\mathbf{u}_2 = \begin{pmatrix} 0 & 1 \end{pmatrix}^T$,

$$\mathbf{G}^{\mathbf{u}_1,\mathbf{u}_2} = \begin{pmatrix} \overline{(\mathbf{G}_{x,y}\mathbf{u}_1)\cdot(\mathbf{G}_{x,y}\mathbf{u}_1)} & \overline{(\mathbf{G}_{x,y}\mathbf{u}_1)\cdot(\mathbf{G}_{x,y}\mathbf{u}_2)} \\ \overline{(\mathbf{G}_{x,y}\mathbf{u}_1)\cdot(\mathbf{G}_{x,y}\mathbf{u}_2)} & \overline{(\mathbf{G}_{x,y}\mathbf{u}_2)\cdot(\mathbf{G}_{x,y}\mathbf{u}_2)} \end{pmatrix} \tag{1.33}$$

in which $\mathbf{G}_{x,y} = \begin{pmatrix} \mathbf{g}_x & \mathbf{g}_y \end{pmatrix}$. From the Lie group of transformation several other choices of perpendicular projections can be derived [5], [7]. They include feature extraction for circle, spiral and star-like structures.

The star and circle detector is given as an example. It is based on $\mathbf{u}_1 = \frac{1}{\sqrt{x^2+y^2}} \begin{pmatrix} x & y \end{pmatrix}^T$ which coincide with the derivative pattern of a circular patterns and $\mathbf{u}_2 = \frac{1}{\sqrt{x^2+y^2}} \begin{pmatrix} -y & x \end{pmatrix}^T$ which denotes the perpendicular vector field which coincides with the derivative pattern of star-like patterns. These vectors can be used to compute the adapted structure tensor with Eq. 1.33. Only the elements on the diagonal have non zero entries and are equal to

$$\mathbf{H} = \begin{pmatrix} \overline{\frac{x^2}{x^2+y^2}\mathbf{g}_x\cdot\mathbf{g}_x} + \overline{\frac{2xy}{x^2+y^2}\mathbf{g}_x\cdot\mathbf{g}_y} + \overline{\frac{y^2}{x^2+y^2}\mathbf{g}_y\cdot\mathbf{g}_y} & 0 \\ 0 & \overline{\frac{x^2}{x^2+y^2}\mathbf{g}_y\cdot\mathbf{g}_y} - \overline{\frac{2xy}{x^2+y^2}\mathbf{g}_x\cdot\mathbf{g}_y} + \overline{\frac{y^2}{x^2+y^2}\mathbf{g}_x\cdot\mathbf{g}_x} \end{pmatrix}, \tag{1.34}$$

here λ_1 describes the amount of derivative energy contributing to circular structures and λ_2 the derivative energy which describes a star-like structure.

Curvature is another feature which can be derived from an adaption of the structure tensor. For vector data the equation for the curvature is given by

$$\kappa = \frac{\overline{w^2\mathbf{g}_v\cdot\mathbf{g}_v} - \overline{w^2}\cdot\overline{\mathbf{g}_w\cdot\mathbf{g}_w} - \sqrt{\left(\overline{w^2}\cdot\overline{\mathbf{g}_w\cdot\mathbf{g}_w} - \overline{w^2\mathbf{g}_v\cdot\mathbf{g}_v}\right)^2 + 4\overline{w^2}\cdot\overline{w\mathbf{g}_v\cdot\mathbf{g}_w}^2}}{2\overline{w^2}\cdot\overline{w\mathbf{g}_v\cdot\mathbf{g}_w}} \tag{1.35}$$

in which $\mathbf{g}_v$ and $\mathbf{g}_w$ are the derivatives in gauge coordinates.

Color Canny Edge Detection

We now introduce the Canny color edge detector based on eigenvalues. The algorithm consists of the following steps:

1. Compute the spatial derivatives, $\mathbf{f}_x$, and combine them if desired into a quasi-invariant as discussed in section 1.2.3.

2. Compute the maximum eigenvalue (Eq. 1.30) and its orientation (Eq. 1.31).

3. Apply non-maximum suppression on λ_1 in the prominent direction.

To illustrate the performance, the results of Canny color edge detection for several photometric quasi-invariants is shown in Fig. 1.1. The image is recorded in 3 RGB-colors with the aid of the SONY XC-003P

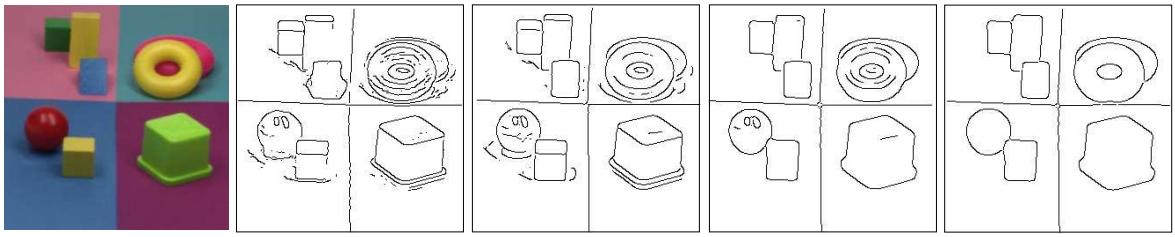


Figure 1.1: (a) Input image with Canny edge detection based on successively, (b) luminance derivative, (c) RGB derivatives, (d) the shadow-shading quasi-invariant, (e) the shadow-shading-specular quasi-invariant.

CCD color camera (3 chips) and the Matrox Magic Color frame grabber. Two light sources of average day-light color are used to illuminate the objects in the scene. The digitization was done in 8 bits per color. The results show that the luminance-based Canny, Fig. 1.1b, misses several edges which are correctly found by the RGB-based method, Fig. 1.1c. Also the removal of spurious edges by photometric invariance is demonstrated. In Fig. 1.1d the edge detection is robust to shadow and shading changes and only detects material and specular edges. In Fig. 1.1e only the material edges are depicted.

