

Color, shape and texture

- T. Gevers, *Color in Image Search Engines*. In Principles of Visual Information Retrieval (M.S. Lew, ed.), London: Springer-Verlag, 2001.
- *Shading and Color Models*.
- *Shape Analysis*.
- A.D. Bimbo, Visual Information Retrieval, Chapter: *Image Retrieval by Similarity*, pp. 134-159, Morgan Kaufmann, 1999.



Color in Image Search Engines

Theo Gevers

Faculty of Science,
University of Amsterdam,
Kruislaan 403, 1098 SJ Amsterdam, The Netherlands

Intelligent Sensory Information Systems
Faculty of Science
University of Amsterdam
The Netherlands



Published in
Principles of Visual Information Retrieval,
editor Michael S. Lew,
Springer-Verlag, London,
ISBN 1-852333-381-2, February 2001

Contents

1	Introduction	3
2	Color Fundamentals	4
3	Color Appearance	6
3.1	The Light Source	7
3.2	The Object	8
3.3	The Observer	9
3.3.1	Trichromacy Theory	9
3.3.2	Opponent Theory	10
3.3.3	Color Response of the Eye	10
3.3.4	The CIE 2° Standard Observer	11
3.3.5	Summary	11
4	Colorimetry	12
4.1	<i>XYZ</i> System	12
4.2	<i>RGB</i> System	14
4.3	<i>HSI</i> System	15
4.4	<i>YIQ</i> and <i>YUV</i> System	16
4.5	Color Order Systems	16
4.5.1	Munsell's Color Order System	16
4.5.2	CIELAB Color Order Systems	16
4.5.3	Color Difference	17
4.5.4	Summary	18
5	Color Invariance	18
5.1	Reflection from Inhomogeneous Dielectric Materials	19
5.2	Reflectance with White Illumination	19
5.2.1	Invariance for Matte Surfaces	20
5.2.2	Invariance for Shiny Surfaces	21
5.3	Color Constancy	22
6	Color System Taxonomy	23
6.1	Grey-Value System	24
6.2	<i>RGB</i> Color System	24
6.3	<i>rgb</i> Color System	25
6.4	<i>XYZ</i> Color System	25
6.5	<i>xyz</i> Color System	26
6.6	<i>U*V*W*</i> Color System	26
6.7	<i>L*a*b*</i> Color System	28
6.8	<i>I1I2I3</i> Color System	28
6.9	<i>YIQ</i> and <i>YUV</i>	29
6.10	<i>HSI</i> Color System	30
6.11	Color Ratio's	30
7	Color and Image Search Engines	31

Intelligent **S**ensory **I**nformation **S**ystems
Faculty of Science
University of Amsterdam
Kruislaan 403
1098 SJ Amsterdam
The Netherlands
tel: +31 20 525 7516
fax: +31 20 525 7490
e-mail: gevers@wins.uva.nl

1 Introduction

Can you imagine an existence without color? Indeed, color does not only add beauty to objects but does give more information about objects as well. Further, color information often facilitates your life, like in traffic, or in sport to identify your favorite team when both teams wear dark shirts.

After more than three hundred years since Newton established the fundamentals of color in his "Opticks" (1704) [18], color has been involved in many fields ranging from pure scientific, to abstract art, and applied areas. For example, the scientific work on light and color resulted in the quantum mechanics started and elaborated by Max Planck, Albert Einstein and Niels Bohr. In painting, Albert Munsell provided the theoretical basis in his "A Color Notation" (1905) [16] on which most painters derived their notions about color ordering. The emotional and psychological influence of color on humans has been studied by Goethe in its famous book "Farbenlehre" (1840) [11]. Further, the value of the biological and therapeutic effect of light and color has been analyzed, and views on color from folklore, philosophy and language have been articulated by Descartes, Schopenhauer, Hegel and Wittgenstein.

Today, with the growth and popularity of the World Wide Web, a new application field is born through the tremendous amount of visual information, such as images and video's, which has been made accessible publicly. With this new application area, color has returned to the center of interest of a growing number of scientists, artists and companies. Aside from decorating and advertising potentials for Web-design, color information has already been used as a powerful tool in content-based image and video retrieval. Various color based image search schemes have been proposed based on various representation schemes such as color histograms, color moments, color edge orientation, color texture, and color correlograms, [1], [2], [5], [20], [27], [28]. These image representation schemes have been created on the basis of *RGB*, and other color systems such as *HSI* and $L^*a^*b^*$ [1], [2], [5], [20], [27]. In particular, the Picasso [3] and ImageRover [21] system use the $L^*u^*v^*$ color space for image indexing and retrieval. The QBIC system [5] evaluates similarity of global color properties using histograms based on a linear combination of the *RGB* color space. MARS [15] is based on the $L^*a^*b^*$ color space which is (like $L^*u^*v^*$) a perceptual uniform color space. The PicToSeek system [10] is based on color models robust to a change in viewing direction, object geometry and illumination. Hence, the choice of color systems is of great importance for the purpose of proper image retrieval. It induces the equivalence classes to the actual retrieval algorithm. However, no color system can be considered as universal, because color can be interpreted and modeled in different ways. Each color system has its own set of color models, which are the parameters of the color system. Color systems have been developed for different purposes: 1. display and printing processes: *RGB*, *CMY*; 2. television and video transmission efficiency: *YIQ*, *YUV*; 3. color standardization: *XYZ*; 4. color uncorrelation: $I_1 I_2 I_3$; 5. color normalization and representation: *rgb*, *xyz*; 6. perceptual uniformity: $U^*V^*W^*$, $L^*a^*b^*$, $L^*u^*v^*$; 7. and intuitive description: *HSI*, *HSV*. With this large variety of color systems, the inevitable question arises which color system to use for which kind of image retrieval application. To this end, criteria are required to classify the various color systems for the purpose of content-based image retrieval. Firstly, an important criterion is that the color system is independent of the underlying imaging device. This is required when images in the image database are recorded by different imaging devices such as scanners, camera's and camrecorder (e.g. images on Internet). Another prerequisite is that the color system should exhibit perceptual uniformity meaning that numerical distances within the color space can be related to human perceptual differences. This is important when images are to be retrieved which should be visually similar (e.g. stamps, trademarks and paintings databases). Also, the transformation needed to compute the color system should be linear. A non-linear transformation may introduce instabilities with respect to noise causing poor retrieval accuracy.

Further, the color system should be composed of color models which are understandable and intuitive to the user. Moreover, to achieve robust and discriminative image retrieval, color invariance is an important criterion. In general, images and video's are taken from objects from different viewpoints. Two recordings made of the same object from different viewpoints will yield different shadowing, shading and highlighting cues changing the intensity data fields considerably. Moreover, large differences in the illumination color will drastically change the photometric content of images even when they are taken from the same object. Hence, a proper retrieval scheme should be robust to imaging conditions discounting the disturbing influences of a change in viewpoint, object pose, and illumination.

In this chapter, the aim is to provide a survey on the basics of color, color models and ordering systems, and the state-of-the-art on color invariance. For the purpose of color-based image retrieval, our aim is to provide a taxonomy on color systems composed according to the following criteria:

- Is the color system device independent
- Is the color system perceptual uniform
- Is the color system linear
- Is the color system intuitive
- Is the color system robust against varying imaging conditions
 - Invariant to a change in viewing direction;
 - Invariant to a change in object geometry;
 - Invariant to a change in the direction of the illumination;
 - Invariant to a change in the intensity of the illumination;
 - Invariant to a change in the spectral power distribution (SPD) of the illumination.

The color system taxonomy can be used to select the proper color system for a specific application. For example, consider an image database of textile printing samples (e.g. curtains). The application is to search for samples with similar color appearances. When the samples have been recorded under the same imaging conditions (i.e. camera, illumination and sample pose), a perceptual uniform color systems (e.g. $L^*a^*b^*$) is most suitable. When the lightning conditions are different between the recordings, a color invariant system is most appropriate eliminating the disturbing influences such as shading, shadows and highlights.

This chapter is organized as follows. Firstly, the fundamentals of color will be given in Section 2. As color appearance depends on the light, object and observer, we will study and analyze this triplet in detail in Section 3. Further, in Section 4, the standardization of the light-object-observer triplet will be presented. A survey on color invariance is presented in Section 5. The taxonomy of color systems is given in Section 6. Color and image search engines are discussed in Section 7. Conclusions will be drawn in Section 8. Further, the various color systems and their performance can be experienced within the PicToSeek and Pic2Seek systems on-line at: <http://www.wins.uva.nl/research/isis/zomax/>.

2 Color Fundamentals

Fundamentally, color is part of the electromagnetic spectrum with energy in the range from 380- to 780-nm wavelength. This is the part of the spectrum to which the human eyes are sensitive. For color measurements, it is often restricted to 400-700 nm. This visible part of the

spectrum is perceived as the colors from violet through indigo, blue, green, yellow, and orange to red. This continuous spectrum of colors is obtained when a beam of sunlight is split through a glass prism. Wavelength is the physical difference between the various regions of the spectrum. The unit length of the wavelength is the nanometer (nm). Each wavelength value within the visible band corresponds to a distinct color. Most of the colors that we see do not correspond to one single wavelength but are a mixture of wavelengths from the electromagnetic spectrum, where the amount of energy at each wavelength is represented by a spectral energy distribution. For example, the energy emitted by a white-light source contains a nearly equal amount of all wavelengths, see Figure 1.a. When white light shines upon an object, some wavelengths are reflected and some are absorbed. For example, a green object reflects light with wavelengths primarily around the 500 nm range. The other wavelengths are absorbed, see Figure 1.b.

Color can be described in different ways than only physically by its wavelength characteristics. A system that describes color, is called a color system. Color can be defined and modeled in different ways and each color system has its own set of color models (usually three). The three color features which are usually taken to describe the visual (intuitive) sensation of color are hue, saturation and lightness. The **hue** corresponds with the dominant wavelength of the spectral energy distribution and is the color we see when viewing the light. For instance, if one sees a color with predominant high wavelengths then the color is interpreted as red. Thus, the hue is the kind of color, like red, green, blue, yellow, cyan, etc. **Saturation** corresponds to the excitation purity of the color and is defined as the proportion of pure light with respect to white light needed to produce the color. High saturation denotes little dilution. In other words, saturation is the richness of a hue and therefore denotes how pure a hue is. **Lightness** is related to the intensity (or energy) of the light reflected from objects. Further, **brightness** corresponds to the intensity of the light coming from light sources (i.e. self-luminous objects). The higher the emitted intensity, the brighter the color appears. The description of spectral information in terms of these three color models is derived as follows. Consider the energy emitted by a white light source as shown in Figure 1.a. All wavelengths contribute more or less equally to the total energy. When the color of the light source has a dominant wavelength around the 506 nm part of the spectrum, as shown in Figure 1.b, then the perceived color will be perceived as greenish. Let the power energy of the dominant wavelength be denoted by E_H and the wavelengths contributing to produce white light of intensity by E_W , see Figure 1.b. Hence, the hue is green corresponding to the dominant wavelength E_H of 506 nm where the energy power is given by E_H . Further, saturation depends on the difference between E_H and E_W : the larger the difference the more pure the color is. The lightness is equal to the area under the curve of the spectral power distribution (i.e. total energy).

In conclusion, the relative spectral power (light) gives complete information about the color (i.e. color fingerprints). However, the human eye is incapable of analyzing color into its spectral components. An intuitive approximation of the spectral information of color by humans is in terms of hue, saturation and lightness. Although, the approximation of the spectral curves by these three visual attributes suffice for a large number of problems, there are still a number of problems for which spectral information is essential and where human interpretation fails such as standards for color measurement, color matches under different illumination, and color mixture analysis.

The spectral power distribution of the light from objects (for example 1.b) is the result of different complex factors such as the light source and material characteristics. In the next section, these complex factors will be discussed.

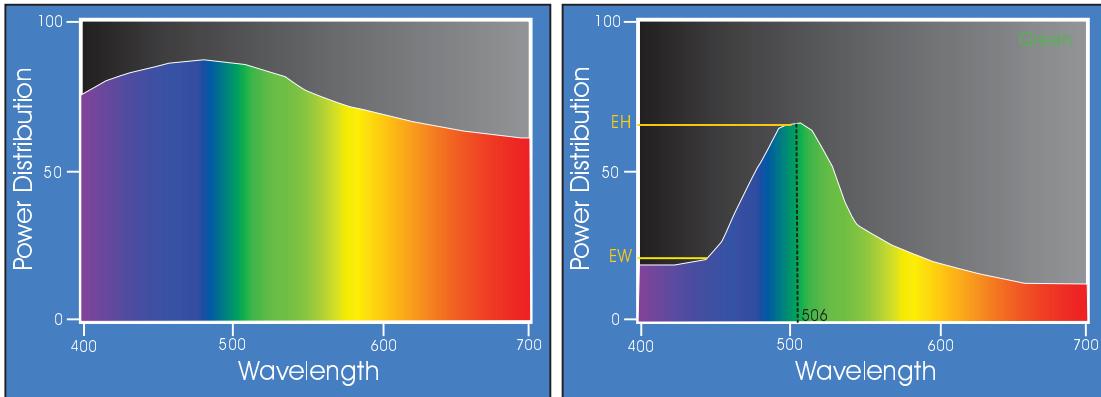


Figure 1: *a.* Relative spectral power distribution of a white light source. *b.* Spectral power distribution of the light reflected from a green sample.

3 Color Appearance

Let's follow the light emitted by a light source. Through gas discharge (e.g. fluorescent lamps) or by heating up a material (the filament of the lamp), the light source generates light illuminating an object. Some part of the spectrum of the light is reflected from the object and is subsequently measured by an observer such as our light-sensitive eyes or by a color camera. The measured light is then sent to our brain (or computer) where the color of the light is observed (interpreted). The light-object-observer triplet depends on very complex factors: the observed color may vary with a change in the intensity and energy distribution of the light source, material characteristics, mode of viewing, the observer and so on.

Let's take a closer look at the triplet light-object-observer. As stated above, the traveling of light starts with the **light source**. It is known that the color of the light source has a large influence on the observed color. Examples are the spotlights installed in theaters or disco's. A substantial change in spotlight color will subsequently change the color of the dresses of the dancers. In contrast, people have a large degree of color constancy: the ability of observing the same color under different lightning conditions. For example, an object illuminated by daylight is perceived the same as when the object is illuminated by a candle light.

The second major component is the **object** or **sample** itself. The object color determines the observed color by reflecting (opaque or reflecting materials), absorbing (transparent materials), or absorbing and transmitting (translucent materials) parts of the incident light. Also the viewing mode influences the observed color. A change in viewing position of the camera will change the color and intensity field of the object color with respect to shading and highlighting cues. Further, material characteristics of the object, in terms of structure, gloss and metallic effects, influence the observed color.

Thirdly, the light-sensitive detectors of the **observer**, such as the human eyes or color (ccd) camera's, determine which color will be observed. It often occurs that different human observers will judge the color of an object differently even when the object is seen under the same viewing conditions. Similarly, two recording made from the same object with two different camera's (i.e. with different color filters) but under the same imaging conditions, may yield different pictures.

Hence, the observed color of an object depends on complex set of imaging conditions. To get more insight in the imaging process, it is essential to understand the basics of color measurement. Therefore, in this section, we will subsequently focus on the three major components: the light source in Section 3.1, the object characteristics and viewing mode in Section 3.2, and the

observer (eye and camera) in Section 3.3.

3.1 The Light Source

The main light source is the sun. Further, artificial light sources exist generating light by gas discharge (neon, argon or xenon) such as fluorescent lamps, or by heating up material (e.g. the filament of a lamp). Light produced by different light sources may vary with respect to their spectral power distribution (SPD), which is the amount of radiant power at each wavelength of the visible spectrum. Instruments exist for measuring the radiation power at each wavelength. To make the SPD independent of the amount of light, the spectral power distribution is computed with respect to the radiation of light at a single wavelength (usually 560 nm which is given the value 100). The relative spectral power distribution of a light source is denoted by $E(\lambda)$.

Color temperature is commonly used to express the relative SPD of light sources. Color temperatures correspond to the temperature of a heated black body radiator. The color of the blackbody radiator changes with temperature. For example, the radiator changes from black at 0 K (Kelvin), to red at about 1000 K, white at 4500 K to bluish white at about 6500 K. An incandescent lamp would give a color temperature of 2900 K and a fluorescent lamp about 4100 K. Historically, color temperature has been introduced to match and describe (old fashioned) light sources such as candles and incandescent lamps, generating light in a similar way as the black body due their thermal radiation. However, modern light sources such as fluorescent lamps often do not properly match a color temperature. These light sources have so-called correlated color temperatures instead of exact color temperatures.

The most important light source is the **sun**. The color temperature of the sun may vary during the time of the day (e.g. reddish at sunrise and bluish at noon). Further, the weather plays also an important role in the determination of the color of the sun. For example, on a clear day the color temperature is about 5500 K. The color of the sun changes with the day and month of the year due to latitude and atmospheric conditions. Hence, the color of the sun may vary between color temperatures of 2000 K to 10000 K (indoor north sky daylight). For standardization of colorimetric measurements, in 1931, the international lighting commission (CIE), recommended that the average daylight has the color temperature of 6500 K and has been specified as the standard illuminant D65. Also other standard illuminants have been provided such as D50, D55 and D75. Note the difference between light sources and illuminants. In contrast to real light sources, illuminants may not be physically realisable but are given by numerical values determining the SPD. Before illuminant D65, standard illuminant C has been used as the average daylight. In contrast to the D series, illuminant C can be reproduced by filtering a tungsten filament lamp. However, real daylight has more ultraviolet than illuminant C. This ultraviolet band (380-400 nm) has been specified in D65. In Figure 2, the relative spectral power distribution are given for illuminant A, C and D65. The spectral power distribution of illuminant A (black body with a color temperature of 2856 K) is similar to that of a incandescent lamp.

Artificial light sources have been used in a large variety of places for illuminating, for example, indoor buildings and outdoor roads. The most important light sources are the fluorescent lamps based on gas-discharge of rare gasses such as neon and xenon, or metallic vapors such as mercury and sodium vapor. The filtered xenon arc lamp approximates the D65 illuminant (i.e. average daylight) the closest from all existing light sources. The xenon arc lamp has a high amount of continuum in contrast to other discharge lamps which have short peaks in their SPD corresponding to the characteristics of the excited molecules (gas dependent). Finally, standard illuminants have been recommended by the CIE denoted by illuminant F1-F12 corresponding to SPD of different type of fluorescent lamps.

In conclusion, color temperatures are used to describe the relative spectral power distribu-

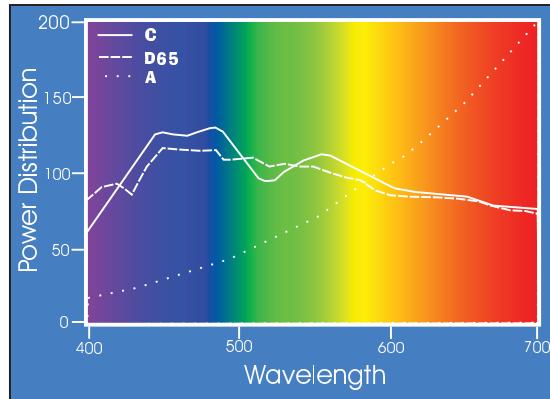


Figure 2: a. Relative spectral power distribution of illuminant A, C, and D65.

tion of light sources. The CIE recommended different SPD's as standards such as illuminant A (incandescent light), D65 and C (average daylight), and F1-F12 (fluorescent lamps). Manufacturers are in search of lamps having a SPD similar to that of average daylight D65. Filtered xenon light is the closest approximation but is very expensive. Fluorescent lamps are cheap and mostly used for the lightning in stores. However, their SPD differ significantly from daylight. This causes the problem that the color of clothes may differ significantly if you wear it indoors or outdoors (under daylight).