Circular Object Detection

In this section, the combination of photometric invariant orientation and curvature estimation is used to detect circles robust against varying imaging conditions such as shadows and illumination changes.

The following algorithm is introduced for the invariant detection of color circles [20]:

1. Compute the spatial derivatives, $\mathbf{f}_x$, and combine them if desired into a quasi-invariant as discussed in section 1.2.3.

2. Compute the local orientation, Eq. 1.31, and curvature, Eq. 1.35.

3. Compute the hough space [25], $H\left(R, x^0, y^0\right)$, where R is the radius of the circle and x^0 and y^0 indicate the center of the circle. The computation of the orientation and curvature reduces the number of votes per pixel to one. Namely, for a pixel at position $\mathbf{x} = \left(x^1, y^1\right)$,

$$\begin{aligned} R &= \tfrac{1}{\kappa} \\ x^0 &= x^1 + \tfrac{1}{\kappa}\cos\theta \\ y^0 &= y^1 + \tfrac{1}{\kappa}\sin\theta. \end{aligned} \tag{1.36}$$

   Each pixel will vote by means of its derivative energy $\sqrt{\mathbf{f}_x \cdot \mathbf{f}_x}$.

4. Compute the maxima in the hough space. These maxima indicate the circle centers and the radii of the circle.

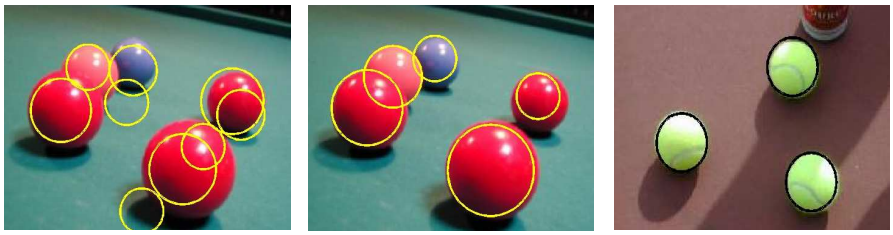


Figure 1.2: (a) Detected circles based on luminance, (b) detected circles based on shadow-shading-specular quasi-invariant, (c) detected circles based on shadow-shading-specular quasi-invariant.

To illustrate the performance, the results of the circle detection are given in Fig. 1.2. Images have been recorded by the Nikon Coolpix 950, a commercial digital camera of average quality. The images have size 267x200 pixels with JPEG compression. The digitization was done in 8 bits per color. It is shown that the luminance-based circle detection is sensitive to photometric variations as nine circles are detected before the five balls were extracted. For the circle detector based on the (shadow-shading-specular) quasi-invariant the five most prominent peaks in the hough space (not shown here) correspond to the radii and center points of the circles found. In Fig. 1.2.c, an outdoor example with a shadow partially covering the objects (tennis balls) is given. The detector finds the right circular objects and hence performs well, even under severe varying imaging conditions such as shading and shadow, and geometrical changes of the tennis balls.

1.4 Color Feature Detection: Fusion of Color Derivatives

In the previous section, various image feature detection methods have been discussed to extract locale image structures such as edges, corners and circles. As there are many color invariant models available, the inherent difficulty is how to automatically select the weighted subset of color models producing the best result for a particular task. In this section, we outline how to select and weight color (invariant) models for discriminatory and robust image feature detection.

To achieve proper color model selection and fusion, we discuss a method that exploits non-perfect correlation between color models or feature detection algorithms derived from the principles of diversification. As a consequence, an optimal balance is obtained between repeatability and distinctiveness. The result is a weighting scheme which yields maximal feature discrimination [18], [19].

1.4.1 Problem Formulation

The measuring of a quantity u can be stated as:

$$u = E\left(u\right) \pm \sigma_u \tag{1.37}$$

where $E(u)$ is the best estimate for u (e.g. the average value) and σ_u represents the uncertainty or error in the measurement of u (e.g. the standard deviation). Estimates of a quantity u, resulting from N different methods, may be constructed using the following weighting scheme:

$$E(u) = \sum_{i}^{N} w_i E(u_i), \tag{1.38}$$

where $E(u_i)$ is the best estimate of a particular method i. Simply taking the weighted average of the different methods allows features from very different domains to be combined.

For a function $u(u_1, u_2, \cdots, u_N)$ depending on N correlated variables, the propagated error σ_u is:

$$\sigma_u(u_1, u_2, \cdots, u_N) = \sum_{i=1}^{N}\sum_{j=1}^{N} \frac{\partial u}{\partial u_i}\frac{\partial u}{\partial u_j} \operatorname{cov}(u_i, u_j), \tag{1.39}$$

where $\operatorname{cov}(u_i, u_j)$ denotes the covariance between two variables. From this equation it can be seen that if the function u is non-linear, the resulting error, σ_u, depends strongly on the values of the variables $u_1, u_2, \cdots, u_N$. Since equation (1.38) involves a linear combination of estimates, the error of the combined estimate is only dependent on the covariances of the individual estimates. So through equation (1.39), we have established that the proposed weighting scheme guarantees robustness, in contrast to possible, more complex, combination schemes.

Now we left with the problem of determining the weights w_i in a principled way. In the next section, we will propose such an algorithm.

1.4.2 Feature Fusion

When using equation (1.38), the variance of the combined color models can be found through equation (1.39):

$$\sigma_u^2 = \sum_{i=1}^{N}\sum_{j=1}^{N} w_i w_j \operatorname{cov}(u_i, u_j), \tag{1.40}$$

or, equivalently,

$$\sigma_u^2 = \sum_{j=1}^{N} w_i^2 \sigma_{u_i}^2 + \sum_{i=1}^{N}\sum_{j\neq i} w_i w_j \operatorname{cov}(u_i, u_j), \tag{1.41}$$

where w_i denotes the weight assigned to color channel i, u_i denotes the average output for channel i, σ_u denotes the standard deviation of quantity u in channel i, and $\operatorname{cov}(u_i, u_j)$ corresponds to the covariance between channel i and j.

From equation (1.41), it can be seen how diversification over various channels can reduce the overall variance due to the covariance that may exist between channels. The Markowitz selection model [26] is a mathematical method for finding weights that achieve an optimal diversification. The model will minimise the variance for a given expected estimate for quantity u or will maximise the expected estimate for a given

variance σ_u. The model defines a set of optimal u and σ_u pairs. The constraints of this selection model are given as follows:

$$\text{minimise } \sigma_u, \tag{1.42}$$

for the formula described in equation (1.38). The weights are constrained by the following conditions:

$$\sum_{i=1}^{N} w_i = 1, \tag{1.43a}$$

$$-1 <= w_i <= 1, i = 1, \cdots, N. \tag{1.43b}$$

The constraint in equations (1.43a) ensures that all channels are fully allocated and constraint (1.43b) limits the search space for w_i.

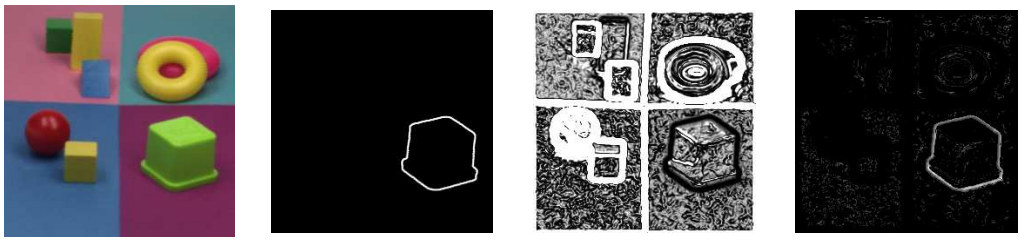
This model is quadratic with linear constraints and can be solved by linear programming ([27]). When σ_u is varied parametrically, the solutions for this system will result in mean-variance pairs representing different weightings of the feature channels. The pairs which maximise the expected u v.s. σ_u or minimise the σ_u v.s. expected u, define the optimal frontier. They form a curve in the mean-variance plane and the corresponding weights are optimal.

A point of particular interest on this curve is the point which has the maximal ratio between the expected combined output $E(u)$ and the expected variance σ_u^2. This point has the weights for which the combined feature space has offers the best trade-off between repeatability and distinctiveness.

In summary, the discussed selection model is used to arrive at a set of weights to combine different color models into one feature. The expected value of this feature $E(u)$ is the weighted average of its component expected values. The standard deviation of this combined feature will be less than or equal to the weighted average of the component standard deviations. When the component colors or features are not perfectly correlated, the weighted average of the features will have a better variance to output ratio than the individual components on their own. New features or colors can always safely be added, the ratio will never deteriorate, because zero weights can be assigned to components that will not improve the ratio.

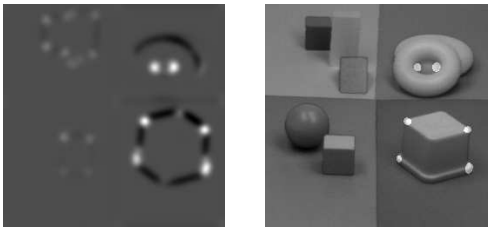
1.4.3 Corner Detection

The purpose is to detect corners by learning. Our aim is to arrive at an optimal balance between color invariance (repeatability) and discriminative power (distinctiveness). In the context of combining feature detectors, in particular in the case of color (invariant) edge detection, a default method to combine edges is to use equal weights for the different color features. This naive approach is used by many feature detectors. Instead of experimenting with different ways to combine algorithms, in this section, we use the principled method, outlined in the previous section, on the basis of the benefits of diversification. Since our method is based on learning, we need a set of training examples. The problem of corners is, however, that there are



(a) (b) (c) (d)

Figure 1.3: (a) Lab image, (b) groundtruth for learning edges. Input image for the edge and corner detection, on the left the edge is indicated for the learning algorithm. (c) The χ-squared error of the transformed image and the predicted expected value, here the edges have a very low intensity, (d) the local signal-to-noise ratio for the transformed image. The edges have a higher ratio.



(a) (b)

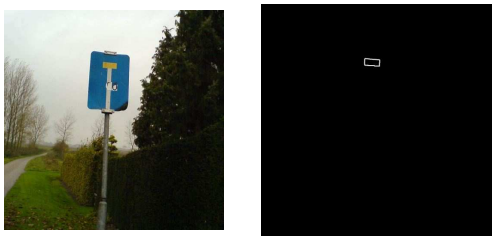
Figure 1.4: (a) Results of the Harris corner detector, (b) corners projected on input image. The results of the Harris corner detector, trained on the lower right cube.

always few pixels at a corner, making it hard to create a training set. We circumvent this problem by training on the edges, the first order derivatives, where many more pixels are located.

Since the structure tensor (equation 1.28) and the derived Harris operator (equation 1.32) are based on spatial derivatives, we will train the weighting vector $\mathbf{w}$ on edges. This will allow for a much simpler collection of training points. So the weights are trained with the spatial derivatives of the color channels as input. The resulting weights are then put in the $\mathbf{w}$ weights vector of the Harris operator.