3.2 The Object

In this chapter, colored materials are called objects or samples. Objects may consist of different materials such as plastics, glass, wood, textile, paper, coating, metals, brick, etc. Objects can be divided into three major groups. 1. Transparent objects, where some part of the light is absorbed and some part goes unscattered through the sample (e.g. sunglasses, windows, bottles etc), 2. Translucent objects, which absorb, transmit and scatter parts of the light (e.g. lamp panels and shades), 3. Opaque objects, where objects absorb and reflect a portion of the light and no light is transmitted (e.g. paper, textile and walls). Because most of the material that surround us is opaque, in the chapter, we will focus on opaque materials.

Objects are characterized by the percentage of light that they reflect (or transmit) at each wavelength of the visible spectrum. The amount of reflected light from an object depends on the color of the material. The amount of light reflected from an object is computed with respect to that of a white standard. The ratio of the two reflected values is the reflectance factor determining the so-called relative spectral reflectance curve, a number from 0 to 1 (or 0 to 100 in percentage). Thus the relative spectral reflectance curve is not absolute but relative to a white reference and is denoted by $S(\lambda)$. Spectral reflectance curves can be obtained by measurement instruments such as spectrophotometers. The reflectance curve can be seen as the fingerprint of the object's color. In Figure 3.a the spectral curves are shown for green and orange paint samples. The green paint sample absorbs the blue and red parts of the visible spectrum. Further, the orange paint sample absorbs the green and blue part of the visible spectrum. The SPD of a reflecting object is calculated by adding the product of the spectral power distribution of the illuminant and the reflectance (transmittance) of the object at each wavelength of the visible spectrum:

$$P(\lambda) = E(\lambda)S(\lambda) \quad (1)$$

where $P(\lambda)$ is the spectral power in the light reflected by the object with reflectance $S(\lambda)$ under illuminant $E(\lambda)$.

Figure 3 illustrates this concept. The spectral power distribution reflected from a green sample with reflectance given in Figure 3.a when the light illuminant has the spectral power distribution of Figure 3.b is shown in Figure 3.c.

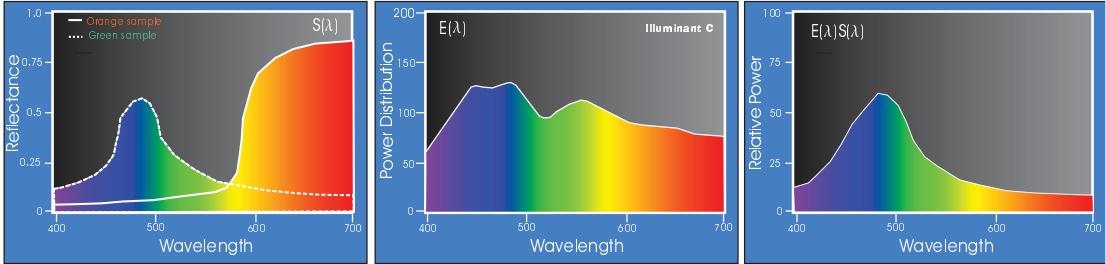


Figure 3: a. Spectral reflectance curves of a green and orange sample. b. Relative spectral distribution of the power from illuminant C. c. Spectral power distribution of the light reflected from a green sample under illuminant C.

In conclusion, the color of an object can be measured objectively (i.e. independent of the illuminant) and is given by the relative spectral reflectance curve. The color reflected from an object is the product of SPD of the illuminant and the spectral reflectance of the object and is computed by $P(\lambda) = E(\lambda)S(\lambda)$. In the next section, the measured SPD of $P(\lambda)$ will be transformed into three numbers corresponding to the numbers we use to describe the color of the object.

3.3 The Observer

The third component of the triplet is the observer. The observer measures light coming directly from a light source $E(\lambda)$ or light which has been reflected (or transmitted) from objects in the scene $P(\lambda)$. The observer can be a color camera or the human eyes. For the human eye, the retina contains two different types of light-sensitive receptors, called *rods* and *cones*. Rods are more sensitive to light and are responsible for vision in twilight. There are theories stating that rods also play a little role in color vision. The cones are responsible for color vision and consist of three types of receptors sensitive to long (red), middle (green) and short (blue) wavelengths. The human eyes have been studied by many scientists. There are two theories which have been widely accepted and used in practice: the trichromacy and opponent theory. In fact, the two theories are compleutive. It has been recognized that both views have their value.

In this section, the trichromacy and opponent theories are outlined. Further, the spectral sensitivities of the different receptors (cones) of the eye are given.

3.3.1 Trichromacy Theory

The comprehension of color perception started with Newton in 1666, the year in which he dispersed (white) light with a glass prism. The result was the amazing discovery that white light is composed of a full range of colors. In 1801, the English scientist Thomas Young suggested the theory that just three primary additive colors were sufficient to produce all colors. The trichromacy theory has been expanded by the German scientist Hermann von Helmholtz and

is usually called the Young-Helmholtz approach. They suggested that the human eye perceives color by the stimulation of three pigments in the cones of the retina. So the perceived color by the human eye has three degrees of freedom. By taking three arbitrary colors from the electromagnetic spectrum, the range of perceived colors which is obtained by the mixture of these three colors with corresponding intensities, is called the *color gamut* and the colors themselves are called *primary colors*. The trichromacy theory has been confirmed only in 1960, where three types of receptors have been identified in the retina. The maximum responses of these receptors correspond to blue (near 440 nm), green (near 540 nm) and red (near 590 nm).

The trichromacy theory has found its way in different applications such as color camera's, camrecorders and video monitors. For example, a T.V. tube is composed of a large array of triangular dot patterns of electron-sensitive phosphor. Each dot in a triad is capable of producing light in one of the primary colors. When each dot in a triad is activated by electrons it emits light, and the amount of light depends upon the strength of the electron stream. The three primary colors from each phosphor triad are added together and the result will be a color because the phosphor dots are very small. The color of a triangular dot pattern depends upon the ratio of the primary color intensities.

3.3.2 Opponent Theory

The opponent color theory started at about 1500 when Leonardo da Vinci came to the conclusion that colors are produced by the mixture of yellow and green, blue and red, white and black. Arthur Shopenhauer noted the same opposition of red-green, yellow-blue and white-black. This opponent color theory has been completed by Edwald Hering concluding that the working of the eye is based on the three kinds of opposite colors. An example of opponent color theory is the so-called after-image. Looking for a while at a green sample will cause a red after-image (excluding yellow and blue). Focusing on the chromatic channels (i.e. red-green and blue-yellow), they are opponent in two different ways. First, as mentioned above, no color seems to be a mixture of both members of any opponent pair. Secondly, each member of an opponent pair exhibits the other. In other words, by adding a balanced proportion of green and red, a hue will be produced which is neither greenish nor reddish. The opponent color theory has been confirmed in 1950 where opponent color signals were detected in the optical connection between eye and brain.

Modern theories combine the trichromacy and opponent color theory; the process starts by light entering the eye, which is detected by trichromacy cones in the retina, and is further processed into three opponent signals on their way to the brain. More recent developments are in the Retinex theory proposed by Edwin Land. Experiments show that people have a considerable amount of color constancy (i.e. color are perceived the same even under different illumination). Unfortunately, all theories discussed so far are constrained and are incapable of explaining the effects of, for example, influences of intensities/colors adjacent to the sample, intensity and color distribution of the light source, mode of viewing and so on.

3.3.3 Color Response of the Eye

We have seen that the light from a light source $E(\lambda)$ as well as light reflected by an object $S(\lambda)$ can be determined with the aid of physical measurement instruments. However, the sensation of a human observer can not be measured by an objective instrument. Therefore, the spectral sensitivities of the human eyes are measured only indirectly as follows.

Experiments have been conducted on human observers without any vision anomalies. The observers were asked to match a test light, consisting of only one wavelength, by adjusting the energy level of three separate primary lights. The three primary lights were additively mixed to match the test light of one wavelength. At each wavelength the amount of energy was recorded for the three primary colors yielding the so-called color matching functions. Different triplets of

primaries can be used to get different color matching functions matching the visible spectrum. Color matching functions can be transformed to those which are obtained by other primaries. Historically, above experiments have been carried by W.D. Wright (7 observers) and J. Guild (10 observers) using different primary colors: 460, 530 and 650 nm (Wright) and 460, 543 and 630 nm (Guild). The viewing angle was 2° , corresponding to the viewing of a sample of about 0.4 inches at reading distance of 10 inches, in which the light illuminates only the fovea. Later experiments have been conducted with a viewing angle of 10° , corresponding to the viewing of a sample of about 1.9 inches at reading distance of 10 inches, by Stiles and Burch (about 50 observers). The tricolor functions of the human eye are shown in Figure 4.a.

3.3.4 The CIE 2° Standard Observer

The complication of the color matching functions so far is that a negative amount of at least one of the primaries was necessary to produce the full set of spectral colors, see Figure 4.a. A negative amount was accomplished by adding one of the primaries to the test spectral light. However, a negative amount is inconvenient for computational reasons. To this end, the CIE recommended mathematical transformations based on three primary standards X , Y and Z . These primary standards are not real but imaginary. The reason to choose these imaginary primary standards are: 1. No negative values in the color matching functions, 2. The primaries X , Y and Z enclose all possible colors 3. The Y primary color corresponds with intensity.

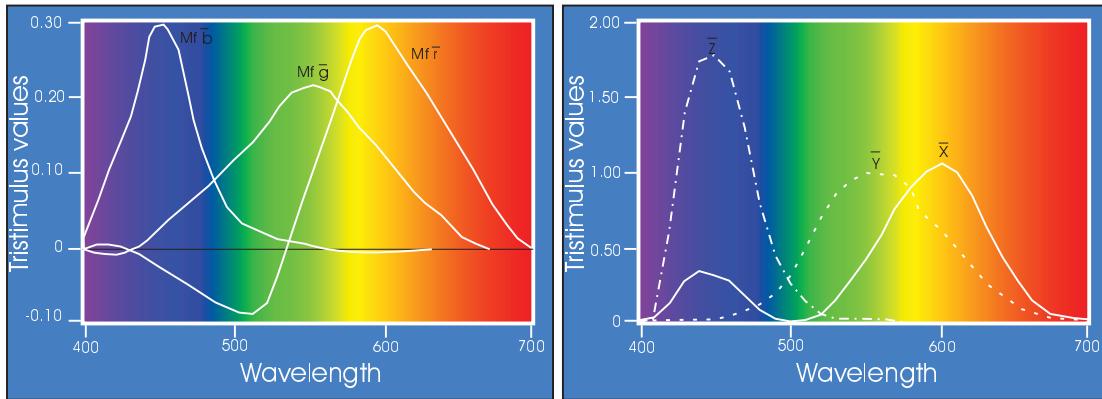


Figure 4: a. Color matching functions \bar{r} , \bar{g} and \bar{b} of the human observer. b. Color matching functions \bar{x} , \bar{y} and \bar{z} from the standard observer 2° (CIE 1931).

This resulted in the CIE color matching functions denoted by \bar{x} , \bar{y} and \bar{z} giving the amount of each of the primary colors required to match a color of one watt of radiant power for each wavelength, see Figure 4.b. The curves correspond to the spectral sensitivity curves of the three receptors in the human eyes. Note that \bar{x} correspond to the spectral luminance factor of the eye. Depending on the viewing angle, again two versions exist: the standard observer - CIE 1931, 2° (shorten to the 2° standard observer), and the standard observer - CIE 1931 10° (cf. 10° standard observer). The standard observer - CIE 1931 10° has been introduced to correspond better to the visual matching of larger samples.

3.3.5 Summary

In conclusion, the spectral sensitivities of the different cone types of the human retina have been determined indirectly by testing on the additive mixture of three primary colors. The

CIE recommends two sets of standard spectral curves (also called color matching functions) depending on the angle an observer views a sample: 2° standard observer (corresponding to the viewing of a sample of about 0.4 inches at reading distance of 10 inches) and 10° standard observer (corresponding to the viewing of a sample of about 1.9 inches at reading distance of 10 inches). In the next section, the matching functions are used to express a color spectrum into three numbers.

4 Colorimetry

Standardization of the light-object-observer triplet is required to measure objectively the imaging process. As discussed in Section 3.1, different light sources exist with different spectral power distributions $E(\lambda)$. To get comparable results, CIE recommended various illuminants. Further, in Section 3.2, the light reflected by objects $S(\lambda)$ can be measured. Then, the reflected color of the object can be determined by $P(\lambda) = E(\lambda)S(\lambda)$. Further, the observer perceives color in terms of three color signals based on the trichromacy theory can be modeled by:

$$R = \int_{\lambda} E(\lambda)S(\lambda)f_R(\lambda)d\lambda \quad (2)$$

$$G = \int_{\lambda} E(\lambda)S(\lambda)f_G(\lambda)d\lambda \quad (3)$$

$$B = \int_{\lambda} E(\lambda)S(\lambda)f_B(\lambda)d\lambda \quad (4)$$

where the tristimulus values are obtained by adding the product of the SPD of the light source $E(\lambda)$, the reflectance (or transmittance) factor of the object $S(\lambda)$ and the color matching functions $f_C(\lambda)$ for $C \in \{R, G, B\}$ of the observer (eye or camera) at each wavelength of the visible spectrum. Depending on the color matching functions (\bar{r} , \bar{g} and \bar{b} or \bar{x} , \bar{y} and \bar{z}), the sum of each equation is equal to the amount of primaries to match the sample.

This section is outlined as follows. First, in Section 4.1 the calculation of the tristimulus values is discussed for the CIE XYZ color system. Further, chromaticity coordinates are discussed to graphically represent color. In Section 4.2, calculation of the tristimulus values is derived from color matching functions from a RGB system (e.g. color camera). Different color system are then discussed and defined in terms of R , G and B coordinates. In Section 4.5, color order systems are presented computing color differences approximating human perception.

4.1 XYZ System

Having three color matching functions of the standard observer, we are now be able to compute three numbers (called the tristimulus values) equivalent to what a standard observer perceives:

$$X = \int_{\lambda} E(\lambda)S(\lambda)\bar{x}(\lambda)d\lambda \quad (5)$$

$$Y = \int_{\lambda} E(\lambda)S(\lambda)\bar{y}(\lambda)d\lambda \quad (6)$$

$$Z = \int_{\lambda} E(\lambda)S(\lambda)\bar{z}(\lambda)d\lambda \quad (7)$$

where $\bar{x}(\lambda)$, $\bar{y}(\lambda)$ $\bar{z}(\lambda)$ are the CIE color matching functions of the 2° CIE standard observer, see Figure 4.b.

In this way, we are able to compute the color tristimuli values equivalent to a CIE 1931 2° standard observer viewing an object of green paint illuminated by light source D65. Consider the reflectance graph of a green sample. Further, assume that the light source is illuminant D65,

then the X , Y and Z tristimulus values are computed by adding up the product of light, object and matching functions at each wavelength.

The tristimulus values do not have a rather comprehensive meaning for human beings. For example, if we have two colors $X_1 = 34$, $Y_1 = 45$ and $Z_1 = 102$, and $X_2 = 64$, $Y_2 = 90$ and $Z_2 = 202$ respectively, then it can be deduced that the first color is twice that intense the second color. However, it is difficult to describe the chromaticity of the color. Therefore, the notion of the chromaticity of the color is accomplished by defining the so-called chromaticity coordinates:

$$x = \frac{X}{X + Y + Z} \quad (8)$$

$$y = \frac{Y}{X + Y + Z} \quad (9)$$

$$z = \frac{Z}{X + Y + Z} \quad (10)$$

It is obvious that the intensity information is factored out of the system, because the chromaticity coordinates reflect only the ratio of the three standard primary colors.

Since the sum of the chromaticity coordinates equals unity, two of these three quantities are sufficient to describe a color. When the x and y values are represented in a plane, the *C.I.E. 1931 chromaticity diagram* is obtained, see Figure 5. Spectral colors are lying on the tongue-shaped curve according to their wavelengths. In the middle of the diagram, we find neutral white at $x = 0.3333$ and $y = 0.3333$ with corresponding color temperature of 6000 K. For illuminant D65, we obtain $x = 0.3127$ and $y = 0.3290$. Illuminant C has chromaticity coordinates $x = 0.3101$ and $y = 0.3163$, see Figure 5.a. A third dimension can be imagined related to the brightness which is perpendicular to the chromaticity diagram. Instead of using X , Y and Z , a color is more easily understood when it is specified in terms of x , y and Y (i.e. intensity).

Hue and saturation are defined in the chromaticity diagram as follows, see Figure 5.b. First a reference white-light point should be defined. This point is defined as a point in the chromaticity diagram that approximately represents the average daylight, for example illuminant C or D65. For a color $G1$, the hue is defined as the wavelength on the spectral curve that intersects the line from reference white through $G1$ to the spectral curve, which is $G2$ at 523 nm. When $\|G1\|$ is the distance from $P1$ to the white-point, then the saturation is the relative distance to the white-point, given by $\frac{\|G1\|}{\|G2\|}$, where $\|G2\|$ is the distance from $G2$ to the white point. Hence the most saturated colors are the spectral colors.

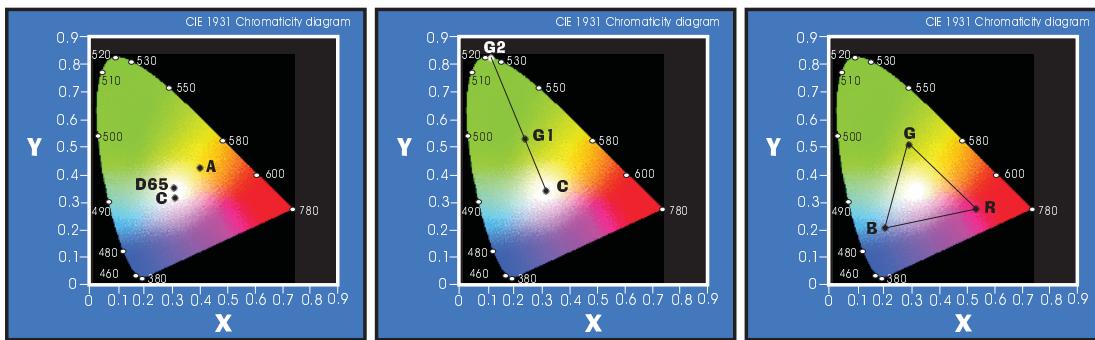


Figure 5: a. Illuminant A, C and D65 plotted in the xy -plane. b. Definition of hue and saturation under illuminants A and D65. c. Color gamuts.

Color gamuts are represented in the chromaticity diagram by lines joining the points defining the color gamut, see Figure 5.c. All colors along the line joining R and G can be obtained by mixing the amounts, corresponding to the distances from R and G , of the colors represented by the endpoints. The color gamut for three points R , G , and B is the triangle formed by the three vertices of the three colors. Any color within the triangle can be produced by a weighted sum of the three colors of the triangle, but no colors represented by points outside the triangle could be produced by these colors. It is now obvious why no set of three primary colors can produce all colors, since no triangle within the diagram can encompass all colors. It also explains why there are many different standards possible for the primary colors, because any triangle in the diagram, defined by its vertices, could be taken as a standard.

In conclusion, we are now able to assign precise numerical values to the color sensation of a standard observer in terms of intuitive attributes hue, saturation and brightness in a total objective manner. In fact, the XYZ system introduced by CIE is the scientific basis of objective color measurement. The XYZ system allows us to compute tristimulus values describing the sensation of a human being, given by matching functions $\bar{x}(\lambda)$, $\bar{y}(\lambda)$ and $\bar{z}(\lambda)$, viewing an object $S(\lambda)$, illuminated by a light source $E(\lambda)$). Although, any set of primary colors can be taken, in the next section, the RGB primaries are given of a color camera. The purpose of the RGB system is to derive mathematical formulae to define color systems in terms of RGB coordinates directly coming from a color camera.

4.2 RGB System

As discussed above, a linear function of the tristimulus values converts a set of color primaries into another set. The standard RGB established by CIE in 1931 with three monochromatic primaries at wavelengths 700 nm (R)ed, 546.1 nm (G)reen, and 435.8 nm (B)lue is the RGB Spectral Primary Color Coordinate System corresponding to the 2° standard observer. The color matching functions have already been shown in Figure 4.b. Another set of primaries have been recommended by the National Television Systems Committee (NTSC) for phosphor standardization. As the NTSC primaries are not pure monochromatic sources of radiation, the color gamut produced by the NTSC primaries is smaller than available from the spectral primaries. Digital camera's and video's mostly use RGB NTSC.