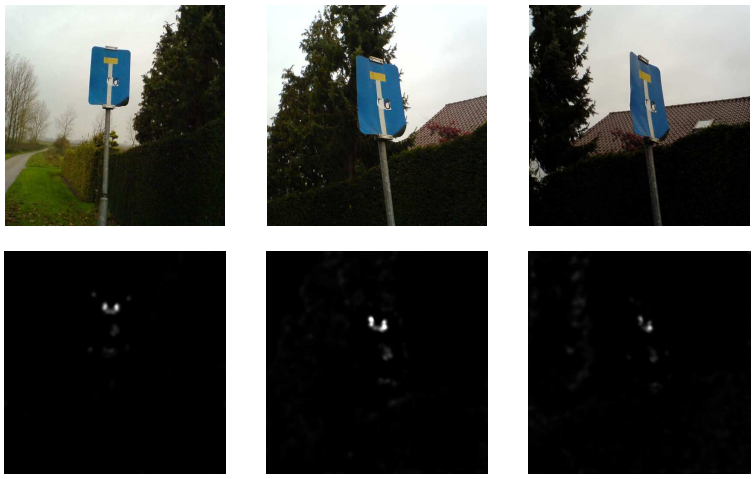
To illustrate the performance of the corner detector based on learning, the first experiment was done on the image of section 1.3 which have recorded in 3 RGB-colors with the aid of the SONY XC-003P CCD color camera. The weights were trained on the edges of the green cube (see figures 1.3.a and 1.3.b). The edges were trained on the first order derivatives in all color spaces. The results of applying these weights to the same image are shown in figures 1.3.c and 1.3.d. The edge is especially visible in signal-to-noise image. Using the weights learned on the edges with the Harris operator, according to equation (1.32), the corners of the green cube stand particularly out (see figure 1.4).

Another experiment is done on images taken from an outdoor object i.e. a traffic sign (see figure 1.5). The weights were trained on one image and tested on images of the same object while varying the viewpoint.



(a) (b)

Figure 1.5: The input image for the edge and corner detection. (a) the training image, (b) the trained edges.



(a) (b) (c)

Figure 1.6: (a) Corners for image 1, (b) corners for image 2, (c) corners for image 3. The results of the Harris corner detector trained on red-blue edges.

Again the edges were defined by the first order derivative in gauge coordinates. The results of the Harris operator are shown in figure 1.6. The corner detector performs well even under varying viewpoints and illumination changes. Note that the learning method results in an optimal balance between repeatability and distinctiveness.

1.5 Color Feature Detection: Boosting Color Saliency

So far, we have outlined how to obtain color invariant derivatives for image feature detection. Further, we discussed how to learn a proper set of weights to yield proper color model selection and fusion of feature detection algorithms.

In addition, it is known that color is important to express saliency [15], [17]. To this end, in this section, we review on how color distinctiveness can be explicitly incorporated in the design of image feature detectors [16], [17]. The method is based upon the analysis of the statistics of color derivatives. When studying the statistics of color image derivatives points of equal frequency form regular structures [17]. Van de Weijer

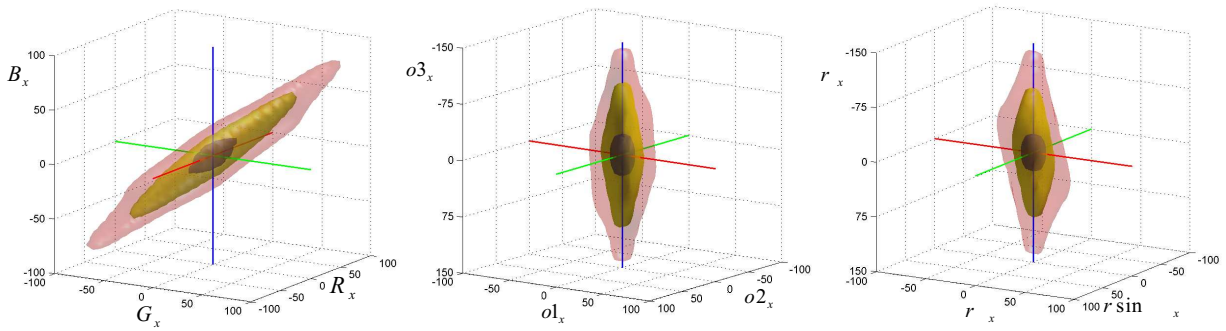


Figure 1.7: The histograms of the distribution of the transformed derivatives of the Corel image database in respectively the (a) RGB coordinates, (b) the opponent coordinates and (c) the spherical coordinates. The three planes correspond with the isosalient surfaces which contain (from dark to light) respectively 90%, 99%, 99.9% of the total number of pixels.

et. al. [17], proposes a color saliency boosting function which is based on transforming the color coordinates and using the statistical properties of the image derivatives. The RGB color derivatives are correlated. By transforming the RGB color coordinates to other systems, photometric events in images can be ignored as discussed in section 1.2 where is was shown that the spatial derivatives are separated into photometrical variant, and invariant parts. For the purpose of color saliency, the three different color spaces are evaluated i.e. the spherical color space (equation 1.21), the opponent color space (equation 1.23), and the hsi color space (equation 1.25). In these decorrelated color spaces only the photometric axes are influenced by these common photometric variations.

The statistics of color images are shown for the Corel database [28], which consists of 40,000 images (black and white images were excluded). In Fig. 1.7, the distributions (histograms) of the first order derivatives, f_x, are given for the various color coordinate systems.

When the distributions of the transformed image derivatives are observed from Fig. 1.7, regular structures are generated by points of equal frequency i.e. isosalient surfaces. These surfaces are formed by connecting the points in the histogram that occur the same number of times. The shapes of the isosalient surfaces correspond to to ellipses. The major axis of the ellipsoid coincides with the axis of maximum variation in the histogram i.e. the intensity axes. Based on the observed statistics, a saliency measure can be derived in which vectors with an equal information content have an equal effect on the saliency. This is called the ”color saliency boosting function”. It is obtained by deriving a function that describes the isosalient surfaces.