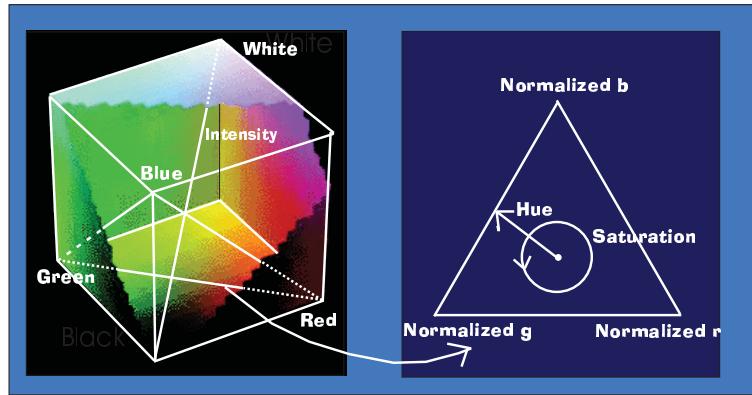


Figure 6: a. RGB -color space b. Definition of hue and saturation in the chromaticity plane.

To represent the RGB -color space, a cube can be defined on the R , G , and B axes, see Figure 6.a. White is produced when all three primary colors are at M , where M is the maximum light

intensity, say $M = 255$. The axis connecting the black and white corners defines the intensity:

$$I(R, G, B) = R + G + B \quad (11)$$

All points in a plane perpendicular to the grey axis of the color cube have the same intensity. The plane through the color cube at points $R = G = B = M$ is one such plane. The projection of RGB points on the rgb chromaticity triangle is defined by:

$$r(R, G, B) = \frac{R}{R + G + B} \quad (12)$$

$$g(R, G, B) = \frac{G}{R + G + B} \quad (13)$$

$$b(R, G, B) = \frac{B}{R + G + B} \quad (14)$$

yielding the rgb color space which is normalized with respect to intensity and graphically represented by Figure 6.a and 6.b where the intensity axis I in RGB -space is projected onto $r = g = b = 1/3$ in the chromaticity plane.

4.3 HSI System

Hue and saturation are defined in the chromaticity triangle in the standard way as follows. Similar to Section 4.1 for the xy diagram, the reference white point is first defined in the rg chromaticity diagram. Let's assume a white light source and hence the reference point is represented by $r = g = b = 1/3$. Then saturation S_{rgb} is defined as the radial distance of a point from the reference white-light point mathematically specified as:

$$S_{rgb}(r, g, b) = \sqrt{(r - 1/3)^2 + (g - 1/3)^2 + (b - 1/3)^2} \quad (15)$$

and graphically represented by Figure 6.b.

H_{rgb} is a function of the angle between a reference line (e.g. horizontal axis) and the color point:

$$H_{rgb}(r, g, b) = \arctan\left(\frac{r - 1/3}{g - 1/3}\right) \quad (16)$$

see Figure 6.c as well.

The transformation from RGB used here to compute H in terms of R , G and B is given by [14]:

$$H(R, G, B) = \arctan\left(\frac{\sqrt{3}(G - B)}{(R - G) + (R - B)}\right) \quad (17)$$

and S measuring the relative white content of a color as having a particular hue by:

$$S(R, G, B) = 1 - \frac{\min(R, G, B)}{R + G + B} \quad (18)$$

Note that rgb , H and S are undefined for achromatic colors (i.e. $R = G = B = 0$).

In conclusion, hue, saturation and intensity are calculated from the original R , G , B values from the corresponding red, green, and blue images provided by the color camera.

4.4 YIQ and YUV System

Various camera's provide images in terms of the *YIQ* NTSC transmission color coordinate system. The National Television Systems Committee (N.T.S.C.) developed the three color attributes *Y*, *I*, and *Q* for transmission efficiency. The tristimulus value *Y* corresponds to the luminance of a color. *I* and *Q* correspond closely the hue and saturation of a color. By reducing the spatial bandwidth of *I* and *Q* without noticeable image degradation, efficient color transmission is obtained. The PAL and SECAM standards used in Europe, the *Y*, *U*, and *V* tristimulus values are used. The *I* and *Q* color attributes are related to *U* and *V* by a simple rotation of the color coordinates in color space. The conversion matrix to compute the *YIQ* values from the original *RGB* NTSC tristimulus values is given by: $Y = 0.299 * R + 0.587 * G + 0.114 * B$, $I = 0.596 * R - 0.274 * G - 0.312 * B$, and $Q = 0.211 * R - 0.523 * G + 0.312 * B$.

4.5 Color Order Systems

A color system is visual uniform when numerical distances can be related to human perceptual differences: the closer a color is to another color in the color space, the more similar they are. It is known that *XYZ* and *RGB* are not visual uniform. To achieve visual uniformity, color order systems have been proposed. A color order system is a multi-dimensional (usually three) space arranging the gamut of colors of visual sensation. This geometric ordering relates a notation to each color corresponding to its position. Each color is given a number yielding an objective classification criterion. Hence, the goal of a color order system is to represent and classify colors objectively, and to provide color differences. Color order systems are often accompanied by atlases and catalogues containing real samples from specific position within the space. Color systems are usually based on perceptual uniformity where the Euclidean distance between two position in the space closely corresponds to the difference between the two colors as perceived by a human being. In this section, the Munsell's color system is discussed in Section 4.5.1. The CIELAB colorimetric space will be given in Section 4.5.2. Color difference will be discussed in Section 4.5.3

4.5.1 Munsell's Color Order System

Albert H. Munsell proposed in 1905, in his "A Color Notation", a geometric ordering space to classify colors. The publication was accompanied with an Atlas containing charts, derived from psychological and experimental measurements, designed to produce a uniform color space. The Munsell color space is based on three perceptual attributes. First, he developed the notation of *Value* ranging from white to black with perceptual equal steps. Further, a *Hue* circle has been designed corresponding to a certain *Chroma* value. The hue circle contains the property that steps between two different hues closely correspond to their perceptual differences. To graphically illustrate the Munsell color space, the color order system can be represented by a cylinder from which the center axis relates to *Value*. Further, *Chroma* corresponds to the distance from this central axis of constant *Hue* forming circles around the *Value* axis. In this way, Munsell provided a visually balanced ordering of colors to yield an objective color identification scheme. The Munsell system has been updated by modern instruments. Today, it is possible to compute the Munsell notations from the CIE *XYZ* tristimulus values.

4.5.2 CIELAB Color Order Systems

As stated above, a color system is visual uniform when numerical distances can be related to human perceptual differences. Hence, the closer a color is to another color in the color space, the more similar they are. MacAdam proved the visual non-uniformity of the *XYZ* color system by doing experiments. These experiments are based on the *just noticeable difference* (JND) concept.

The JND is defined as the minimal visual difference of two colors: let Q_0 be a color then color Q_1 is just noticeable different from Q_0 , if there is no color Q_2 , lying on the line from Q_0 through Q_1 , which is closer to and noticeable different from Q_0 . The results of the experiments made by MacAdam is that colors with coordinates (x_i, y_i) in the xy chromaticity diagram, which are just noticeable different from color center (x_0, y_0) , are lying on an ellipsis with center (x_0, y_0) and all colors which are not (just) noticeable different from (x_0, y_0) are lying outside the ellipsis. The ellipses of an arbitrary number of colors were calculated and analyzed. MacAdam proved that the differences between the axis-diameters of the ellipses were rather (too) large and therefore non-uniformity of the XYZ color system can easily be seen.

Over the last decades various attempts have been made to develop perceptual uniform spaces and color difference equations. To this end, in 1976, the CIE recommended a new system $L^*a^*b^*$, computed from the XYZ color system, having the property that the closer a point (representing a color) is to another point (also representing a color -possibly the same-), the more visual similar the colors (represented by the points) are. In other words, the magnitude of the perceived color difference of two colors corresponds to the Euclidean distance between the two colors in the color system.

The $L^*a^*b^*$ system is based on the three dimensional coordinate system based on the opponent theory using black-white L^* , red-green a^* , and yellow-blue b^* components. The L^* axis corresponds to the lightness where $L^* = 100$ is white and $L^* = 0$ is black. Further, a^* ranges from red $+a^*$ to green $-a^*$ while b^* ranges from yellow $+b^*$ to blue $-b^*$.

The L^* , a^* and b^* coordinates are computed from the X , Y and Z tristimulus values as follows:

$$L^* = 116f\left(\frac{Y}{Y_n}\right)16 \quad (19)$$

$$a^* = 500[f\left(\frac{X}{X_n} - \frac{Y}{Y_n}\right)] \quad (20)$$

$$b^* = 500[f\left(\frac{Y}{Y_n} - \frac{Z}{Z_n}\right)] \quad (21)$$

where X_n , Y_n and Z_n are the tristimulus values of the nominal white stimulus. For example, for D65/10°: $X_n = 94.81$ m $Y_n = 100.00$ and $Z_n = 107.304$. Further, $f\left(\frac{X}{X_n}\right) = \left(\frac{X}{X_n}\right)^{1/3}$ for values $\frac{X}{X_n}$ greater than 0.008856 and $f\left(\frac{X}{X_n}\right) = 7.787\left(\frac{X}{X_n}\right) + 16/116$ for values $\frac{X}{X_n}$ equal or less than 0.008856. Equal arguments hold for Y and Z .

The intuitive visual attributes hue and chroma (saturation) can be expressed by:

$$C^* = (a^{*2} + b^{*2})^{1/2} \quad (22)$$

$$h = \arctan\left(\frac{b^*}{a^*}\right) \quad (23)$$

Note that the CIE $L^*a^*b^*$ chroma is not the same as the Munsell Chroma. Further, the CIE $L^*a^*b^*$ is still not perfectly visual uniform. However, the system allows us to define a color difference by measuring the geometric distance between two colors in the color space. The color difference ruler is discussed in the following section.

4.5.3 Color Difference

Many image retrieval applications must provide the ability to retrieve visual similar images satisfying the expectation of the operator. Therefore, images should be retrieved within the limits of acceptable variations of color. For example, consider a database of images taken from paper samples. The operator searches for samples with a specific color. Then, color differences

are essential for the retrieval task at hand. Note the difference between perceptibility and acceptability. The color difference is perceptible if the difference can be seen by a human operator. Even when the difference is perceptible for different tasks, the color difference can still be acceptable. As discussed above, the $L^*a^*b^*$ is approximately visual uniform and provides us the ability to define a perceptual color difference in the Euclidean way.

First, the differences are subtracted from the L^* , a^* , and b^* components of the trial from the standard:

$$\Delta L^* = L^*_{\text{trial}} - L^*_{\text{standard}} \quad (24)$$

$$\Delta a^* = a^*_{\text{trial}} - a^*_{\text{standard}} \quad (25)$$

$$\Delta b^* = b^*_{\text{trial}} - b^*_{\text{standard}} \quad (26)$$

where a positive value of ΔL^* implies that the sample is lighter than the standard one. Further, a positive value of Δa^* and Δb^* indicates that the sample is redder and yellow than the standard.

The total CIE $L^*a^*b^*$ is given by:

$$\Delta E_{ab}^* = \sqrt{(\Delta L^*)^2 + (\Delta a^*)^2 + (\Delta b^*)^2} \quad (27)$$

The difference in saturation is given by:

$$\Delta C^* = C^*_{\text{trial}} - C^*_{\text{standard}} \quad (28)$$

Due to the circular nature of hue, the hue difference is given by:

$$\Delta H_{ab}^* = \sqrt{(\Delta E_{ab}^*)^2 + (\Delta C^*)^2 + (\Delta L^*)^2} \quad (29)$$

4.5.4 Summary

Color order systems are required to identify colors into a geometric structure. The Munsell space was one of the first color order system with the aim to represent color objectively. Due to the perceptual nonuniformity of the CIE XYZ system, the *CIE* recommended the $L^*a^*b^*$ color order system where the Euclidean distance between two colors closely corresponds to the visual color difference.

5 Color Invariance

In the previous section, the scientific measurement of color has been outlined to measure objectively the light-object-observer process yielding basic notions on color, color models and ordering systems. The advantage of using, for example, the $L^*a^*b^*$ system, is that it corresponds with human perception which is useful when retrieving images which are visually similar. However, it is known that the $L^*a^*b^*$ and *RGB* color system are dependent on the imaging conditions. Therefore, for general image retrieval, an important color property is color invariance. A color invariant system contains color invariant models which are more or less insensitive to the varying imaging conditions such as variations in illumination (e.g. lamps having different spectral power distribution) and object pose (changing shadows, shading and highlighting cues). Color invariants for the purpose of viewpoint invariant image retrieval is given in [10]. In this section, an overview is given of color invariants.

This section is outlined as follows. First, in Section 5.1, a general reflection model is discussed to model the interaction between light and material. Then, in Section 5.2, assuming white illumination, basic color systems are analyzed with respect to their invariance. In Section 5.3,

color constant color systems are given. Color constant color systems are independent of the relative spectral power distribution of the light source.

5.1 Reflection from Inhomogeneous Dielectric Materials

To arrive at a uniform description it is common to divide opaque materials into two classes on the basis of their optical properties: optically homogeneous and inhomogeneous materials.

Optically homogeneous materials have a constant index of refraction throughout the material. This means that when light is incident upon the surface of an homogeneous materials, some fraction of it is reflected. A smooth surface reflects light only in the direction such that the angle of incidence equals the angle of reflection. The properties of this reflected light are determined by the optical and geometric properties of the surface. Metals are homogeneous materials having a larger specular component than other materials.

Optically inhomogeneous materials are composed of a vehicle with many embedded colorant particles that differ optically from the vehicle. The fraction of the incident light that is not reflected from the surface enters the body of the material. The body is composed of a vehicle and many colorant particles. When light encounters a colorant particle, some portion of it is reflected. After many reflections, the light is diffused, and a significant fraction can exit back through the surface in a wide range of directions. Some examples of inhomogeneous materials are plastics, paper, textiles, and paints.

The Dichromatic Reflection Model describes the light, which is reflected from a point on a dielectric, inhomogeneous material as a mixture of the light reflected at the material surface and the light reflected from the material body.

Let $E(\vec{x}, \lambda)$ be the spectral power distribution of the incident (ambient) light at the object surface at \vec{x} , and let $L(\vec{x}, \lambda)$ be the spectral reflectance function of the object at \vec{x} . The spectral sensitivity of the k th sensor is given by $F_k(\lambda)$. Then ρ_k , the sensor response of the k th channel, is given by:

$$\rho_k(\vec{x}) = \int_{\lambda} E(\vec{x}, \lambda) L(\vec{x}, \lambda) F_k(\lambda) d\lambda \quad (30)$$

where λ denotes the wavelength, and $L(\vec{x}, \lambda)$ is a complex function based on the geometric and spectral properties of the object surface. The integral is taken from the visible spectrum (e.g. 380-700 nm).

Further, consider an opaque inhomogeneous dielectric object, then the geometric and surface reflection component of function $L(\vec{x}, \lambda)$ can be decomposed in a body and surface reflection component as described by Shafer [24]:

$$\begin{aligned} \phi_k(\vec{x}) &= G_B(\vec{x}, \vec{n}, \vec{s}) \int_{\lambda} E(\vec{x}, \lambda) B(\vec{x}, \lambda) F_k(\lambda) d\lambda + \\ &G_S(\vec{x}, \vec{n}, \vec{s}, \vec{v}) \int_{\lambda} E(\vec{x}, \lambda) S(\vec{x}, \lambda) F_k(\lambda) d\lambda \end{aligned} \quad (31)$$

giving the k th sensor response. Further, $B(\vec{x}, \lambda)$ and $S(\vec{x}, \lambda)$ are the surface albedo and Fresnel reflectance at \vec{x} respectively. \vec{n} is the surface patch normal, \vec{s} is the direction of the illumination source, and \vec{v} is the direction of the viewer. Geometric terms G_B and G_S denote the geometric dependencies on the body and surface reflection component respectively.

5.2 Reflectance with White Illumination

Considering the neutral interface reflection (NIR) model (assuming that $S(\vec{x}, \lambda)$ has a nearly constant value independent of the wavelength) and approximately white illumination, then $S(\vec{x}, \lambda) = S(\vec{x})$, and $E(\vec{x}, \lambda) = E(\vec{x})$. Then, the measured sensor values are given by [9]:

$$\begin{aligned}\omega_k(\vec{x}) &= G_B(\vec{x}, \vec{n}, \vec{s}) E(\vec{x}) \int_{\lambda} B(\vec{x}, \lambda) F_k(\lambda) d\lambda + \\ &G_S(\vec{x}, \vec{n}, \vec{s}, \vec{v}) E(\vec{x}) S(\vec{x}) \int_{\lambda} F_k(\lambda) d\lambda\end{aligned}\quad (32)$$

giving the k th sensor response of an infinitesimal surface patch under the assumption of a white light source.

If the integrated white condition holds (i.e. the area under the sensor spectral functions is approximately the same):

$$\int_{\lambda} F_i(\lambda) d\lambda = \int_{\lambda} F_j(\lambda) d\lambda \quad (33)$$

the reflection from inhomogeneous dielectric materials under white illumination is given by [9]:

$$\begin{aligned}\omega_k(\vec{x}) &= G_B(\vec{x}, \vec{n}, \vec{s}) E(\vec{x}) \int_{\lambda} B(\vec{x}, \lambda) F_k(\lambda) d\lambda + \\ &G_S(\vec{x}, \vec{n}, \vec{s}, \vec{v}) E(\vec{x}) S(\vec{x}) F\end{aligned}\quad (34)$$

If $\omega(\vec{x})$ is not dependent on \vec{x} , we obtain:

$$\omega_k = G_B(\vec{n}, \vec{s}) E \int_{\lambda} B(\lambda) F_k(\lambda) d\lambda + G_S(\vec{n}, \vec{s}, \vec{v}) E S F \quad (35)$$

5.2.1 Invariance for Matte Surfaces

Consider the body reflection term of equation (34):

$$\beta_k(\vec{x}) = G_B(\vec{x}, \vec{n}, \vec{s}) E(\vec{x}) \int_{\lambda} B(\vec{x}, \lambda) F_k(\lambda) d\lambda \quad (36)$$

giving the k th sensor response of an infinitesimal *matte* surface patch under the assumption of a white light source.

The body reflection component describes the way light interacts with a dull surface. The light spectrum E falls on a surface B . The geometric and photometric properties of the body reflection depends on many factors. If we assume a random distribution of the pigments, the light exits in random directions from the body. In this case, the distribution of exiting light can be described by Lambert's law. Lambertian reflection models dull, matte surfaces which appear equally bright regardless from angle they are viewed. They reflect light with equal intensity in all directions.

As a consequence, a uniformly colored surface which is curved (i.e. varying surface orientation) gives rise to a broad variance of *RGB* values. The same argument holds for intensity I .

In contrast, rgb is insensitive to surface orientation, illumination direction and illumination intensity mathematically specified by substituting equation (36) in equation (12) - (14):

$$r(R_b, G_b, B_b) = \frac{G_B(\vec{x}, \vec{n}, \vec{s}) E(\vec{x}) k_R}{G_B(\vec{x}, \vec{n}, \vec{s}) E(\vec{x})(k_R + k_G + k_B)} = \frac{k_R}{k_R + k_G + k_B} \quad (37)$$

$$g(R_b, G_b, B_b) = \frac{G_B(\vec{x}, \vec{n}, \vec{s}) E(\vec{x}) k_G}{G_B(\vec{x}, \vec{n}, \vec{s}) E(\vec{x})(k_R + k_G + k_B)} = \frac{k_G}{k_R + k_G + k_B} \quad (38)$$

$$r(R_b, G_b, B_b) = \frac{G_B(\vec{x}, \vec{n}, \vec{s}) E(\vec{x}) k_B}{G_B(\vec{x}, \vec{n}, \vec{s}) E(\vec{x})(k_R + k_G + k_B)} = \frac{k_B}{k_R + k_G + k_B} \quad (39)$$

factoring out dependencies on illumination and object geometry and hence only dependent on the sensors and the surface albedo.

Because S corresponds to the radial distance from the color to the main diagonal in the RGB -color space, S is an invariant for matte, dull surfaces illuminated by white light cf. equation (36) and equation (18):

$$\begin{aligned} S(R_b, G_b, B_b) &= 1 - \frac{\min(G_B(\vec{x}, \vec{n}, \vec{s})E(\vec{x})k_R, G_B(\vec{x}, \vec{n}, \vec{s})E(\vec{x})k_G, G_B(\vec{x}, \vec{n}, \vec{s})E(\vec{x})k_B)}{G_B(\vec{x}, \vec{n}, \vec{s})E(\vec{x})(k_R + k_G + k_B)} = \\ &1 - \frac{\min(k_R, k_G, k_B)}{(k_R + k_G + k_B)} \end{aligned} \quad (40)$$

only dependent on the sensors and the surface albedo.

Similarly, H is an invariant for matte, dull surfaces illuminated by white light cf. equation (36) and equation (17):

$$\begin{aligned} H(R_b, G_b, B_b) &= \arctan\left(\frac{\sqrt{3}G_B(\vec{x}, \vec{n}, \vec{s})E(\vec{x})(k_G - k_B)}{G_B(\vec{x}, \vec{n}, \vec{s})E(\vec{x})((k_R - k_G) + (k_R - k_B))}\right) = \\ &\arctan\left(\frac{\sqrt{3}(k_G - k_B)}{(k_R - k_G) + (k_R - k_B)}\right) \end{aligned} \quad (41)$$

Obviously, in practice, the assumption of objects composed of matte, dull surfaces is not always realistic. To that end, the effect of surface reflection (highlights) is discussed in the following section.

5.2.2 Invariance for Shiny Surfaces

Consider the surface reflection term of equation (34):

$$C_s = G_S(\vec{x}, \vec{n}, \vec{s}, \vec{v})E(\vec{x})S(\vec{x})F \quad (42)$$

for $C_s \in \{R_s, G_s, B_s\}$ giving the red, green and blue sensor response for a highlighted infinitesimal surface patch with white illumination.

When light hits the surface of a dielectric inhomogeneous material, it must first pass through the interface between the surrounding medium (e.g. air) and the material. Since the refraction index of the material is generally different from that of the surrounding medium, some percentage of the incident light is reflected at the surface of the material.

Several models have been developed in the physics and computer graphics communities to describe the geometric properties of the light reflected from rough surfaces. Issues in modeling this process involve the roughness scale of the surface, compared to the wavelengths of the incident light, and self-shadowing effects on the surface which depend on the viewing direction and the direction of the incident light. The common used surface reflection model is described by a function with a sharp peak around the angle of perfect mirror reflection. For smooth surfaces, surface (specular) reflections are in the direction of \vec{r} , which is \vec{s} mirrored about \vec{n} . For non-perfect rough reflectors, the maximum surface reflection occurs when α , the angle between \vec{r} and \vec{e} is zero; and falls off rapidly as α increases. The falloff is approximated by $\cos^n \alpha$, where n is the surface's surface-reflection exponent or degree of shininess.

Because H is a function of the angle between the main diagonal and the color point in RGB -sensor space, all possible colors of the same (shiny) surface region (i.e. with fixed albedo) have to be of the same hue as follows from substituting equation (42) in equation (17):

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

$$\begin{aligned} \arctan\left(\frac{\sqrt{3}G_B(\vec{x}, \vec{n}, \vec{s})E(\vec{x})(k_G - k_B)}{G_B(\vec{x}, \vec{n}, \vec{s})E(\vec{x})((k_R - k_G) + (k_R - k_B))}\right) = \\ \arctan\left(\frac{\sqrt{3}(k_G - k_B)}{(k_R - k_G) + (k_R - k_B)}\right) \end{aligned} \quad (43)$$

factoring out dependencies on illumination, object geometry, viewpoint and specular reflection coefficient and hence only dependent on the sensors and the surface albedo. Note that $C_w = G_B(\vec{x}, \vec{n}, \vec{s})E(\vec{x}) \int_{\lambda} B(\vec{x}, \lambda)F_C(\lambda)d\lambda + G_S(\vec{x}, \vec{n}, \vec{s}, \vec{v})E(\vec{x})S(\vec{x})F$ for $C = \{R, G, B\}$.

Obviously other color features depend on the contribution of the surface reflection component and hence are sensitive to highlights.

5.3 Color Constancy

Existing color constancy methods require specific a priori information about the observed scene (e.g. the placement of calibration patches of known spectral reflectance in the scene) which will not be feasible in practical situations, [6], [7], [13] for example. To circumvent these problems, simple and effective illumination-independent color ratio's have been proposed by Funt and Finlayson [8] and Nayar and Bolle [17]. In fact, these color models are based on the ratio of surface albedos rather than the recovering of the actual surface albedo itself. However, these color models assume that the variation in spectral power distribution of the illumination can be modeled by the coefficient rule or von Kries model, where the change in the illumination color is approximated by a 3x3 diagonal matrix among the sensor bands and is equal to the multiplication of each *RGB*-color band by an independent scalar factor. The diagonal model of illumination change holds exactly in the case of narrow-band sensors. Consider the body reflection term of the dichromatic reflection model:

$$C_c = G_B(\vec{x}, \vec{n}, \vec{s})E(\vec{x}) \int_{\lambda} B(\vec{x}, \lambda)F_k(\lambda)d\lambda \quad (44)$$

for $C = \{R, G, B\}$, where $C_c = \{R_c, G_c, B_c\}$ gives the red, green and blue sensor response of a matte infinitesimal surface patch of an inhomogeneous dielectric object under unknown spectral power distribution of the illumination.

Suppose that the sensor sensitivities of the color camera are narrow-band with spectral response be approximated by delta functions $f_K(\lambda) = \delta(\lambda - \lambda_K)$, then the measured sensor values are:

$$C_K = G_B(\vec{x}, \vec{n}, \vec{s})E(\vec{x}, \lambda_K)B(\vec{x}, \lambda_K) \quad (45)$$

The color ratio's proposed by Nayar and Bolle is given by [17]:

$$N(C^{\vec{x}_1}, C^{\vec{x}_2}) = \frac{C^{\vec{x}_1} - C^{\vec{x}_2}}{C^{\vec{x}_2} + C^{\vec{x}_1}} \quad (46)$$

and by Funt and Finlayson by [8]:

$$F(C^{\vec{x}_1}, C^{\vec{x}_2}) = \frac{C^{\vec{x}_1}}{C^{\vec{x}_2}} \quad (47)$$

expressing color ratio's between two neighboring image locations, for $C \in \{R, G, B\}$, where \vec{x}_1 and \vec{x}_2 denote the image locations of the two neighboring pixels. Note that the set $\{R, G, B\}$ must be colors from narrow-band sensor filters and that they are used in defining the color ratio because they are immediately available from a color camera, but any other set of narrow-band colors derived from the visible spectrum will do as well. Although standard video camera's are

not equipped with narrow-band filters, spectral sharpening could be applied [4] to achieve this to a large extend.

Assuming that the color of the illumination is locally constant (i.e. $E(\vec{x}_1, \lambda_K) = E(\vec{x}_2, \lambda_K)$) and that neighboring points have the same surface orientation (i.e. $G_B(\vec{x}_1, \vec{n}, \vec{s}) = G_B(\vec{x}_2, \vec{n}, \vec{s})$), then the color ratio N is independent of the illumination intensity and color as shown by substituting equation (46) in equation (44):

$$\frac{G_B(\vec{x}_1, \vec{n}, \vec{s})E(\vec{x}_1, \lambda_K)B(\vec{x}_1, \lambda_K) - G_B(\vec{x}_2, \vec{n}, \vec{s})E(\vec{x}_2, \lambda_K)B(\vec{x}_2, \lambda_K)}{G_B(\vec{x}_1, \vec{n}, \vec{s})E(\vec{x}_1, \lambda_K)B(\vec{x}_1, \lambda_K) + G_B(\vec{x}_2, \vec{n}, \vec{s})E(\vec{x}_2, \lambda_K)B(\vec{x}_2, \lambda_K)} = \frac{B(\vec{x}_1, \lambda_K) - B(\vec{x}_2, \lambda_K)}{B(\vec{x}_1, \lambda_K) + B(\vec{x}_2, \lambda_K)} \quad (48)$$

Equal arguments hold for the color ratio F by substituting equation (47) in equation (44):

$$\frac{G_B(\vec{x}_1, \vec{n}, \vec{s})E(\vec{x}_1, \lambda_K)B(\vec{x}_1, \lambda_K)}{G_B(\vec{x}_2, \vec{n}, \vec{s})E(\vec{x}_2, \lambda_K)B(\vec{x}_2, \lambda_K)} = \frac{B(\vec{x}_1, \lambda_K)}{B(\vec{x}_2, \lambda_K)} \quad (49)$$

However, it is assumed that the neighboring points, from which the color ratio's are computed, have the same surface normal. Therefore, the method depends on varying surface orientation of the object (i.e. the geometry of the objects) affecting negatively the recognition performance. To this end, a color constant color ratio has been proposed not only independent of the illumination color but also discounting the object's geometry [9]:

$$m(C_1^{\vec{x}_1}, C_1^{\vec{x}_2}, C_2^{\vec{x}_1}, C_2^{\vec{x}_2}) = \frac{C_1^{\vec{x}_1}C_2^{\vec{x}_2}}{C_1^{\vec{x}_2}C_2^{\vec{x}_1}}, C_1 \neq C_2 \quad (50)$$

expressing the color ratio between two neighboring image locations, for $C_1, C_2 \in \{R, G, B\}$ where \vec{x}_1 and \vec{x}_2 denote the image locations of the two neighboring pixels.

The color ratio is independent of the illumination intensity and color, and also to a change in viewpoint, object geometry, and illumination direction as shown by substituting equation (50) in equation (44):

$$\frac{(G_B(\vec{x}_1, \vec{n}, \vec{s})E(\vec{x}_1, \lambda_K)B(\vec{x}_1, \lambda_{C_1}))(G_B(\vec{x}_2, \vec{n}, \vec{s})E(\vec{x}_2, \lambda_K)B(\vec{x}_2, \lambda_{C_2}))}{(G_B(\vec{x}_2, \vec{n}, \vec{s})E(\vec{x}_2, \lambda_K)B(\vec{x}_2, \lambda_{C_1}))(G_B(\vec{x}_1, \vec{n}, \vec{s})E(\vec{x}_1, \lambda_K)B(\vec{x}_1, \lambda_{C_2}))} = \frac{B(\vec{x}_1, \lambda_{C_1})B(\vec{x}_2, \lambda_{C_2})}{B(\vec{x}_2, \lambda_{C_1})B(\vec{x}_1, \lambda_{C_2})} \quad (51)$$

factoring out dependencies on object geometry and illumination direction $G_B(\vec{x}_1, \vec{n}, \vec{s})$ and $G_B(\vec{x}_2, \vec{n}, \vec{s})$, and illumination for $E(\vec{x}_1, \lambda_{C_2}) = E(\vec{x}_2, \lambda_{C_2})$, and hence only dependent on the ratio of surface albedos, where \vec{x}_1 and \vec{x}_2 are two neighboring locations on the object's surface not necessarily of the same orientation.

Note that the color ratio's do not require any specific a priori information about the observed scene, as the color model is an illumination-invariant surface descriptor based on the ratio of surface albedos rather than the recovering of the actual surface albedo itself. Also the intensity and spectral power distribution of the illumination is allowed to vary across the scene (e.g. multiple light sources with different SPD's), and a certain amount of object occlusion and cluttering is tolerated due to the local computation of the color ratio.

6 Color System Taxonomy

The purpose of this section is to give a compact formulation on the relevance of the color systems for the purpose of color based image retrieval. Therefore, a survey is given on the color models. Each color system is briefly discussed and transformations are given in terms of *RGB* NTSC tristimulus color coordinate system. Further, the following criteria are used to classify the color systems: 1. Is the color system device independent; 2. Is the color system perceptual uniform; 3. Is the color system non-linear; 4. Is the color system intuitive; 5. Is the color system robust against varying imaging conditions. The section will end up with a table summarizing the main and important characteristics of the color systems.

6.1 Grey-Value System

The color model *GREY* or *INTENSITY* is calculated from the original *R*, *G*, *B* NTSC tristimulus values from the corresponding red, green, and blue images provided by a ccd color camera and hence dependent on the imaging device. Grey is not perceptual uniform as a just noticeable brighter grey-value does not correspond with a difference between two successive grey-values. Grey is heavily influenced by the imaging conditions.

- COLOR FEATURE: *GREY*;
- TRANSFORMATION:

$$GREY = 0.299R + 0.587G + 0.144B; \quad (52)$$

- CHARACTERISTICS:

- Device dependent
- Not perceptual uniform
- Linear
- Intuitive
- Dependent on viewing direction, object geometry, direction of the illumination, intensity and color of the illumination

- REMARKS: Grey value information.

6.2 RGB Color System

The *RGB* color system represents the color (R)ed, (G)reen and (B)lue color. In this chapter, the *R*, *B*, and *G* color features correspond to the primary colors where *R* = 700 nm, *G* = 546.1 nm, and *B* = 435.8 nm. Similar to grey-value, the *RGB* color system is not perceptual uniform and dependent on the imaging conditions. Using *RGB* values for image retrieval cause problems when the query and target image are recorded under different imaging conditions.

- COLOR MODELS: *R*, *G*, and *B*;
- TRANSFORMATION: no transformation;
- CHARACTERISTICS:
 - Device dependent
 - Not perceptual uniform
 - None
 - Not intuitive
 - Dependent on viewing direction, object geometry, direction of the illumination, intensity and color of the illumination
- REMARKS: No transformation required.

6.3 *rgb* Color System

The *rgb* color system has three color features: r , g , and b . These color models are called normalized colors, because each of them is calculated by dividing the values of respectively R , G , and B by their total sum. Because the r , g , and b coordinates depend only on the ratio of the R , G , and B coordinates (i.e. factoring luminance out of the system), they have the important property that they are not sensitive to surface orientation, illumination direction and illumination intensity cf. equations (37-39). An other important property is that it is convenient to represent these features in the chromaticity diagram. Normalized colors become unstable and meaningless when the intensity is small [12].

- COLOR MODELS: r , g , and b ;
- TRANSFORMATION:

$$r = \frac{R}{(R + G + B)} \quad (53)$$

$$g = \frac{G}{(R + G + B)} \quad (54)$$

$$b = \frac{B}{(R + G + B)} \quad (55)$$

- CHARACTERISTICS:
 - Device dependent
 - Not perceptual uniform
 - Nonlinear: become unstable when intensity is small
 - Not intuitive
 - Dependent on highlights and a change in the color of the illumination.
- REMARKS: conveniently represented in the chromaticity diagram.

6.4 *XYZ* Color System

This color system is based on the additive mixture of three imaginary primaries X , Y , and Z introduced by CIE. These primaries can not be seen by a human eye or produced, because they are too saturated. The fact that these primaries are imaginary colors is not important, since any perceived color can be described mathematically by the amounts of these primaries. An other important property is that the luminance is determined only by the Y value. Because *XYZ* system is a linear combination of R , G and B values, the *XYZ* color system inherits all the dependencies on the imaging conditions from the *RGB* color system. Note that the color system is device independent as the X , Y and Z values are objective in their interpretation. Further, the conversion matrix given below is based on the *RGB* NTSC color coordinates system.

- COLOR MODELS: X , Y , Z ;
- TRANSFORMATION:

$$X = 0.607R + 0.174G + 0.200B \quad (56)$$

$$Y = 0.299R + 0.587G + 0.114B \quad (57)$$

$$Z = 0.000R + 0.066G + 1.116 \quad (58)$$

- CHARACTERISTICS:
 - Device independent
 - Not perceptual uniform
 - Linear transformation
 - Not intuitive
 - Dependent on viewing direction, object geometry, highlights, direction of the illumination, intensity and color of the illumination
- REMARKS: all colors are described mathematically by three color features, luminance is based on the Y value alone.

6.5 xyz Color System

The color features of this system are the chromaticity coordinates: x , y , and z . Two of those chromaticity coordinates are enough to provide a complete specification and can be represented in the chromaticity diagram. Similar to rgb , this system cancels intensity out yielding independence of surface orientation, illumination direction and illumination intensity. Again problems arise when intensity is low.

- COLOR MODELS: x , y , z ;
- TRANSFORMATION:

$$x = \frac{X}{(X + Y + Z)} \quad (59)$$

$$y = \frac{Y}{(X + Y + Z)} \quad (60)$$

$$z = \frac{Z}{(X + Y + Z)} \quad (61)$$

- FEATURE PROBLEMS:
 - Device independent
 - Not perceptual uniform
 - Nonlinear transformation: become unstable when intensity is small
 - Not intuitive
 - Dependent highlights and a change in the color of the illumination
- REMARKS: conveniently represented in the chromaticity diagram.

6.6 $U^*V^*W^*$ Color System

C.I.E. introduced the $U^*V^*W^*$ color system which has three color features: U^* , V^* , and W^* . The color model W^* is based on the scaling of luminance. The luminance of a color is determined only by its Y value. Scaling luminance between 0 (black) and 100(white), the scaling method starts with black and selects a *just noticeable brighter* grey-value. Taking this just noticeable brighter grey-value the next just noticeable brighter grey-value is selected. This process continues until white is reached. This process is a method to scale brightness visually and the curve obtained by the scaling method can be approximated by the following formula:

$$W^* = \begin{cases} 116\left(\frac{Y}{Y_0}\right)^{\frac{1}{3}} - 16 & , \text{ if } \frac{Y}{Y_0} > 0.008856 \\ 903.3\left(\frac{Y}{Y_0}\right) & , \text{ if } \frac{Y}{Y_0} \leq 0.008856 \end{cases} \quad (62)$$

and Y_0 is a nominally white object-color stimulus.

The other two color features solve the problem of large difference of the axis-diameters of the ellipses in the chromaticity diagram, where colors which are not noticeable different for a particular color are lying on the ellipses and all colors which are (just) noticeable different are lying outside the ellipses. The system is visual uniform, because a luminance difference corresponds with the same noticed luminance difference and the ellipses in the adjusted chromaticity diagram have constant axis-diameters. The U^* and V^* color models become unstable and meaningless when intensity is small.

- COLOR MODELS: U^* , V^* , W^* ,
- TRANSFORMATION:

$$U^* = 13W^*(u - u_0), \quad (63)$$

$$V^* = 13W^*(v - v_0), \quad (64)$$

$$W^* = \begin{cases} 116\left(\frac{Y}{Y_0}\right)^{\frac{1}{3}} - 16 & , \text{ if } \frac{Y}{Y_0} > 0.008856 \\ 903.3\left(\frac{Y}{Y_0}\right) & , \text{ if } \frac{Y}{Y_0} \leq 0.008856 \end{cases} \quad (65)$$

$$u = \frac{4X}{(X + 15Y + 3Z)}, \quad (66)$$

$$v = \frac{6Y}{(X + 15Y + 3Z)}, \quad (67)$$

$$u_0 = \frac{4X_0}{(X_0 + 15Y_0 + 3Z_0)}, \quad (68)$$

$$v_0 = \frac{6Y_0}{(X_0 + 15Y_0 + 3Z_0)}, \quad (69)$$

- CHARACTERISTICS:
 - Device independent
 - Perceptual uniform
 - Nonlinear transformation: become unstable when intensity is small
 - Not intuitive
 - Dependent on viewing direction, object geometry, highlights, direction of the illumination, intensity and color of the illumination
- REMARKS: Visual uniform.

X_0 , Y_0 , Z_0 are values of a nominally white object-color stimulus.

6.7 $L^*a^*b^*$ Color System

An other kind of visual uniform color system proposed by C.I.E. is the $L^*a^*b^*$ color system. The color feature L^* correlates with the perceived luminance and corresponds to W^* of the $U^*V^*W^*$ color system. Color feature a^* correlates with the red-green content of a color and b^* reflects the yellow-blue content.