More precisely, the ellipsoids are equal to

$$\left(\alpha h_x^1\right)^2 + \left(\beta h_x^2\right)^2 + \left(\gamma h_x^3\right)^2 = \left|\mathbf{\Lambda} h\left(\mathbf{f}_x\right)\right|^2 \tag{1.44}$$

	$\mathbf{f}_x$	$\lvert\mathbf{f}_x\rvert_1$	$\mathbf{f}_x^{\tilde{s}}$	$\tilde{\mathbf{S}}_x^c$	$\mathbf{f}_x^{\tilde{o}}$	$\tilde{\mathbf{O}}_x^c$	$\mathbf{f}_x^h$	$\mathbf{H}_x^c$
Λ_{11}	0.577	1	0.851	0.856	0.850	0.851	0.858	1
Λ_{22}	0.577	-	0.515	0.518	0.524	0.525	0.509	0
Λ_{33}	0.577	-	0.099	0	0.065	0	0.066	0

Table 1.2: *The diagonal entries of* Λ *for the Corel data set computed for Gaussian derivatives with* $\sigma = 1$.

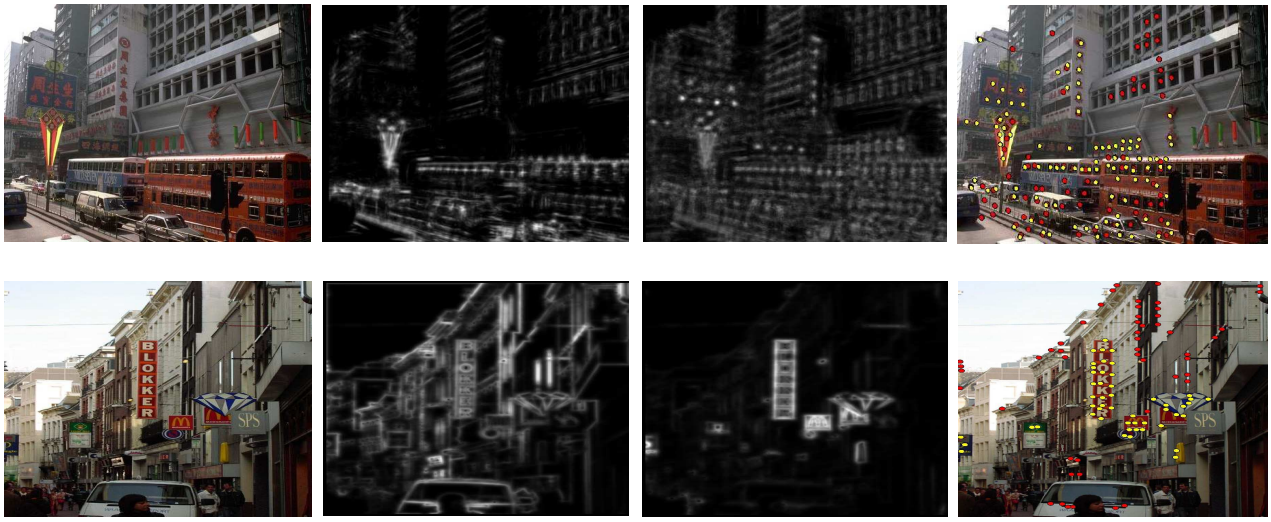


Figure 1.8: In columns respectively, (a) input image, (b) RGB-gradient based saliency map, (c) color boosted saliency map, and (d) the results with red dots (lines) for gradient-based method and yellow dots (lines) for salient points after color saliency boosting.

then the following holds

$$p\left(\mathbf{f}_x\right) = p\left(\mathbf{f}_x^{'}\right) \leftrightarrow \left|\mathbf{\Lambda} h\left(\mathbf{f}_x\right)\right| = \left|\mathbf{\Lambda}^T h\left(\mathbf{f}_x^{'}\right)\right|, \tag{1.45}$$

where $\mathbf{\Lambda}$ is a 3x3 diagonal matrix with $\mathbf{\Lambda}_{11} = \alpha$, $\mathbf{\Lambda}_{22} = \beta$, and $\mathbf{\Lambda}_{33} = \gamma$. Λ is restricted to $\Lambda_{11}^2 + \Lambda_{22}^2 + \Lambda_{33}^2 = 1$. The desired saliency boosting function is obtained by

$$g\left(\mathbf{f}_x\right) = \mathbf{\Lambda} h\left(\mathbf{f}_x\right). \tag{1.46}$$

By a rotation of the color axes followed by a rescaling of the axis, the oriented isosalient ellipsoids are transformed into spheres, and thus vectors of equal saliency a re transformed into vectors of equal length.

Before color saliency boosting can be applied, the Λ-parameters have to be initialized by fitting ellipses to the histogram of the data set. The results for the various transformations are summarized in Table 1.2. The relation between the axes in the various color spaces clearly confirms the dominance of the luminance axis in the RGB-cube, since Λ_{33}, the multiplication-factor of the luminance axis, is much smaller than the color-axes multiplication factors, Λ_{11} and Λ_{22}. After color saliency boosting, there is an increase in information context, see [17] for more details.

To illustrate the performance of the color boosting method, Fig. 1.8 show the results before and after

saliency boosting. Although focus point detection is already an extension from luminance to color, black-and-white transition still dominate the result. Only after boosting the color saliency, the less interesting black-and-white structures in the image are ignored and most of the red Chinese signs are found.

1.6 Color Feature Detection: Classification of Color Structures

In section 1.2, we showed that color models may contain a certain amount of invariance to the imaging process. From the taxonomy of color invariant models, shown in table 1.1, we now discuss a classification framework to detect and classify local image structures based on photometrical and geometrical information [18]. The classification of local image structures (e.g. shadow vs. material edges) is important for many image processing and computer vision tasks (e.g. object recognition, stereo vision, 3D reconstruction).