- COLOR MODELS: L^*, a^*, b^* ,
- TRANSFORMATION:

$$L^* = \begin{cases} 116\left(\frac{Y}{Y_0}\right)^{\frac{1}{3}} - 16 & , \text{ if } \frac{Y}{Y_0} > 0.008856 \\ 903.3\left(\frac{Y}{Y_0}\right) & , \text{ if } \frac{Y}{Y_0} \leq 0.008856 \end{cases} \quad (70)$$

$$a^* = 500\left[\left(\frac{X}{X_0}\right)^{\frac{1}{3}} - \left(\frac{Y}{Y_0}\right)^{\frac{1}{3}}\right], \quad (71)$$

$$b^* = 200\left[\left(\frac{Y}{Y_0}\right)^{\frac{1}{3}} - \left(\frac{Z}{Z_0}\right)^{\frac{1}{3}}\right]; \quad (72)$$

- CHARACTERISTICS:

- Device independent
- Perceptual uniform
- Nonlinear transformation: become unstable when intensity is small
- Not intuitive
- Dependent on viewing direction, object geometry, highlights, direction of the illumination, intensity and color of the illumination

6.8 $I1I2I3$ Color System

When RGB images are highly correlated, it might be desirable to down-weight this correlation. This can be achieved by computing the Karhunen-Loëve transformation. This transformation is calculated from the covariance matrix and calculates an uncorrelated basis.

Three color models I_1 , I_2 and I_3 have been presented by doing experiments and by analyzing the results of eight color scenes by [19]. The color scenes were digitalized with 256*256 spatial resolution and 6-bit intensity resolution for each R , G , and B . Ohta et al. calculated the eigenvectors of for each of the eight color scenes. From the analysis, three orthogonal color models were derived: $I_1 = (R + G + B)$, $I_2 = (R - G)/2$ and $I_3 = (2G - R - B)/4$. Note that I_1 corresponds to intensity.

- COLOR MODELS:
 I_1, I_2, I_3 ;
- TRANSFORMATION:

$$I_1 = \frac{(R + G + B)}{3}, \quad (73)$$

$$I2 = \frac{(R - B)}{2}, \quad (74)$$

$$I3 = \frac{(2G - R - B)}{2}, \quad (75)$$

- CHARACTERISTICS:

- Device dependent
- Not perceptual uniform
- Linear
- Intuitive
- Dependent on viewing direction, object geometry, highlights, direction of the illumination, intensity and color of the illumination

- REMARKS: Uncorrelation based on the Karhunen-Loëve transformation of eight different color images.

6.9 YIQ and YUV

The National Television Systems Committee (N.T.S.C.) developed the three color attributes Y , I , and Q for transmission efficiency. The tristimulus value Y corresponds to the luminance of a color. I and Q correspond closely the hue and saturation of a color. By reducing the spatial bandwidth of I and Q without noticeable image degradation, efficient color transmission is obtained. The PAL and SECAM standards used in Europe, the Y , U , and V tristimulus values are used. The I and Q color attributes are related to U and V by a simple rotation of the color coordinates in color space.

- COLOR MODELS: Y , I , Q ;
- TRANSFORMATION:

$$Y = 0.299R + 0.587G + 0.114B \quad (76)$$

$$I = 0.596R - 0.274G - 0.312B \quad (77)$$

$$Q = 0.211R - 0.523G + 0.312B \quad (78)$$

- CHARACTERISTICS:

- Device independent
- Not perceptual uniform
- Linear transformation
- Not intuitive
- Dependent on viewing direction, object geometry, highlights, direction of the illumination, intensity and color of the illumination

- REMARKS: Y is the luminance of a color.

6.10 HSI Color System

The human color perception is conveniently represented by the following set of color features: I (ntensity), S (aturation), and H (ue). I is an attribute in terms of which a light or surface color may be ordered on a scale from dim to bright. S denotes the relative white content of a color and H is the color aspect of a visual impression.

The problem of hue is that it becomes unstable when S is near zero due to the non-removable singularities in the nonlinear transformation, which a small perturbation of the input can cause a large jump in the transformed values.

- COLOR MODELS: I, H, S ;
- TRANSFORMATION:

$$I = \frac{(R + G + B)}{3}, \quad (79)$$

$$H(R, G, B) = \arctan\left(\frac{\sqrt{3}(G - B)}{(R - G) + (R - B)}\right) \quad (80)$$

$$S = 1 - 3 \min(r, g, b); \quad (81)$$

- CHARACTERISTICS:

- Device dependent
- Not perceptual uniform
- I is linear. Saturation is nonlinear: becomes unstable when intensity is near zero.
Hue is nonlinear: becomes unstable when intensity and saturation are near zero
- Intuitive
- Intensity I : dependent on viewing direction, object geometry, direction of the illumination, intensity and color of the illumination.
- Saturation S : dependent on highlights and a change in the color of the illumination.
- Hue H : dependent on the color of the illumination.

6.11 Color Ratio's

Color constant color ratio's have been proposed by Funt and Finlayson [8], Nayar and Bolle [17] and Gevers and Smeulders [9]. These color constant models are based on the ratio of surface albedos rather than the recovering of the actual surface albedo itself. The constraint is that the illumination can be modeled by the coefficient rule. The coefficient model of illumination change holds exactly in the case of narrow-band sensors. Although standard video camera's are not equipped with narrow-band filters, spectral sharpening could be applied [4] to achieve this to a large extend.

- COLOR MODELS: The color ratio's proposed by Nayar and Bolle are given by [17]:

$$N(C^{\vec{x}_1}, C^{\vec{x}_2}) \quad (82)$$

Funt and Finlayson by [8]:

$$F(C^{\vec{x}_1}, C^{\vec{x}_2}) \quad (83)$$

expressing color ratio's between two neighboring image locations, for $C \in \{R, G, B\}$, where \vec{x}_1 and \vec{x}_2 denote the image locations of the two neighboring pixels.

The color ratio's of Gevers and Smeulders [9] are given by:

$$m(C_1^{\vec{x}_1}, C_1^{\vec{x}_2}, C_2^{\vec{x}_1}, C_2^{\vec{x}_2}) \quad (84)$$

expressing the color ratio between two neighboring image locations, for $C_1, C_2 \in \{R, G, B\}$ where \vec{x}_1 and \vec{x}_2 denote the image locations of the two neighboring pixels.

- TRANSFORMATION:

Nayar and Bolle is given by [17]:

$$N(C^{\vec{x}_1}, C^{\vec{x}_2}) = \frac{C^{\vec{x}_1} - C^{\vec{x}_2}}{C^{\vec{x}_2} + C^{\vec{x}_1}} \quad (85)$$

Funt and Finlayson by [8]:

$$F(C^{\vec{x}_1}, C^{\vec{x}_2}) = \frac{C^{\vec{x}_1}}{C^{\vec{x}_2}} \quad (86)$$

and Gevers and Smeulders [9]:

$$m(C_1^{\vec{x}_1}, C_1^{\vec{x}_2}, C_2^{\vec{x}_1}, C_2^{\vec{x}_2}) = \frac{C_1^{\vec{x}_1} C_2^{\vec{x}_2}}{C_1^{\vec{x}_2} C_2^{\vec{x}_1}} \quad (87)$$

- CHARACTERISTICS:

- Device dependent
- Not perceptual uniform
- Nonlinear: Becomes unstable when intensity is near zero.
- Not intuitive
- N and F : dependent on the object geometry. m no dependencies.

7 Color and Image Search Engines

Very large digital image archives have been created and used in a number of applications including archives of images of postal stamps, textile patterns, museum objects, trademarks and logos, and views from everyday life as it appears in home videos and consumer photography. Moreover, with the growth and popularity of the World Wide Web, a tremendous amount of visual information is made accessible publicly. As a consequence, there is a growing demand for search methods retrieving pictorial entities from large image archives. Attempts have been made to develop general purpose image retrieval systems based on multiple features (e.g. color, shape and texture) describing the image content [1], [2], [5], [20], [27], for example. Further, a number of systems are available for retrieving images from the World Wide Web, for example [21], [22], [23], [26]. Aside from different representation schemes, these systems retrieve images on the basis color. The Picasso [3] and ImageRover [21] system the $L^*u^*v^*$ color space has been used for image retrieval. The QBIC system [5] evaluates similarity of global color properties using histograms based on a linear combination of the RGB . MARS [15] uses the the $L^*a^*b^*$ color space because the color space consists of perceptually uniform colors, which better matches the human perception of color. The PicToSeek system [10] is based on color models robust to a change in viewing direction, object geometry and illumination.

We have seen that each color system has its own characteristics. A number of systems are linear combinations of the R , G and B values, such as the XYZ and the $I1I2I3$ color system, or normalized with respect to intensity, such as the rgb and the xyz color system. The $U^*V^*W^*$

and the $L^*a^*b^*$ color systems have distances which reflect the perceived similarity. Each image retrieval application demands a specific color system. In Figure 7, a taxonomy is given of the different color systems. Further, the various color systems and their performance can be experienced within the PicToSeek and Pic2Seek systems on-line at:
<http://www.wins.uva.nl/research/isis/zomax/>.

Color system	Device indep.	Perc. Uniform	Linear	Intuitive	View point	Object shape	Highlights	Illum. Intensity	Illum. SPD
RGB	-	-	+	-	-	-	-	-	-
XYZ	+	-	+	-	-	-	-	-	-
Norm. rgb	-	-	-	-	+	+	-	+	-
Norm. xyz	+	-	-	-	+	+	-	+	-
$L^*a^*b^*$	+	+	-	-	-	-	-	-	-
$U^*V^*W^*$	+	+	-	-	-	-	-	-	-
I1I2I3	-	-	+	-	-	-	-	-	-
YIQ	-	-	+	-	-	-	-	-	-
YUV	-	-	+	-	-	-	-	-	-
Intensity	-	-	+	+	-	-	-	-	-
Hue	-	-	-	+	+	+	+	+	-
Saturation	-	-	-	+	+	+	-	+	-
F, N	-	-	-	-	+	-	-	+	+
M	-	-	-	-	+	+	-	+	+

Figure 7: a. Overview of the dependencies differentiated for the various color systems.
+ denotes that the condition is satisfied - denotes that the condition is not satisfied.

8 Conclusion

In this chapter, a survey on the basics of color has been given. Further, color models and ordering systems were discussed, and the state-of-the-art on color invariance has been presented. For the purpose of color-based image retrieval, a taxonomy on color systems has been provided. The color system taxonomy can be used to select the proper color system for a specific application.

References

- [1] Proceedings of First International Workshop on Image Databases and Multi Media Search, IDB-MMS '96, Amsterdam, The Netherlands, 1996.

-
- [2] Proceedings of IEEE Workshop on Content-based Access and Video Libraries, CVPR, 1997.
- [3] A. Del Bimbo, M. Mugnaini, P. Pala, F. Turco, *Visual Querying by Color Perceptive Regions*, Pattern recognition, Vol. 31, No. 9, pp. 1241-1253, 1998.
- [4] Finlayson, G.D., Drew, M.S., and Funt, B.V., *Spectral Sharpening: Sensor Transformation for Improved Color Constancy*, JOSA, 11, pp. 1553-1563, May, 1994.
- [5] Flickner, M. et al, *Query by Image and Video Content: the QBIC system*, IEEE Computer, Vol. 28, No. 9, 1995.
- [6] Forsyth, D., *A Novel Algorithm for Color Constancy*, International Journal of Computer Vision, Vol. 5, pp. 5-36, 1990.
- [7] Funt, B. V. and Drew, M. S., *Color Constancy Computation in Near-Mondrian Scenes*, In Proceedings of the CVPR, IEEE Computer Society Press, pp. 544-549, 1988.
- [8] Funt, B. V. and Finlayson, G. D., *Color Constant Color Indexing*, IEEE PAMI, 17(5), pp. 522-529, 1995.
- [9] T. Gevers and Arnold W.M. Smeulders, *Content-based Image Retrieval by Viewpoint-invariant Image Indexing*, Image and Vision Computing, (17)7, 1999.
- [10] T. Gevers and Arnold W.M. Smeulders, *Color Based Object Recognition*, Pattern Recognition, 32, pp. 453-464, March, 1999.
- [11] Goethe, *Farbenlehre*, 1840.
- [12] Kender, J. R., *Saturation, Hue, and Normalized Colors: Calculation, Digitization Effects, and Use*, Technical Report, Department of Computer Science, Carnegie-Mellon University, 1976.
- [13] E. H. Land, The Retinex Theory of Color Vision, Scientific American, 218(6), pp. 108-128, 1977.
- [14] Levkowitz, H. and Herman G. T., *GLHS: A Generalized Lightness, Hue, and Saturation Color Model*, CVGIP: Graphical Models and Image Processing, Vol. 55, No. 4, pp. 271-285, 1993.
- [15] S. Servetto, Y. Rui, K. Ramchandran, and T. S. Huang, *A Region-Based Representation of Images in MARS*, Journal on VLSI Signal Processing Systems, Volume 20, Issues 2, pp. 137-150, October 1998.
- [16] A. H. Munsell, A Color Notation, 1st edn (Geo. H. Ellis Co., Boston); 11th edn (edited and rearranged)(Munsell Color Company, Baltimore, 1946); Now available from the Munsell Color Group, New Windsor, New York; 1905.
- [17] S. K. Nayar, and R. M. Bolle, Reflectance Based Object Recognition, International Journal of Computer Vision, Vol. 17, No. 3, pp. 219-240, 1996
- [18] I. Newton, Opticks, (w. Innys, London; reprinted: Dover, New York, 1952), 1704.
- [19] Otha, Y., *Knowledge-Based Interpretation of Outdoor Natural Scenes*, Pitman Publishing, London, 1985.

-
- [20] Pentland, A., Picard, R. W. and Sclaroff, S., *Photobook: Tools for Content-based Manipulation of Image Databases*, In Proceedings of Storage and Retrieval for Image and Video Databases II, 2, 185, SPIE, Bellingham, Wash. pp. 34-47, 1994
- [21] Sclaroff, S., Taycher, L., La Cascia, M., *ImageRover: A Content-based Image Browser for the World Wide Web*, In Proceedings of IEEE Workshop on Content-based Access and Video Libraries, 1997.
- [22] C. Frankel, M. Swain and Athitsos Webseer: An Image Search Engine for the World Wide Web, *TR-96-14*, U. Chicago, 1996.
- [23] Smith, J. R. and Chang S.-F., *Visualseek: a Fully Automated Content-based Image Query System*, In Proceedings of ACM Multimedia 1996, 1996.
- [24] Shafer, S. A., *Using Color to Separate Reflection Components*, COLOR Res. Appl., 10(4), pp 210-218, 1985.
- [25] Swain, M. J. and Ballard, D. H., *Color Indexing*, International Journal of Computer Vision, Vol. 7, No. 1, pp. 11-32, 1991.
- [26] Gupta, A., *Visual Information Retrieval Technology: A Virage Perspective*, TR 3A, Virage Inc., 1996.
- [27] Proceedings of Visual97 Information Systems: The Second International Conference on Visual Information Systems, San Diego, USA, 1997.
- [28] Proceedings of Visual99 Information Systems: The Third International Conference on Visual Information Systems, Amsterdam, The Netherlands, 1999.

14-1 Modeling Light Intensities

The intensity of light seen on each surface of an object depends on the type of light sources in the vicinity and the surface characteristics of the object. Some objects have shiny surfaces, and some have dull, or matte, surfaces. In addition, some objects are constructed of opaque materials, while others are more or less transparent. A shading model to produce realistic intensities over the surfaces of an object must take these various properties into consideration.

Light Sources

When we view an object, we see the intensity of reflected light from the surfaces of the object. The light reflected from the surfaces comes from the various light sources around the object. If the object is transparent, we also see light from any sources that may be behind the object.

Light sources that illuminate an object are of two basic types, light-emitting sources and light-reflecting sources (Fig. 14-1). Light-emitting sources include light bulbs and the sun. Light-reflecting sources are illuminated surfaces of other objects, such as the walls of a room, that are near the object we are viewing. A surface that is not exposed directly to a light-emitting source will still be visible if nearby objects are illuminated. The multiple reflections of light from such nearby objects combine to produce a uniform illumination called ambient light, or background light.

When the dimensions of a light source are small compared to the size of an object, we can model it as a point source, as in Fig. 14-1. This approximation is used for most sources, such as the sun, that are sufficiently far from the object. In other cases, we have a distributed light source. This occurs when we have a large, nearby source, such as the long neon light in Fig. 14-2, whose area cannot be considered as infinitely small compared to the size of the illuminated object. Shading models based on the intensity laws for ambient light and point sources provide highly effective surface shading for objects with distant light sources. Only slight modifications to this model are needed to accommodate most "distributed sources" that may be included in a scene.

A shading model for calculating the intensity of light reflected from a surface can be established by considering contributions from the ambient light, sources and point sources in the vicinity of the surface. Both of these sources produce light reflections that are scattered in all directions. This scattered light is called diffuse reflection, and it results from the surface roughness, or graininess. A matte surface produces primarily diffuse reflections, so that the surface appears equally bright from all viewing directions. In addition to diffuse reflection, point sources create highlights, or bright spots, called specular reflection. This highlighting effect is more pronounced on shiny surfaces than on dull surfaces. These two types of reflections are illustrated in Fig. 14-3.

For transparent objects, we can expand the reflection-shading model to include light transmission effects. As in reflection, both diffuse and specular trans-

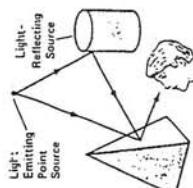


FIGURE 14-1

Surfaces of objects are illuminated by both light-emitting and light-reflecting sources.

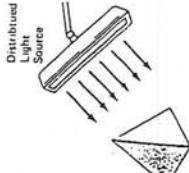


FIGURE 14-2

An object illuminated with a distributed light source.



FIGURE 14-3

Diffuse and specular reflections.

277

h2 276 - 281 (tot 10c reflected 1,111") 122e~
study
GCC - 121

h2 281 - 283 (photon /

14 SHADING AND COLOR MODELS

Realistic displays of objects are obtained by generating perspective projections with hidden surfaces removed and then applying shading and color patterns to the visible surfaces. A shading model is used to calculate the intensity of light that we should see when we view a surface. These intensity calculations are based on the optical properties of surfaces, the relative positions of the surfaces, and their orientation with respect to light sources. We first consider how intensity calculations can be modeled from the laws of optics, then explore some of the techniques for applying calculated intensities to surfaces. Finally, we discuss the structure and applications of color models useful in graphics packages.

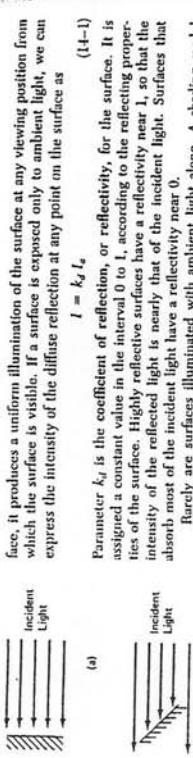


FIGURE 14-4
A surface perpendicular to the direction of the incident light
(a) is more illuminated than a surface at an angle (b) to the light direction.

face, it produces a uniform illumination of the surface at any viewing position from which the surface is visible. If a surface is exposed only to ambient light, we can express the intensity of the diffuse reflection at any point on the surface as

$$I_d = k_d I_a \quad (14-1)$$

Parameter k_d is the coefficient of reflection, or reflectivity, for the surface. It is assigned a constant value in the interval 0 to 1, according to the reflecting properties of the surface. Highly reflective surfaces have a reflectivity near 1, so that the intensity of the reflected light is nearly that of the incident light. Surfaces that absorb most of the incident light have a reflectivity near 0.

Rarely are surfaces illuminated with ambient light alone. A shading model calculates intensities according to Eq. 14-1 would shade all visible surfaces of an object with the same intensity. More realistic shading is obtained by including the effects of point sources in the shading model.

The calculation of diffuse reflection due to a point source of light depends on the angle of illumination. A surface that is perpendicular to the direction of the incident light appears brighter than a surface that is at an angle to the direction of the incoming light. As the angle increases, less of the incident light falls on the surface, as shown in Fig. 14-4. This figure shows a group of parallel light rays incident on two surfaces with different spatial orientations relative to the incident light direction from a distant source. A light source that is close to the surface, however, produces incident light rays that are not parallel (Fig. 14-5). To simplify calculations in our model, we assume that each point source in a scene is sufficiently far from the surface so that the light rays from that source are parallel as they strike the surface. Later we consider extensions to the model for other types of sources.