By combining the differential structure and reflectance invariance of color images, local image structures are extracted and classified into one of the following types: (1) shadow-geometry edges, corners, and T-junctions; (2) highlight edges, corners, and T-junctions (3) material edges, corners, and T-junctions. First, for detection, the differential nature of the local image structures is derived. Then, color invariant properties are taken into account to determine the reflectance characteristics of the detected local image structures. The geometrical and photometrical properties of these salient points (structures) are represented as vectors in a multi-dimensional feature space. Finally, a classifier is built to learn the specific characteristics of each class of image structures.

1.6.1 Combining Shape and Color

By combining geometrical and photometrical information, we are able to specify the physical nature of salient points. For example, to detect highlights we need to use both a highlight invariant color space, and one (or more) highlight variant spaces. It has already been shown in table 1.1 that hue is invariant to highlights. Intensity I and saturation S are not invariant. Further, a highlight will yield a certain image shape: a local maximum (I) and a local minimum (S). These local structures are detected by differential operators as discussed in section 1.2. Therefore, in the brightness image we are looking for a local maximum. The saturation at a highlight is lower than its surroundings, yielding a local minimum. Finally, for hue the values will be near zero at that location. In this way, a 5-dimensional feature vector is formed by combining the color space HSI and the differential information each location in an image.

The same procedure holds for the detection of shadow-geometry/highlight/material edges, corners, and T-junctions. The features used to detect shadow-geometry edges are first-order derivate applied on both the RGB and the $c_1c_2c_3$ color channels. Further the second-order derivative is only applied on the RGB color image. To be precise, in this section, we use the curvature gauge to characterize local structures that are

only characterized by their second order structure. It is a coordinate system on which the Hessian becomes diagonal yielding the (p, q)-coordinate system. The two eigenvectors of the Hessian are κ_1 and κ_2 and are defined by:

$$\kappa_1 = f_{xx} + f_{yy} - \sqrt{(f_{xx} + f_{yy})^2 + 4f_{xy}^2}, \tag{1.47}$$

$$\kappa_2 = f_{xx} + f_{yy} + \sqrt{(f_{xx} + f_{yy})^2 + 4f_{xy}^2}. \tag{1.48}$$

To obtain the appropriate density distribution of feature values in feature space, classifiers are built to learn the density functions for shadow-geometry/highlight/material edges, corners, and T-junctions.

In this section, the learning-based classification approach is taken as proposed by Gevers [19]. This approach is adaptive as the underlying characteristics of image feature classes are determined by training. In fact, the probability density functions of the local image structures are learned by determining the probability that an image patch under consideration is of a particular class (e.g. edge, corner or T-junctions). If two image patches share similar characteristics (not only the same color, but also the same gradient size, and curvature) both patches are represented by the same point in feature space. These points are represented by a $(n \times d)$-matrix, in which d depends on the number of feature dimensions and n on the number of training samples.

Then, the density function $p(\mathbf{x}|\omega)$ is computed where $\mathbf{x}$ represents the data of the pixel under consideration, and ω is the class to be determined. From the data and training points, the parameters of the density function $p(\mathbf{x}|\omega)$ are estimated. We use a single Gaussian and multiple Gaussians (Mixture of Gaussians - MoG). Besides this, the k-Nearest Neighbour method is used.

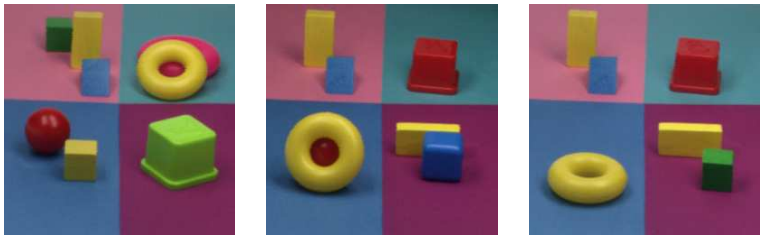
1.6.2 Experimental Results

In this section, the results are given to classify the physical nature of salient points by learning. A separate set of tests is computed for each classifier (i.e. Gaussian, Mixture of Gaussians and k-Nearest Neighbour).

The images are recorded in 3 RGB-colors with the aid of the SONY XC-003P CCD color camera (3 chips) and the Matrox Magic Color frame grabber. Two light sources of average day-light color are used to illuminate the objects in the scene. The digitization was done in 8 bits per color. Three examples of the five images are shown in figure 1.9. For all experiments, $\sigma = 2$ is used for the Gaussian smoothing parameter of the differential operators.

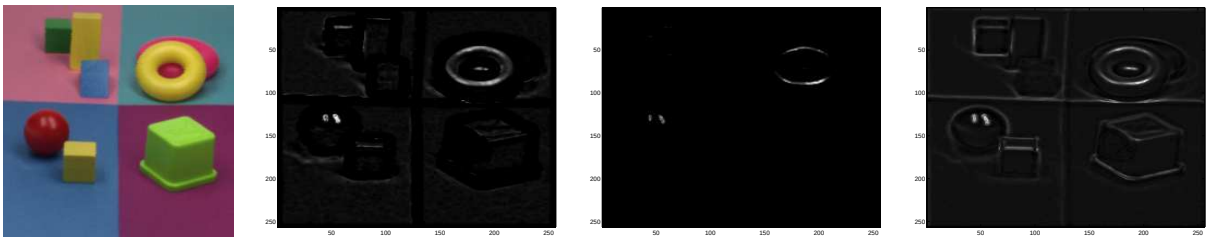
The classifiers are trained using all but one image. This last image is used as a test image. In this way, the test image is not used in the training set.

In all experiments, a total of three Gaussian components is used for the MoG classifier. $k = 3$ for the k-NN classifier.



(a) (b) (c)

Figure 1.9: (a) Example image 1, (b) Example image 2, (c) Example image 3. The images are recorded in 3 RGB-colors with the aid of the SONY XC-003P CCD color camera (3 chips) and the Matrox Magic Color frame grabber. Two light sources of average day-light color are used to illuminate the objects in the scene.