We can describe the orientation of a surface with a unit normal vector N and the direction of the light source with a unit vector L . The angle θ between these two vectors is called the angle of incidence (Fig. 14-6), and Lambert's cosine law states that the intensity of reflected light is proportional to $\cos \theta$. We can calculate $\cos \theta$ from the dot product of these two unit vectors:

$$\cos \theta = N \cdot L \quad (14-2)$$

A surface is illuminated by a point source only if the angle of incidence is between 0° and 90° ($\cos \theta$ is in the interval from 0 to 1). When $\cos \theta$ is negative, the light source is "behind" the surface.

We can expect the brightness of an illuminated surface to depend on the distance to the light source, since more distant sources are fainter than those that are nearer. If d represents the distance from a light source to a point on the surface position on the surface can be modeled as

$$I_d = \frac{k_d I_p}{d + d_0} (N \cdot L) \quad (14-3)$$

where parameter d_0 is a constant that is included to prevent the denominator from approaching zero when d is small. In an implementation of this intensity calculation, it is often convenient to assume that a point source at the viewer's position is illuminating a scene. Then d can be set to the distance from the surface position to the projection reference point, and d_0 can be adjusted until satisfactory shading patterns are obtained.

Equation 14-3 is an adaptation of Lambert's cosine law that has been found to produce realistic shading of surfaces. Theoretically, the light intensity arriving at

a surface is proportional to $1/d^2$, where d is the distance from the surface to the point source. However, most light sources are larger than points, and we can expect the intensity to decrease less rapidly. The factor $d + d_0$ in the denominator of Eq. 14-3 more accurately models the intensity reflections for surfaces at varying distances from a nearby light source.

Total diffuse reflection for a surface illuminated by ambient light and one point source is given by

$$I = k_d I_a + \frac{k_d I_p}{d + d_0} (N \cdot L) \quad (14-4)$$

If more than one point source is to be included in a scene, Eq. 14-4 is expanded to include terms for the additional light sources.

When color is to be included in a scene, Eq. 14-4 must be expressed in terms of the color components of the intensity. For an RGB video monitor, color components are red, green, and blue. Parameters for intensity and reflectivity then become three-element vectors, with one element for each of the color components. The vector representing the coefficient of reflection has components (k_{dr}, k_{dg}, k_{db}) . A green surface, for example, has a nonzero value for the green reflectivity component, k_{dg} , while the red and blue components are set to zero ($k_{dr} = k_{db} = 0$). For any light falling on this surface, the red and blue components of the light are absorbed, and only the green component is reflected. The intensity calculation for this example reduces to the single expression

$$I_g = k_d I_a + \frac{k_d I_p}{d + d_0} (N \cdot L) \quad (14-5)$$

using the green components of the intensity and reflectivity vectors. In general, a surface can reflect all three color components of the incident light, and three equations would be needed to calculate the three color components of the reflected light. Calculated intensity levels for each color component are then used to adjust the corresponding electron gun in the RGB monitor.

Specular Reflection

At certain viewing angles, a shiny surface reflects all incident light, independently of the reflectivity values. This phenomenon, called specular reflection, produces a spot of reflected light that is the same color as the incident light. Normally, objects are illuminated with white light, so that the specular reflection is a bright white spot. For an ideal reflector (perfect mirror), the angle of incidence and the angle of specular reflection are the same (Fig. 14-7). We use unit vector R to represent the direction for specular reflection. Unit vector L points in the direction of the viewer, and unit vector V points to the light source. Specular reflection can be seen with a perfect reflector only when V and R coincide ($\theta = 0$).

Real objects exhibit specular reflection over a range of positions about the

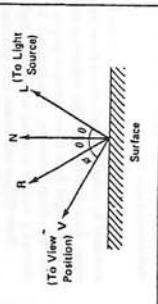


FIGURE 14-7
For a perfect reflector, angle of incidence θ is the same as the angle of reflection.

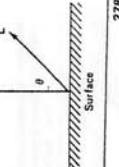


FIGURE 14-6
Angle of incidence θ between the light direction L and the surface normal N .



FIGURE 14-5
A nearby light source illuminates a surface with nonparallel rays of incident light.

vector R . Shiny surfaces have a narrow reflection range, and dull surfaces have a wider reflection range. Specular-reflection models must produce the highest intensity in the direction of R , with the intensity decreasing rapidly as the viewing angle ϕ increases.

One method for modeling specular reflection, developed by Phong But Tuong and called the Phong model, sets the intensity of specular reflection proportional to $\cos\phi$. The value assigned to n determines the type of surface that is being viewed. A very shiny surface is modeled with a large value for n (200 or more), and smaller values (down to 1) are used for duller surfaces. For a perfect reflector, n is infinite. A rough surface, such as carboard, might be assigned a value near 1. The effect of n on the angular range for viewing specular reflection is shown in Fig. 14-8.

Specular reflection also depends on the angle of incidence. In general, the intensity increases as the angle of incidence increases. This effect is incorporated into the specular-reflection model by making the intensity proportional to a reflection function $W(\theta)$, so that the complete reflection model is written as

$$I = k_d I_a + \frac{I_p}{d + d_o} [k_s k_d (N \cdot L) + W(\theta) \cos^2\phi] \quad (14-6)$$

The functional form for $W(\theta)$ depends on the surface material. Some materials, such as glass, exhibit very little specular reflection at smaller angles of incidence but increase the intensity of the specular reflection as θ approaches 90°. For these materials, W should vary from a value near 0 up to a value of 1 as the angle of incidence varies from 0° to 90°. Some materials have nearly constant specular reflection for all incidence angles, so that W could be assigned a constant value in the interval 0 to 1.

Since V and R are unit vectors in the viewing and reflection directions, we can set $\cos\phi = V \cdot R$ in Eq. 14-6. Also, for many applications, we can simplify the intensity calculations by setting $W(0)$ to a constant value k_s for the surface. The complete intensity model for reflection, due to ambient light and a single point source, can then be written as

$$I = k_d I_a + \frac{I_p}{d + d_o} [k_s (N \cdot L) + k_s (V \cdot R)^2] \quad (14-7)$$

In this model, constant values are assigned to parameters k_d , k_s , and d_o for each illuminated surface. Intensity values for the ambient light and the point sources are set, and values for the unit vectors are established. For each point on an illuminated surface, we calculate the relevant dot products and determine the intensity of the reflected light.

If the point source is far from a plane surface, the dot product $N \cdot L$ is approximately constant over the surface. Similarly, if the view reference point is sufficiently far from the surface, the product $V \cdot R$ is constant. When these simplifications can be made, the number of calculations is significantly reduced. If one or both of these simplifications cannot be made, dot products must be evaluated at each point. Since we can expect the values of these products to change only slightly from one point to the next, coherence methods can be used to calculate the dot products across a scan line.

Vectors L and N can be used to determine vector R for the calculation of $\cos\phi$. Alternatively, we can evaluate $\cos\phi$ directly in terms of vectors I_a , N , and V as follows. The direction of the normal vector N is along the bisector of the angle between R and L (Fig. 14-9) as

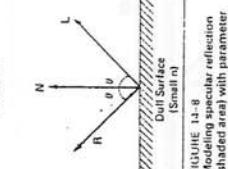
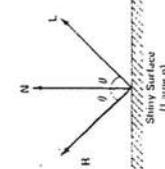


FIGURE 14-8
Modeling specular reflection
(shaded area) with parameter
 n .

FIGURE 14-9
Vector B is along the bisector
of the angle between vectors V
and L .

$$B = \frac{V + L}{|V + L|}$$

$$V \cdot R = N \cdot B$$

281
Sec. 14.1 Modeling Light
Intensities

14-8
14-9

Then, vector B can then be used to calculate $\cos\phi$, since

Other methods for modeling light intensities have been developed. One technique, developed by Torrance and Sparrow and adapted to graphics applications by Blinn, divides each surface in a scene into a set of tiny planes. Each of the small planes is assumed to be an ideal reflector, and the planes are oriented randomly over the total surface. A Gaussian distribution function is used to set the orientation of each plane. The specular reflection for the surface is calculated as the total contribution from the small planes as a function of the intensity I_p from a distant point source and the vectors N , V , and L .

Refracted Light

When a transparent object is to be modeled, the intensity equations must be modified to include contributions from light sources in front of the object. In most cases, these sources are light-reflecting surfaces of other objects, as in Fig. 14-10. Reflected light from these surfaces passes through the transparent object and modifies the object intensity, as calculated by Eq. 14-6 or Eq. 14-7. Light passing through a surface is called transmitted light or refracted light.

Both diffuse refraction and specular refraction can take place at the surfaces of an object. Diffuse effects are important when a partially transparent surface, such as frosted glass, is to be modeled. Light passing through such materials is scattered so that a blurred image of background objects is obtained. Diffuse refractions can be generated by decreasing the intensity of the refracted light and spreading intensity contributions at each point on the refracting surface onto a finite area. These manipulations are time-consuming, and most shading techniques employ only specular effects.

Realistic models of transparent materials, such as clear glass, can be developed by adding specular-refraction contributions to the reflected intensity calculations. When light is incident upon a transparent surface, part of it is reflected and

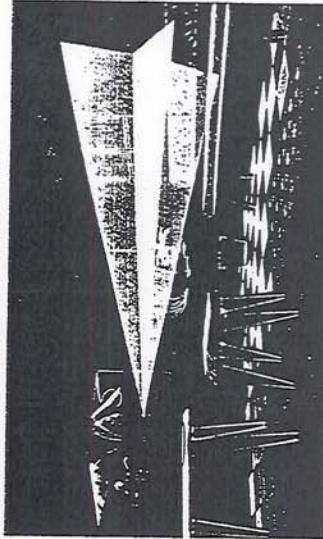


FIGURE 14-10
A transparent object, modeled here as a thin sheet of paper, transmits some light from
sources behind the object.
From Prentice-Hall,
courtesy of Robert Abel &
Associates.

657

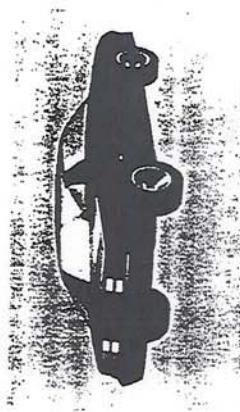


FIGURE 14-14
Illumination patterns produced with an intensity model, displaying diffuse reflections, specular reflections, and reflected light. Courtesy David Warr, General Motors Research Lab.

Texture and Surface Patterns

The shading model we have discussed for calculating light intensities provides a smooth shading for every surface in a scene. However, many objects do not have smooth surfaces. Surface texture is needed to model accurately such objects as brick walls, gravel roads, and shag carpets. In addition, some surfaces contain patterns that must be included in the shading model. The surface of a vase could contain a painted design; a water glass might have the family crest engraved into the surface; a tennis court contains markings for the alleys, service areas, and base line; and a four-lane highway has dividing lines and other markings, such as oil spills and tire skids. Figure 14-15 illustrates the display of surfaces modeled with texture and patterns.

Intensity values furnished by a shading model can be adjusted to accommodate surface texture by altering the surface normal so that it is a function of position over the surface. If we allow the surface normal to vary randomly, we can obtain an irregular textured surface such as that of a raisin. A repeating function can be used to model a more regular surface, such as a sculptured carpet that contains a repeated texture pattern. An irregular surface can also be modeled by dividing the surface into a collection of small, randomly oriented surfaces. In addition, we could allow the coefficient of reflection to vary with position so as to obtain greater variations in intensity.

part is refracted (Fig. 14-11). Because the speed of light is different in different materials, the path of the refracted light is different from that of the incident light. The direction of the refracted light, specified by the angle of refraction, is a function of the index of refraction of a material. Specifically, the angle of refraction θ' is calculated from the angle of incidence θ , the index of refraction n of the material outside the surface (usually air), and the index of refraction n' of the surface material according to the law of refraction:

$$n \sin \theta = n' \sin \theta' \quad (14-10)$$

Actually, the index of refraction of a material is a function of the wavelength of the incident light, so that the different color components of a light ray will be refracted at different angles. For most applications, we can use an average value of refraction for the different materials that are modeled in a scene. The index of refraction of air is approximately 1, and that of crown glass is about 1.5. Using these values in Eq. 14-10 with an angle of incidence of 30° yields an angle of refraction of about 19° . Figure 14-12 illustrates the changes in the path direction for a light ray refracted through a glass object. The overall effect of the refraction is to shift the incident light to a parallel path. Since the calculations of the trigonometric functions in Eq. 14-10 are time-consuming, refraction effects could be modeled by simply shifting the path of the incident light a small amount.

A simpler procedure for modeling transparent objects is to ignore the path shifts altogether. In effect, this approach assumes that there is no change in the index of refraction from one material to another, so that the angle of refraction is always the same as the angle of incidence. This method speeds up the calculation of intensities and produces realistic displays for thin objects, such as champagne glasses.

We can modify the reflection-shading model to include refraction effects by projecting the intensity of background objects to the front surface of the transparent object, as shown in Fig. 14-13. The intensity I_b of a background object is added to the intensity I_r of the transparent object. A more realistic shading pattern is obtained by using a "refraction coefficient" r to weight the reflected and refracted intensity contributions to produce a total intensity I_t , calculated as

$$I_t = I_r + (1 - r)I_b \quad (14-11)$$

For highly transparent objects, we assign r a value near 0. Opaque objects transmit no light from background objects, and we can set $r = 1$ for these materials. It is also possible to allow r to be a function of position over the surface, so that different parts of an object can transmit more or less background intensity according to the values assigned to r .

Background objects relative to a transparent object can be identified with a hidden-surface method that sorts surfaces according to depth. Once the surfaces have been sorted, those that are visible through the transparent object are identified. Then the intensity of corresponding points along the viewing direction is combined according to Eq. 14-11 to produce the total shading pattern for the transparent object.

An example of the application of an intensity model to the surfaces of an automobile is shown in Fig. 14-14. The surfaces of the car display specular reflections (bright spots and lines), diffuse reflections, and refracted light through the windows. An intensity model employing multiple light sources and "light controls" was used to reproduce the lighting effects typically available in a photography studio. The light controls allowed the light direction and the light concentration to be varied so that both spotlight and floodlight effects could be simulated.

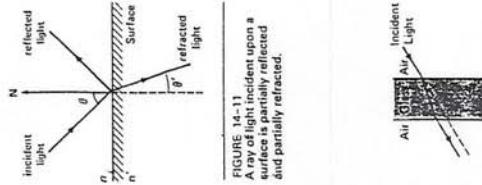
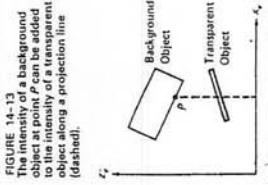


FIGURE 14-11
Refraction of light through a glass object. The emerging refracted ray travels along a path that is parallel to the incident light path (dashed line).



282

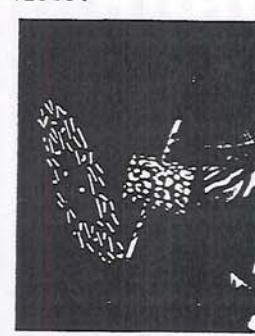


FIGURE 14-15
Combining surface patterns
and textures with an intensity
model. Courtesy Robert Arnal &
Associates.

19 SHAPE ANALYSIS

Several qualitative and quantitative techniques have been developed for characterizing the shape of objects within an image. These techniques are useful for classifying objects in a pattern recognition system and for symbolically describing objects in an image understanding system. Some of the techniques apply only to binary valued images, while others can be extended to gray level images.

19.1. TOPOLOGICAL ATTRIBUTES

Topological shape attributes are properties of a shape that are invariant under "rubbersheet" transformation (1-3). Such a transformation or mapping can be visualized as the stretching of a rubber sheet containing the image of an object of a given shape to produce some spatially distorted object. Mappings that require cutting of the rubber sheet or connection of one part to another are not permissible. Metric distance is clearly not a topological attribute because distance can be altered by rubber sheet stretching. Also, the concepts of perpendicularity and parallelism between lines are not topological properties. Connectivity is a topological attribute. Figure 19.1-1a contains a binary valued image containing two connected object components; Figure 19.1-1b is a spatially stretched version of the same image. Clearly, there are no stretching operations that can either increase or decrease the connectivity of the objects in the stretched image. Connected components of an object may contain holes, as illustrated in Figure 19.1-1c. The number of holes is obviously unchanged by a topological mapping.

There is a fundamental relationship between the number of connected object components C and the number of object holes H in an image called the Euler

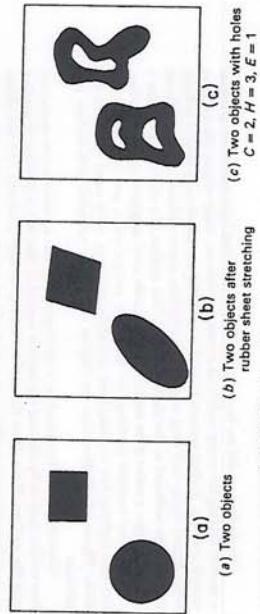


FIGURE 19.1-1. Examples of topological attributes.



FIGURE 19.1-2. Definitions of convex shape descriptors.

number, as defined by

$$E = C - H \quad (19.1-1)$$

The Euler number is also a topological property because C and H are topological attributes.

Irregularly shaped objects can be described by their topological constituents. Consider the tubular shaped object letter R of Figure 19.1-2a, and imagine a rubber band stretched about the object. The region enclosed by the rubber band is called the convex hull of the object. The set of points within the convex hull, that are not in the object, form the convex deficiency of the object. There are two types of convex deficiencies: regions totally enclosed by the object, called lakes; and regions lying between the convex hull perimeter and the object, in terms of its convex hull and convex deficiency. For objects represented over rectilinear grids, the definition of the convex hull must be modified slightly to remain meaningful. Objects such as discretized circles and triangles clearly should be judged as being convex even though their boundaries are jagged. This apparent difficulty can be handled by considering a rubber band to be stretched about the discretized object. A pixel lying totally within the rubber band, but not in the object, is a member of the convex deficiency. Sklansky *et al.* (4, 5) have developed practical algorithms for computing the convex attributes of discretized objects.

19.2. DISTANCE, PERIMETER, AND AREA MEASUREMENTS

Distance is a real-valued function $d[(j_1, k_1), (j_2, k_2)]$ of two image points (j_1, k_1) and (j_2, k_2) satisfying the following properties (6)

$$d[(j_1, k_1), (j_2, k_2)] \geq 0 \quad (19.2-1a)$$

$$d[(j_1, k_1), (j_2, k_2)] = d[(j_2, k_2), (j_1, k_1)] \quad (19.2-1b)$$

$$d[(j_1, k_1), (j_2, k_2)] + d[(j_2, k_2), (j_3, k_3)] \geq d[(j_1, k_1), (j_3, k_3)] \quad (19.2-1c)$$

There are a number of distance functions that satisfy the defining properties. The most common measures encountered in image analysis are the Euclidean distance

$$d_E = [(j_1 - j_2)^2 + (k_1 - k_2)^2]^{1/2} \quad (19.2-2a)$$

the magnitude distance

$$d_M = |j_1 - j_2| + |k_1 - k_2| \quad (19.2-2b)$$

and the maximum value distance

$$d_X = \max\{|j_1 - j_2|, |k_1 - k_2|\} \quad (19.2-2c)$$

In discrete images the coordinate differences $(j_1 - j_2)$ and $(k_1 - k_2)$ are integers, but the Euclidean distance is usually not an integer.

Perimeter and area measurements are meaningful only for binary images. Consider a discrete binary image containing one or more objects, where $F(j, k) = 1$ if a pixel is part of the object and $F(j, k) = 0$ for all non-object or background pixels.

The perimeter of each object is the count of the number of pixel sides traversed around the boundary of the object starting at an arbitrary initial boundary pixel and returning to the initial pixel. The area of each object within the image is simply the count of the number of pixels in the object for which $F(j, k) = 1$. As an example, for a 2×2 pixel square, the object area is $A_O = 4$ and the object perimeter is $P_O = 8$. An object formed of three diagonally connected pixels possesses $A_O = 3$ and $P_O = 12$.

The enclosed area of an object is defined to be the total number of pixels for which $F(j, k) = 0$ or 1 within the outer perimeter boundary P_E of the object. The enclosed area can be computed during a boundary following process while the perimeter is being computed (7, 8). Assume that the initial pixel in the boundary following process is the first black pixel encountered in a raster scan of the image. Then proceeding in a clockwise direction around the boundary, a crack code $C(p)$, as defined in Section 18.6, is generated for each side p of the object perimeter such that $C(p) = 0, 1, 2, 3$ for directional angles 0, 90, 180, 270 degrees respectively. The enclosed area is

$$A_E = \sum_{p=1}^{P_E} j(p-1)\Delta k(p) \quad (19.2-3a)$$

where P_E is the perimeter of the enclosed object and

$$j(p) = \sum_{i=1}^p \Delta i(i) \quad (19.2-3b)$$

TABLE 19.2-1. Example of perimeter and area computation

p	$C(p)$	$\Delta l(p)$	$\Delta k(p)$	$j(p)$	$A(p)$
1	0	0	-1	0	0
2	3	-1	0	-1	0
3	0	1	0	0	-1
4	1	0	1	0	-1
5	0	0	0	0	-1
6	3	0	0	0	-1
7	2	-1	0	0	0
8	3	0	0	-1	0
9	2	0	0	-1	0
10	2	0	0	-1	0
11	1	-1	0	0	0
12	1	0	0	0	0

with $j(0) = 0$. The delta terms are defined by

$$\Delta j(p) = \begin{cases} 1 & \text{if } C(p) = 1 \\ 0 & \text{if } C(p) = 0 \text{ or } 2 \\ -1 & \text{if } C(p) = 3 \end{cases} \quad (19.2-3c)$$

$$\Delta k(p) = \begin{cases} 1 & \text{if } C(p) = 0 \\ 0 & \text{if } C(p) = 1 \text{ or } 3 \\ -1 & \text{if } C(p) = 2 \end{cases} \quad (19.2-3d)$$

Table 19.2-1 gives an example of computation of the enclosed area of the following four pixel object

0	0	0	0
0	1	0	1
0	1	1	0
0	0	0	0

19.2.1. Bit Quads

Gray [9] has devised a systematic method of computing the area and perimeter of binary objects based on matching the logical state of regions of an image to binary patterns. Let $n\{Q\}$ represent the count of the number of matches between image pixels and the pattern Q within the parentheses. By this definition, the object area is then

$$A_o = n\{1\} \quad (19.2-4)$$

If the object is completely enclosed by a border of white pixels, then its perimeter is equal to

$$P_o = 2n\{0\} + 2n\{\begin{matrix} 0 \\ 1 \end{matrix}\} \quad (19.2-5)$$

Now consider the following set of 2×2 pixel patterns called bit quads defined in Figure 19.2-1. The object area and object perimeter of an image can be

0	0
0	0

0	1
0	0

0	0
0	1

0	0
1	0

1	0
0	1

1	0
1	1

FIGURE 19.2-1. Bit quad patterns.

Figure

Figure

Figure

Figure

expressed in terms of the number of bit quad counts in the image as

$$A_o = \frac{1}{4} [n\{Q_1\} + 2n\{Q_2\} + 3n\{Q_3\} + 4n\{Q_4\} + 2n\{Q_b\}] \quad (19.2-6a)$$

$$P_o = n\{Q_1\} + n\{Q_2\} + n\{Q_3\} + 2n\{Q_b\} \quad (19.2-6b)$$

These area and perimeter formulas may be in considerable error if they are utilized to represent the area of a continuous object that has been discretized. More accurate formulas for such applications have been derived by Duda (10):

$$A_o = \frac{1}{4}n\{Q_1\} + \frac{1}{2}n\{Q_2\} + \frac{7}{8}n\{Q_3\} + n\{Q_4\} + \frac{1}{4}n\{Q_b\} \quad (19.2-7a)$$

$$P_o = n\{Q_2\} + \frac{1}{\sqrt{2}}[n\{Q_1\} + n\{Q_3\} + 2n\{Q_b\}] \quad (19.2-7b)$$

Bit quad counting provides a very simple means of determining the Euler number of an image. Gray (9) has determined that, under the definition of four-connectivity, the Euler number can be computed as

$$E = \frac{1}{4}[n\{Q_1\} - n\{Q_3\} + 2n\{Q_b\}] \quad (19.2-8a)$$

and for eight-connectivity

$$E = \frac{1}{4}[n\{Q_1\} - n\{Q_3\} - 2n\{Q_b\}] \quad (19.2-8b)$$

It should be noted that while it is possible to compute the Euler number E of an image by local neighborhood computation, neither the number of connected components C nor the number of holes H , for which $E = C - H$, can be separately computed by local neighborhood computation.

19.2.2. Geometric Attributes

With the establishment of distance, area, and perimeter measurements, various geometric attributes of objects can be developed. In the following it is assumed that the number of holes with respect to the number of objects is small, i.e. E is approximately equal to C . The circularity of an object is defined as

$$C_o = \frac{4\pi A_o}{(P_o)^2} \quad (19.2-9)$$

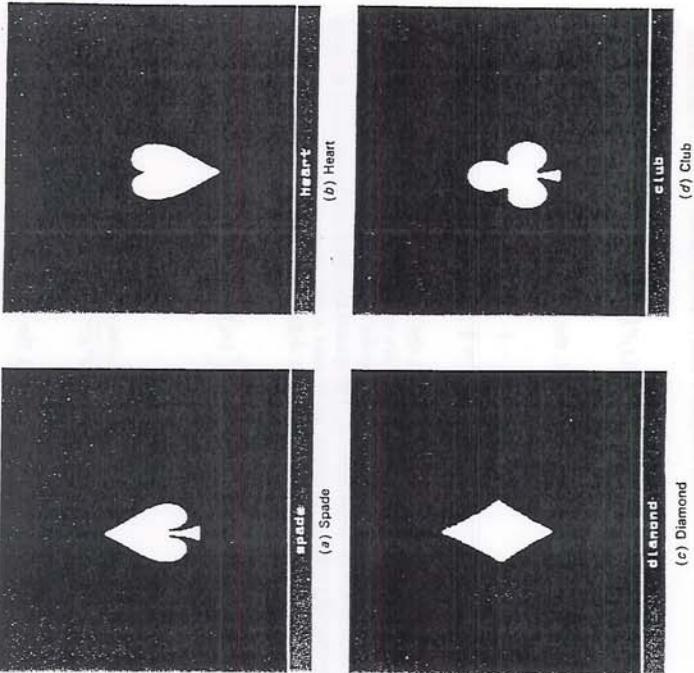


FIGURE 19.2-2. Playing card symbols.

TABLE 19.2-2. Geometric attributes of playing card symbols

Attribute	Spade	Heart	Diamond	Club
Outer perimeter	652	512	548	668
Enclosed area	8,421	8,681	8,562	8,820
Average area	8,421	8,681	8,562	8,820
Average perimeter	652	512	548	668
Average length	326	256	274	334
Average width	25.8	33.9	31.3	26.4
Circularity	0.25	0.42	0.36	0.25

For images containing thin objects such as typewritten or script characters, the average object length and width can be approximated by

$$L_A = \frac{P_A}{2} \quad (19.2-12)$$

$$W_A = \frac{2A_A}{P_A} \quad (19.2-13)$$

These simple measures are useful for distinguishing gross characteristics of an image. For example, does it contain a multitude of small point-like objects, or fewer blob-like objects of larger size; are the objects fat or thin? Figure 19.2-2 contains images of playing card symbols. Table 19.2-2 lists the geometric attributes of these objects.

19.3. SPATIAL MOMENTS

From probability theory the (m, n) th moment of the joint probability density $p(x, y)$ is defined as

$$M(m, n) = \int_{-\infty}^{\infty} x^m y^n p(x, y) dx dy \quad (19.3-1)$$

The central moment is given by

$$U(m, n) = \int_{-\infty}^{\infty} [x - \eta_x]^m [y - \eta_y]^n p(x, y) dx dy \quad (19.3-2)$$

where η_x and η_y are the marginal means of $p(x, y)$. These classical relationships of probability theory have been applied to shape analysis by Hu (11) and Alt (12). The concept is quite simple. The joint probability density $p(x, y)$ of

Shape Analysis

637

Spatial Moments

Eq. 19.3-1 and Eq. 19.3-2 is replaced by the continuous image function $F(x, y)$. Object shape is characterized by a few of the low-order moments. Abu-Mostafa and Psaltis (13, 14) have investigated the performance of spatial moments as features for shape analysis.

19.3.1. Discrete Image Spatial Moments

The spatial moment concept can be extended to discrete images by forming spatial summations over a discrete image function $F(j, k)$. The literature (15–17) is notationally inconsistent on the discrete extension because of the differing relationships defined between the continuous and discrete domains. Following the notation established in Chapter 14, the (m, n) th spatial moment is defined as

$$M_U(m, n) = \sum_{j=1}^J \sum_{k=1}^K (x_k)^m (y_k)^n F(j, k) \quad (19.3-3)$$

where, with reference to Figure 14.1-1, the scaled coordinates are

$$x_k = k - \frac{1}{2} \quad (19.3-4a)$$

$$y_j = J + \frac{1}{2} - j \quad (19.3-4b)$$

The origin of the coordinate system is the lower left corner of the image. This formulation results in moments that are extremely scale dependent; the ratio of second-order ($m + n = 2$) to zero-order ($m = n = 0$) moments can vary by several orders of magnitude (18). The spatial moments can be restricted in range by spatially scaling the image array over a unit range in each dimension. The (m, n) th scaled spatial moment is then defined as

$$M(m, n) = \frac{1}{J^K} \sum_{j=1}^J \sum_{k=1}^K (x_k)^m (y_k)^n F(j, k) \quad (19.3-5)$$

Clearly,

$$M(m, n) = \frac{M_U(m, n)}{J^K K^m} \quad (19.3-6)$$

It is instructive to explicitly identify the lower order spatial moments. The zero-order moment

$$M(0, 0) = \sum_{j=1}^J \sum_{k=1}^K F(j, k) \quad (19.3-7)$$

$F(j, k)$ is a binary image, then its surface is equal to its area. The first order row moment is

$$M(1, 0) = \frac{1}{K} \sum_{j=1}^J \sum_{k=1}^K x_k F(j, k) \quad (19.3-8)$$

and the first-order column moment is

$$M(0, 1) = \frac{1}{J} \sum_{j=1}^J \sum_{k=1}^K y_j F(j, k) \quad (19.3-9)$$

Table 19.3-1 lists the scaled spatial moments of several test images. These images include unit-amplitude gray scale versions of the playing card symbols of Figure 19.2-2, several rotated, minified, and magnified versions of these symbols, as shown in Figure 19.3-1, as well as an elliptically shaped gray scale object shown in Figure 19.3-2.

The ratios

$$\bar{x}_k = \frac{M(1, 0)}{M(0, 0)} \quad (19.3-10a)$$

$$\bar{y}_j = \frac{M(0, 1)}{M(0, 0)} \quad (19.3-10b)$$

of the first-order to the zero-order spatial moments define the centroid of an image. The centroid, called the center of gravity, is the balance point of the image function $F(j, k)$ such that the mass of $F(j, k)$ left and right of \bar{x}_k and above and below \bar{y}_j is equal.

With the centroid established, it is possible to define the scaled spatial central moments of a discrete image, in correspondence with Eq. 19.3-2, as

$$U(m, n) = \frac{1}{J^n K^m} \sum_{j=1}^J \sum_{k=1}^K [x_k - \bar{x}_k]^m [y_j - \bar{y}_j]^n F(j, k) \quad (19.3-11)$$

For future reference, the (m, n) th unscaled spatial central moment is defined as

$$U_v(m, n) = \sum_{j=1}^J \sum_{k=1}^K [x_k - \bar{x}_k]^m [y_j - \bar{y}_j]^n F(j, k) \quad (19.3-12)$$

where

$$\bar{x}_k = \frac{M_v(1, 0)}{M_v(0, 0)} \quad (19.3-13a)$$

$$\bar{y}_j = \frac{M_v(0, 1)}{M_v(0, 0)} \quad (19.3-13b)$$

Image	$M(0, 0)$	$M(1, 0)$	$M(2, 0)$	$M(3, 0)$	$M(4, 0)$	$M(5, 0)$	$M(1, 1)$	$M(2, 1)$	$M(3, 1)$	$M(4, 1)$	$M(1, 2)$	$M(2, 2)$	$M(3, 2)$	$M(4, 2)$
Spade	8.21938	4.01375	4.28128	1.97612	2.08986	2.263.11	980.81	1.028.31	1.144.36	1.213.73	1.213.73	1.213.73	1.213.73	1.213.73
Rotated spade	8.21599	4.19839	3.968.30	2.149.35	2.021.65	1.949.89	1.111.69	1.038.04	1.081.72	1.105.73	1.156.35	1.164.43	1.171.61	1.171.61
Heart	8.61379	4.23835	4.414.36	2.153.90	2.223.79	2.143.28	2.143.28	2.143.28	2.143.28	2.143.28	2.143.28	2.143.28	2.143.28	2.143.28
Rotated heart	2.10497	1.01738	1.713.64	1.713.64	1.713.64	1.713.64	1.713.64	1.713.64	1.713.64	1.713.64	1.713.64	1.713.64	1.713.64	1.713.64
Diamond	8.56182	4.349.00	4.304.71	2.222.43	2.293.10	2.267.24	2.196.97	2.196.97	2.196.97	2.196.97	2.196.97	2.196.97	2.196.97	2.196.97
Club	8.781.71	4.323.54	4.200.10	2.150.47	2.215.32	2.344.02	2.196.98	2.196.98	2.196.98	2.196.98	2.196.98	2.196.98	2.196.98	2.196.98
Ellipse	8.721.74	4.326.93	4.326.93	2.226.61	2.189.76	2.189.76	2.175.78	2.175.78	2.175.78	2.175.78	2.175.78	2.175.78	2.175.78	2.175.78

TABLE 19.3-1. Scaled spatial moments of test images

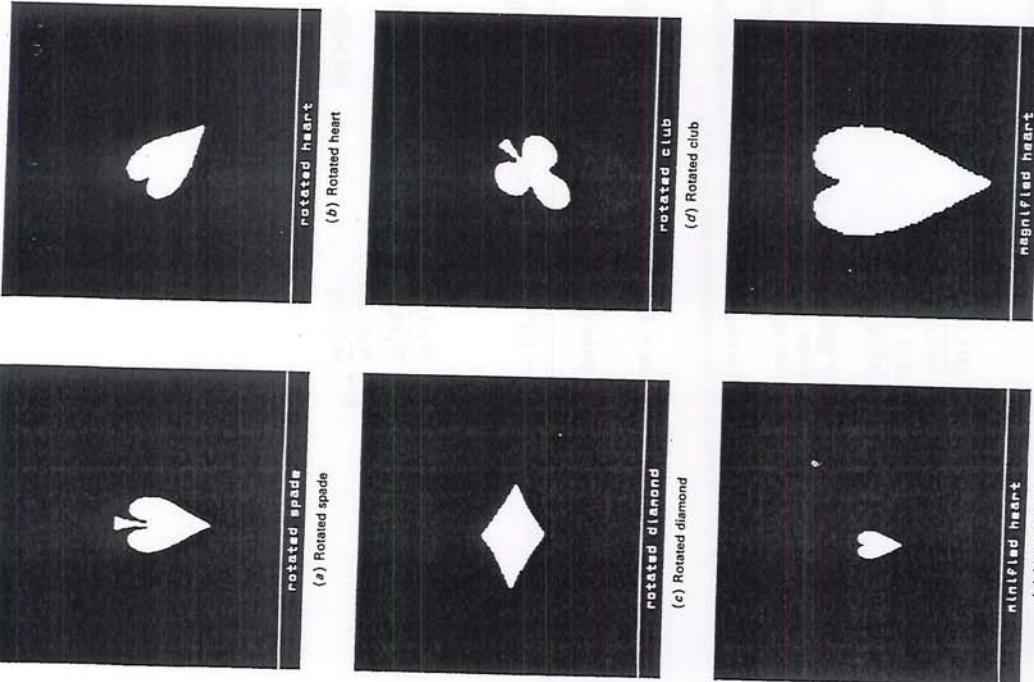


FIGURE 19-3-1. Rotated, magnified, and minified playing card symbols.

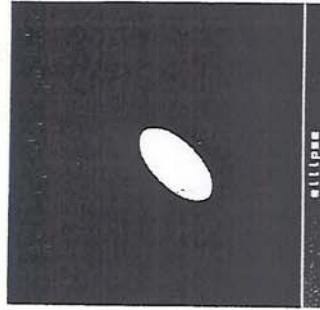


FIGURE 19-3-2. Elliptically shaped object.

It is easily shown that

$$U(m, n) = \frac{U_U(m, n)}{J^n K^m} \quad (19.3-14)$$

The three second-order scaled central moments are the row moment of inertia

$$U(2, 0) = \frac{1}{K^2} \sum_{j=1}^J \sum_{k=1}^K [x_k - \bar{x}_k]^2 F(j, k) \quad (19.3-15)$$

the column moment of inertia

$$U(0, 2) = \frac{1}{J^2} \sum_{j=1}^J \sum_{k=1}^K [y_j - \bar{y}_j]^2 F(j, k) \quad (19.3-16)$$

and the row-column cross moment of inertia

$$U(1, 1) = \frac{1}{JK} \sum_{j=1}^J \sum_{k=1}^K [x_k - \bar{x}_k][y_j - \bar{y}_j] F(j, k) \quad (19.3-17)$$

The central moments of order three can be computed directly from Eq. 9.3-11 for $m + n = 3$, or indirectly according to the following relations.

$$U(3, 0) = M(3, 0) - 3\bar{y}_j M(2, 0) + 2[\bar{y}_j]^2 M(1, 0) \quad (19.3-18a)$$

$$U(2, 1) = M(2, 1) - 2\bar{y}_j M(1, 1) - \bar{x}_k M(2, 0) + 2[\bar{y}_j]^2 M(0, 1) \quad (19.3-18b)$$

$$U(1, 2) = M(1, 2) - 2\bar{x}_k M(1, 1) - \bar{y}_j M(0, 2) + 2[\bar{x}_k]^2 M(0, 1) \quad (19.3-18c)$$

$$U(0, 3) = M(0, 3) - 3\bar{x}_k M(0, 2) + 2[\bar{x}_k]^2 M(0, 1) \quad (19.3-18d)$$

TABLE 19.3-2. Centres of gravity and scaled spatial central moments of test images

Image	Horizontal	Vertical	COG	COG	$U(2, 0)$	$U(1, 1)$	$U(3, 0)$	$U(0, 2)$	$U(2, 1)$	$U(1, 2)$	$U(0, 3)$
Spade	0.488	0.521	162.40	-0.653	33.21	-0.366	162.07	-0.325	0.026	-0.025	0.363
Heart	0.510	0.504	0.483	0.194	36.506	33.215	-0.013	0.024	-0.002	0.027	-0.357
Mitred heart	0.496	0.504	162.07	-0.366	33.215	-0.013	0.024	-0.002	0.027	-0.021	-0.831
Spade	0.488	0.521	162.40	-0.653	33.21	-0.366	162.07	-0.325	0.026	-0.025	0.363
Heart	0.510	0.504	0.483	0.194	36.506	33.215	-0.013	0.024	-0.002	0.027	-0.357
Mitred heart	0.496	0.504	162.07	-0.366	33.215	-0.013	0.024	-0.002	0.027	-0.021	-0.831
Rounded diamond	0.502	0.505	0.549	0.198	21.834	-0.085	13.366	-0.158	-0.026	0.009	-0.029
Club	0.492	0.512	42.198	-0.239	30.073	-0.545	27.979	-0.073	-0.545	-0.029	-0.053
Ellipse	0.496	0.502	0.502	0.236	29.236	-0.050	17.913	-0.268	-0.268	-0.000	-0.000

Table 19-5-2 presents the horizontal and vertical centers of gravity and the scaled central spatial moments of the test images.

$$U = \begin{bmatrix} U(2,0) & U(1,1) \\ U(1,1) & U(0,2) \end{bmatrix} \quad (19.3-19)$$

Performing a singular value decomposition of the covariance matrix results in the diagonal matrix

E^{TUE} = Λ (193-20)

where the columns of

$$\mathbf{E} = \begin{bmatrix} e_{11} & e_{12} \\ e_{21} & e_{22} \end{bmatrix} \quad (19.3-21)$$

the eigenvectors of Π and

contains the eigenvalues of \mathbf{U} . Expressions for the eigenvalues can be derived explicitly. They are

$$\begin{aligned}\lambda_1 &= \frac{1}{2} [U(2,0) + U(0,2)] \\ &\quad + \frac{1}{2} [(U(2,0)^2 + U(0,2)^2 - 2U(2,0)U(0,2) + 4U(1,1)^2)^{1/2}].\end{aligned}$$

$$\lambda_2 = \frac{1}{2} [U(2,0) + U(0,2)] - \frac{1}{4} [(U(2,0)^2 + U(0,2)^2 - 2U(2,0)U(0,2)) + 4U(1,1)^2]^{1/2} \quad (19.3-23a)$$

Let $\lambda_M = \text{MAX}[\lambda_1, \lambda_2]$ and $\lambda_N = \text{MIN}[\lambda_1, \lambda_2]$, and let the orientation angle θ be defined as

$$\begin{aligned} \theta &= \tan^{-1} \left\{ \frac{e_{21}}{e_{11}} \right\} && \text{(19.3-24a)} \\ \theta &= \tan^{-1} \left\{ \frac{e_{22}}{e_{11}} \right\} && \text{(19.3-24b)} \end{aligned}$$

(19.3-24a)

19.3-24b)

The orientation angle can be expressed explicitly as

$$\theta = \tan^{-1} \left\{ \frac{\lambda_M - U(0,2)}{U(1,1)} \right\} \quad (19.3-24c)$$

The eigenvalues λ_M and λ_N and the orientation angle θ define an ellipse, as shown in Figure 19.3-2, whose major axis is λ_M and whose minor axis is λ_N . The major axis of the ellipse is rotated by the angle θ with respect to the horizontal axis. This elliptically-shaped object has the same moments of inertia along the horizontal and vertical axes and the same moments of inertia along the principal axes as does an actual object in an image. The ratio

$$R_A = \frac{\lambda_N}{\lambda_M} \quad (19.3-25)$$

of the minor-to-major axes is a useful shape feature.