(a) (b) (c) (d)

Figure 1.10: (a) Test image, (b) Gaussian classifier, (c) Mixture of Gaussians, (e) k-nearest neighbour. Based on the (training) efficiency and accuracy of the results, the Gaussian or MoG are most appropriate for highlight detection.

1.6.3 Detection of Highlights

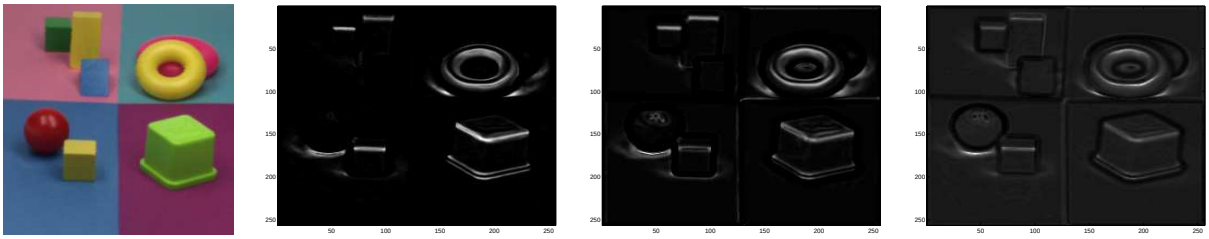
The features used for highlight detection are κ_1 and κ_2 applied on the HSB color channels yielding a 5-dimensional space for each image point.

Gaussian: The Gaussian method performs well to detect highlight, see figure 1.10.b. Most of the highlights are detected. However, only a few false positives are found e.g. bar-shaped structures. This is because the reflectance at these structures are composed of a portion of specular reflection.

Mixture of Gaussians: The MoG method gives slightly better results than the Gaussian method, see figure 1.10.c. For this method, the highlighted bars, found by the Gaussian method, are discarded.

k-Nearest Neighbour: This method performs slightly worse as opposed the detection method based a single Gaussian, see figure figure 1.10.d. The problem with the highlighted bars is still present.

Summary: The detection methods based on a single Gaussian as well as on the MoG are well suited for highlight detection.



(a) (b) (c) (d)

Figure 1.11: (a) Test image, (b) Gaussian classifier, (c) Mixture of Gaussians, (e) k-nearest neighbour. For geometry-shadow detection, the best results are obtained by the Gaussian method. MoG and k-Nearest Neighbour perform a bit less.

1.6.4 Detection of Geometry/Shadow Edges

The features that are used to detect geometry/shadow edges are the first-order derivatives which are applied on both the RGB and the $c_1c_2c_3$ color models. Further, the second order derivative is applied on the RGB color images.

Gaussian: The detection method, based on a single Gaussian, performs well, see figure figure 1.11.b. Most of the geometry-shadow edges have been detected. Further, there are nearly no false positives present. Besides that, the recall is very high.

Mixture of Gaussians: The method based on a Mixture of Gaussians has a similar performance as the Gaussian method, see figure figure 1.11.c. For a few instances, however, material edges are detected.

k-Nearest Neighbour: The accuracy of the method is somewhat lower than the other two classifiers, see figure figure 1.11.d. Still most of the geometry and shadow edges are detected correctly.

Summary For geometry-shadow detection, the best results are obtained by the Gaussian method. MoG and k-Nearest Neighbour perform a bit less.

1.6.5 Detection of Corners

The first-order derivative (f_w) and second-order derivative (f_{vv}) of the RGB color space are used for corner learning and classification. To determine the thresholds for corner detection, various settings have been examined. The results are shown in figure 1.12. The threshold providing the highest accuracy, and subsequently used in our experiments, is 0.75.

From table 1.3, it is observed that the k-Nearest Neighbour classifier provides the highest performance. Examining the precision/recall graphs for the three classifiers reveals that this method provide good performance.

Further, the MoG performs slightly better than the single Gaussian method.

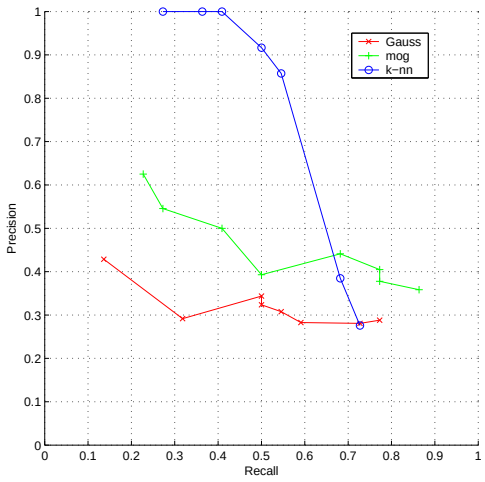


Figure 1.12: Precision/Recall graph for the classifiers of corners

Classifier	Precision	Recall
Gaussian	34,9%	56,6%
Mixture of Gaussians	77,8%	52,8%
k-Nearest Neighbour	83,9%	58,5%

Table 1.3: Classifiers and their performance for corner detection

Summary The k-Nearest Neighbour classifier provides the best performance to detect corners. Although the recall of all three methods is similar, the precisions of the k-NN classifier is higher.

1.7 Conclusion

In this chapter, we have discussed methods and techniques in the field of color feature detection. In particular, the focus was on the following important issues: (1) color invariance, (2) combining derivatives, (3) fusion of color models, (4) color saliency boosting, and (5) classifying image structures.

To this end, the dichromatic reflection model has been outlined first. The dichromatic reflection model explains the RGB-values variations due to the image formation process. From the model, various color models are obtained showing a certain amount of invariance. Then, color (invariant) derivatives have been discussed. These derivatives include quasi-invariants which have proper noise and stability characteristics. To combine color derivatives into a single outcome, the color tensor has been used instead of taking the sum or Euclidean distance. Tensors are convenient to describe color derivative vectors. Based on the color tensor, various image feature detection methods have been introduced to extract locale image structures such as edges, corners and circles. The experimental results of Canny color edge detection for several photometric quasi-invariants showed stable and accurate edge detection. Further, a proper model has been discussed to

select and weight color (invariant) models for discriminatory and robust image feature detection. The use of the fusion model is important as there are many color invariant models available. In addition, we used color to express saliency. It has been shown that after color saliency boosting, (less interesting) black-and-white structures in the image are ignored and more interesting color structures have been detected. Finally, a classification framework has been outlined to detect and classify local image structures based on photometrical and geometrical information. High classification accuracy is obtained by simple learning strategies.

In conclusion, this chapter provides a survey on methods solving important issues in the field of color feature detection. We hope that these solutions on low-level image feature detection will aid the continuing challenging task of handling higher level computer vision task such as object recognition and tracking.

Bibliography

[1] R. Haralick and L. Shapiro, *Computer and Robot Vision, volume II*. Addison-Wesley, 1992.

[2] C. Schmid, R. Mohr, and C. Bauckhage, "Evaluation of interest point detectors," *International Journal of Computer Vision*, vol. 37, no. 2, pp. 151–172, 2000.

[3] J. Shi and C. Tomasi, "Good features to track," in *IEEE conference on Computer Vision and Pattern Recognition*, 1994.

[4] S. Di Zenzo, "Note: A note on the gradient of a multi-image," *Computer Vision, Graphics, and Image Processing*, vol. 33, no. 1, pp. 116–125, 1986.

[5] J. Bigun, "Pattern recognition in images by symmetry and coordinate transformations," *Computer Vision and Image Understanding*, vol. 68, no. 3, pp. 290–307, 1997.

[6] J. Bigun, G. Granlund, and J. Wiklund, "Multidimensional orientation estimation with applications to texture analysis and opitcal flow," *IEEE trans. on pattern analysis and machine intelligence*, vol. 13, no. 8, pp. 775–790, 1991.

[7] O. Hansen and J. Bigun, "Local symmetry modeling in multi-dimensional images," *pattern Recognition Letters*, vol. 13, pp. 253–262, 1992.

[8] J. van de Weijer, L. van Vliet, P. Verbeek, and M. van Ginkel, "Curvature estimation in oriented patterns using curvilinear models applied to gradient vector fields," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 23, no. 9, pp. 1035–1042, 2001.

[9] S. Shafer, "Using color to seperate reflection components," *COLOR research and application*, vol. 10, pp. 210–218, Winter 1985.

[10] T. Gevers and H. Stokman, "Robust histogram construction from color invariants for object recognition," *IEEE Trans. on Pattern Analysis and Machine Intelligence (PAMI)*, vol. 26, no. 1, pp. 113–118, 2004.

[11] T. Gevers and A. W. M. Smeulders, "Color based object recognition," *Pattern Recognition*, vol. 32, pp. 453–464, March 1999.

[12] G. Klinker and S. Shafer, "A physical approach to color image understanding," *Int. Journal of Computer Vision*, vol. 4, pp. 7–38, 1990.

[13] J. Geusebroek, R. van den Boomgaard, A. Smeulders, and H. Geerts, "Color invariance," *IEEE Trans. Pattern Analysis Machine Intell.*, vol. 23, no. 12, pp. 1338–1350, 2001.

[14] J. van de Weijer, T. Gevers, and J. Geusebroek, "Edge and corner detection by photometric quasi-invariants," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 27, no. 4, pp. 625–630, 2005.

[15] L. Itti, C. Koch, and E. Niebur, "Computation modeling of visual attention," *Nature Reviews Neuroscience*, vol. 2, pp. 194–203, March 2001.

[16] J. van de Weijer and T. Gevers, "Boosting color saliency in image feature detection," in *Int'l Conf. Computer Vision and Pattern Recognition*, (San Diego, CA, USA), 2005.

[17] J. van de Weijer, T. Gevers, and A. Bagdanov, "Boosting color saliency in image feature detection," *IEEE Trans. Pattern Analysis and Machine Intelligence*, vol. 28, no. 1, pp. 150–156, 2006.

[18] T. Gevers and H. Stokman, "Classification of color edges in video into shadow-geometry, highlight, or material transitions," *IEEE Trans. on Multimedia*, vol. 5, no. 2, pp. 237–243, 2003.

[19] T. Gevers and F. Aldershoff, "Color feature detection and classification by learning," in *Proceedings IEEE International Conference on Image Processing (ICIP)*, 2005.

[20] J. van de Weijer, T. Gevers, and A. Smeulders, "Robust photometric invariant features from the color tensor," *IEEE Trans. Image Processing*, vol. 154, no. 1, 2006.

[21] S. D. Zenzo, "A note on the gradient of a multi-image," *Computer Vision, Graphics, and Image Processing*, vol. 33, pp. 116–125, 1986.

[22] G. Sapiro and D. L. Ringach, "Anisotropic diffusion of multivalued images with applications to color filtering," *IEEE Transactions Pattern Analysis and Machine Intelligence*, vol. 5, no. 11, pp. 1582–1586, 1996.

[23] G. Sapiro and D. Ringach, "Anisotropic diffusion of multivalued images with applications to color filtering," *IEEE Trans. Image Processing*, vol. 5, pp. 1582–1586, Oct 1996.

[24] C. Harris and M. Stephens, "A combined corner and edge detector," in *Proc. 4th Alvey Vision Conf.*, vol. 15, pp. 147–151, 1988.

[25] D. H. Ballard, "Generalizing the Hough transform to detect arbitrary shapes," *Pattern Recognition*, vol. 12, no. 2, pp. 111–122, 1981.

[26] H. Markowitz, "Portfolio selection," *Journal of Finance*, vol. 7, 1952.

[27] P. Wolfe, "The simplex method for quadratic programming," *Econometrica*, vol. 27, no. 3, 1959.

[28] C. Gallery, "www.corel.com."

Index