Table 19.3-3 provides moment of inertia data of the test images. It should be noted that the orientation angle can only be determined to within plus or minus $\pi/2$ radians.

Hu [11] has proposed a normalization of the unscaled central moments, defined by Eq. 19.3-12, according to the relation

$$V(m, n) = \frac{U_v(m, n)}{[M(0,0)]^{\alpha}} \quad (19.3-26a)$$

where

$$\alpha = \frac{m+n}{2} + 1 \quad (19.3-26b)$$

TABLE 19.3-3. Moment of inertia data of test images

Image	Largest eigenvalue	Smallest eigenvalue	Orientation (radians)	Eigenvalue ratio
Spade	33.386	16.215	-0.153	0.487
Rotated spade	32.223	16.200	-1.549	0.488
Heart	36.408	16.376	1.561	0.449
Rotated heart	36.421	16.400	-0.794	0.450
Magnified heart	589.190	262.290	1.562	0.445
Minified heart	2.165	0.984	1.360	0.454
Diamond	42.189	13.324	1.360	0.316
Rotated diamond	42.223	13.341	-0.030	0.316
Club	37.982	21.831	-1.356	0.575
Rotated club	38.073	21.831	0.802	0.573
Ellipse	47.149	11.324	0.785	0.240

for $m+n=2,3,\dots$. These normalized central moments have been used by Hu to develop a set of seven compound spatial moments that are invariant in the continuous image domain to translation, rotation, and scale change. The Hu invariant moments are defined below.

$$h_1 = V(2,0) + V(0,2) \quad (19.3-27a)$$

$$h_2 = [V(2,0) - V(0,2)]^2 + 4[V(1,1)]^2 \quad (19.3-27b)$$

$$h_3 = [V(3,0) - 3V(1,2)]^2 + [V(0,3) - 3V(2,1)]^2 \quad (19.3-27c)$$

$$h_4 = [V(3,0) + V(1,2)]^2 + [V(0,3) - V(2,1)]^2 \quad (19.3-27d)$$

$$h_5 = [V(3,0) - 3V(1,2)][V(3,0) + V(1,2)] \\ + [V(3,0) + V(1,2)][V(0,3) + V(2,1)] \quad (19.3-27e)$$

$$h_6 = [V(2,0) - V(0,2)][V(3,0) + V(2,1)]^2 \\ + 4V(1,1)[V(3,0) + V(1,2)][V(0,3) + V(2,1)] \quad (19.3-27f)$$

$$h_7 = [3V(2,1) - V(0,3)][V(3,0) + V(1,2)] \\ + [3V(1,2) - V(3,0)][V(0,3) + V(2,1)] \quad (19.3-27g)$$

Table 19.3-4 lists the moment invariants of the test images. As desired, these moment invariants are in reasonably close agreement for the geometrically modified versions of the same object, but differ between objects. The relatively small degree of variability of the moment invariants for the same object is due to the spatial discretization of the objects.

Image	$h_1 \times 10^1$	$h_2 \times 10^3$	$h_3 \times 10^5$	$h_4 \times 10^8$	$h_5 \times 10^{10}$	$h_6 \times 10^{12}$	$h_7 \times 10^{14}$
Spade	1.920	4.387	0.715	0.295	0.123	0.185	-14.159
Rotated spade	1.919	4.371	0.704	0.270	0.097	0.162	-11.102
Heart	1.867	5.052	1.435	8.052	27.340	5.702	-15.483
Rotated heart	1.866	5.004	1.434	8.010	27.126	5.650	-14.788
Magnified heart	1.873	5.170	1.473	8.600	30.575	6.162	-0.559
Minified heart	1.863	4.887	1.443	8.019	27.241	5.583	0.658
Diamond	1.986	10.648	0.018	0.475	0.004	0.490	0.004
Rotated diamond	1.987	10.663	0.024	0.656	0.082	0.678	-0.020
Club	2.033	3.014	2.313	5.641	20.553	3.096	10.226
Rotated club	2.033	3.040	2.323	5.749	20.968	3.167	13.487
Ellipse	2.015	15.242	0.000	0.000	0.000	0.000	0.000

The spatial orientation of an object with respect to a horizontal reference axis is the basis of a set of orientation descriptors developed at the Stanford Research Institute (20). These descriptors, defined below, are described in Figure 19.4-1.

- Image oriented bounding box—the smallest rectangle oriented along the rows of the image that encompasses the object.
- Image oriented box height—dimension of box height for image oriented box.
- Image oriented box width—dimension of box width for image oriented box.
- Image oriented box area—area of image oriented bounding box.
- Image oriented box ratio—ratio of box area to enclosed area of an object for an image oriented box.
- Object oriented bounding box—the smallest rectangle oriented along the major axis of the object that encompasses the object.
- Object oriented box height—dimension of box height for object oriented box.
- Object oriented box width—dimension of box width for object oriented box.
- Object oriented box area—area of object oriented bounding box.
- Object oriented box ratio—ratio of box area to enclosed area of an object for an object oriented box.
- Minimum radius—the minimum distance between the centroid and a perimeter pixel.
- Maximum radius—the maximum distance between the centroid and a perimeter pixel.
- Minimum radius angle—the angle of the minimum radius vector with respect to the horizontal axis.
- Maximum radius angle—the angle of the maximum radius vector with respect of the horizontal axis.
- Radius ratio—ratio of minimum radius angle to maximum radius angle.

Table 19.4-1 lists the orientation descriptors of some of the playing card symbols.

TABLE 19.4-1. Shape orientation descriptors of the playing card symbols

Descriptor	Spade	Rotated heart	Rotated diamond	Rotated club
Row bounding box height	155	122	99	123
Row bounding box width	95	125	175	121
Row bounding box width	14,725	15,250	17,325	14,883
Row bounding box area	1,75	1,76	2,02	1,69
Object bounding box ratio	94	147	99	148
Object bounding box height	154	93	175	112
Object bounding box width	14,476	13,671	17,325	16,576
Object bounding box area	1,72	1,57	2,02	1,88
Minimum radius	1118	3828	38.95	26.00
Maximum radius	9205	8417	88.02	82.22
Minimum radius angle	-1.11	0.35	1.06	0.00
Maximum radius angle	-1.54	-0.76	0.02	0.85

19.5. FOURIER DESCRIPTORS

The perimeter of an arbitrary closed curve can be represented by its instantaneous curvature at each perimeter point. Consider the continuous closed curve drawn on the complex plane of Figure 19.5-1 in which a point on the perimeter is measured by its polar position $z(s)$ as a function of arc length s . The complex function $z(s)$ may be expressed in terms of its real part $x(s)$ and imaginary part $y(s)$ as

$$z(s) = x(s) + iy(s) \quad (19.5-1)$$

The tangent angle defined in Figure 19.5-1 is given by

$$\Phi(s) = \tan^{-1} \left[\frac{dy(s)/ds}{dx(s)/ds} \right] \quad (19.5-2)$$

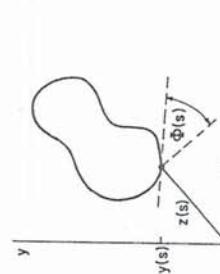


FIGURE 19.5-1. Geometry for curvature definition.

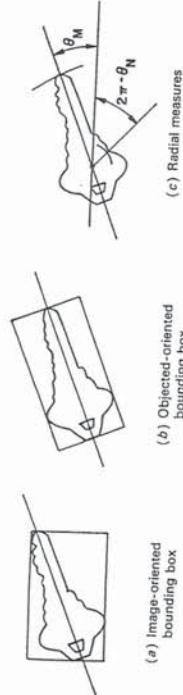


FIGURE 19.4-1. Example of shape orientation descriptors.

4 In book "Visual Information Retrieval" **Image retrieval A. de Bimbo by shape similarity**

Perceiving a shape is to capture prominent elements of an object. To humans, a few selected signs are not only sufficient for identification but also determine the impression of a complete and real representation of the object. On the other hand, computer vision research has provided many different solutions for shape representation and measurement of the difference of two shapes. For the purposes of retrieval by shape similarity, representations are preferred such that the salient perceptual aspects of a shape are captured and the human notion of closeness between shapes corresponds to the topological closeness in the representation space.

In this chapter, we discuss image retrieval by shape similarity. We summarize several methods for shape representation and retrieval by shape similarity.

4.1 SHAPE REPRESENTATION

Shape representation techniques fall in three broad categories: the feature-vector approach, the relational approach and the shape through transformation approach. The choice of a particular representation is usually driven by application needs, like the characteristics of the shapes being analysed, tolerance to occlusions and deformations, robustness against noise, possibility of indexing, etc.

Shape representation through a set of features, modelling prominent attributes of the shape, is the most popular technique. A shape is represented as a numerical vector and the difference between two shapes is evaluated according to a suitable distance. The Euclidean distance is the most used distance metric, but other distances such as the Hausdorff distance can be employed [1]. This method is widely employed in visual information retrieval and permits effective indexing.

According to the relational approach, shapes are broken down into a set of salient component parts. These are individually described through suitable features. The overall description includes both the descriptors of the individual parts and the relations between them. This approach is not commonly used in

Image retrieval by shape similarity

135

shape-based retrieval but, instead, is widely employed for the recognition of complex shapes. Relational distances, measuring the total differences between two descriptions, are typically employed to assess recognition.

Finally, shapes can also be distinguished by measuring the effort needed to transform one shape into another. In this case, similarity is measured as a transformational distance. Shape through transformation methods perform run-time evaluation of shape differences and do not support indexing.

Shape-based representations or follow the shape transformation techniques. Shape representation according to the feature-vector and the shape transformation approaches are briefly reviewed in the following paragraph. For a more complete discussion the reader should refer to [2-5].

4.1.1 Shapes through features

We distinguish feature-based representations by:

1. parametric internal methods: shape features describe the region enclosed by the object contour;
2. parametric external methods: shape features describe the external boundary of the object.

Parametric internal methods

Parametric internal methods use simple geometric attributes of the shape, digital moments, or the first coefficients of transforms such as the 2D Fourier or wavelets.

Use of simple geometric attributes

Description of the geometric properties of a region can be obtained by measuring properties of points belonging to the region. Simple geometric properties are, for example:

- the region area: measured as the count of internal pixels;
- the minimum/maximum rectangle, ellipse or circle which include or are included in the region: by considering, respectively, the ratio between sides, radii or the length of the radius;
- the compactness of the region: measured as the ratio between the squared perimeter and area;
- the elongatedness of the region: defined as the ratio between the length of the chord having the maximum length and the length of its perpendicular

Such simple descriptors only allow discrimination between shapes of very different forms.

A shape can be broken down into smaller regions of predefined form such as squares, circles, ellipses or star-shaped regions [6–12]. Attributes of these primitive components are used as features to model a shape. For example, if a shape is broken down into a set of circles, tangential to the shape boundary in at least two points, it can be described by taking the set of all radii and centres of these circles. Other representations use the set of maximal discs that fit inside the object, and the locus of the centres of these discs, also called symmetrical axis or the shape skeleton. Slight, perceptually insignificant differences between shapes may result in significant changes in these representations.

Based on digital moments

The first few moments are typically used as powerful shape descriptors. If we consider binary images, where region R (whose shape we want to describe) is the set of points so that $f(x, y) = 1$, then the digital i, j -th moment of R is represented by:

$$M_{ij}(R) = \sum_{(x,y) \in R} x^i y^j$$

$M_{0,0}(R)$ is the count of $\{x, y\} \in R$ and represents the area of R . The centroid of R is expressed as:

$$\bar{x} = \frac{M_{10}(R)}{M_{00}(R)} \quad \bar{y} = \frac{M_{01}(R)}{M_{00}(R)}$$

Powerful descriptors based on digital moments are functions of moments that are invariant under scaling, translation, rotation or squeezing.

Normalized moments $m_{ij}(R)$ [13], are invariant under translation, scaling, stretching and squeezing transformations. Normalized moments are defined by:

- considering the central i, j -th moment:

$$\mu_{ij}(R) = \sum_{(x,y) \in R} (x - \bar{x})^i (y - \bar{y})^j$$

- normalizing coordinates by their standard deviations $\sigma_x = \sqrt{(\mu_{20}/M_{00})}$ and $\sigma_y = \sqrt{(\mu_{02}/M_{00})}$:

$$x' = \frac{(x - \bar{x})}{\sigma_x} \quad y' = \frac{(y - \bar{y})}{\sigma_y}$$

$$m_{ij}(R) = \frac{\sum (x')^i (y')^j}{M_{00}} \quad (4.1)$$

Normalized central moments $\eta_{ij}(R)$ [14] are obtained from central moments according to the following transformation:

$$\eta_{ij}(R) = \frac{\mu_{ij}}{\mu_{00}^{\frac{j}{2}}} \quad (4.2)$$

$$\gamma = \frac{i+j}{2} + 1 \quad (4.3)$$

A set of functions are defined from these moments. Six of them are rotation invariant (ϕ_1 – ϕ_6), and one (ϕ_7) is both skew and rotation invariant:

$$\phi_1 = \eta_{20} + \eta_{02}$$

$$\phi_2 = (\eta_{20} + \eta_{02})^2 + 4\eta_{11}^2$$

$$\phi_3 = (\eta_{30} - 3\eta_{12})^2 + (3\eta_{21} - \eta_{03})^2$$

$$\phi_4 = (\eta_{30} + \eta_{12})^2 + (\eta_{21} + \eta_{03})^2$$

$$\begin{aligned} \phi_5 = & (\eta_{30} - 3\eta_{12})(\eta_{30} + \eta_{12})(3(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2) \\ & + (3\eta_{21} - \eta_{03})(\eta_{21} + \eta_{03})(3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2) \end{aligned}$$

$$\begin{aligned} \phi_6 = & (\eta_{20} - \eta_{02})(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2 \\ \phi_7 = & (3\eta_{21} - \eta_{03})(\eta_{30} + \eta_{12})(3(\eta_{30} + \eta_{12})^2 - 3(\eta_{21} + \eta_{03})^2) \\ & - (\eta_{30} - 3\eta_{12})(\eta_{21} + \eta_{03})(3(\eta_{30} + \eta_{12})^2 - (\eta_{21} + \eta_{03})^2) \end{aligned}$$

This representation can be employed to model simple forms [15–17]. In order to reduce the range of values assumed, the *log* of these functions can be used [18]. Shape representation using digital moments has been employed, for retrieval by shape similarity, in the QBIC system [19, 20] discussed in section 4.2.1.

Use of coefficients of 2D transforms

Object shapes can be described through some coefficients of their 2D discrete Fourier transform. The 2D discrete Fourier transform of a function $f(x, y)$ defined over the set $[0, 1, \dots, X-1] \times [0, 1, \dots, Y-1]$ is expressed as:

$$F(m, n) = \sum_{x=0}^{X-1} \sum_{y=0}^{Y-1} f(x, y) e^{-2\pi j(mx/X+ny/Y)} \quad (4.4)$$

With this representation, similarity is evaluated in the frequency domain. There is an advantage with respect to the spatial domain, in that translation is reflected only in a change of the $F(m, n)$ phase, while the $F(m, n)$ module remains constant. However, changes in scale and orientation of the object shape determine substantial changes in this representation.

Coefficients of the 2-D Discrete Wavelet Transform can also be used to represent a shape. Wavelets support a multi-scale representation which can be effective in highlighting discriminant signatures of the shape [21, 22].

Parametric external methods

Parametric-external methods can be based on either global or local boundary properties. Descriptions based on local properties are more insensitive to shape irregularities and are more suitable for handling the retrieval of partially occluded shapes, or shapes with overlapping and touching contours.

Pixel-based representations

Simple external parametric representations can be obtained by considering the set of pixels that compose the boundary of the curve. However, since the number of boundary points is usually too large, a smaller set is preferable. A good solution is to include only perceptually significant points of a shape contour. These points are called **salient** or **interest points** and consist of contour points where the curvature assumes singular values. For every point it is possible to retain:

- its coordinates $x(t)$ and $y(t)$, as a function of length of the curvilinear abscissa t along the curve;
- the direction $\phi(t)$ of the tangent, as a function of length of the curvilinear abscissa t along the curve;
- the length $r(\theta)$ of the line which connects the point with the origin of the reference system (θ being the angle between the line and one of the axes of the reference system);
- the curvature function $c(t)$, defined as the derivative of $\phi(t)$ with respect to the curvilinear abscissa t .

A simple representation is obtained by considering boundary points with **chain encoding**.

A grid is superimposed on the boundary in such a way that boundary points coincide with grid intersections (see Figure 4.1). The curve is approximated by linking points through horizontal, vertical or diagonal line segments. Each link direction corresponds to an integer number between 0 and 7. The curve is therefore represented as a string of numbers. The grid resolution determines the resolution of the boundary representation [23].

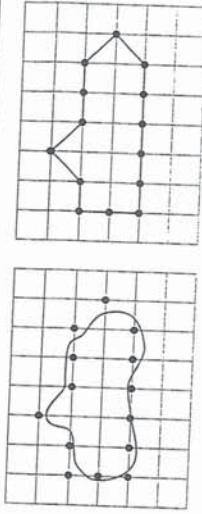


Figure 4.1 Chain encoding of a sample curve.

These and similar representations are usually very sensitive to boundary irregularities. Low-pass filtering must be applied to the original boundary to reduce the influence of irregularities on the quality of retrieval.

Use of boundary feature distributions

Stable representations of object shapes can be obtained, by considering the distribution of length r and orientation angle θ of all chords of the shape boundary. Chord distribution is expressed as:

$$h(r, \theta) = \int \int b(x, y)b(x + r \cos(\theta), y \sin(\theta))dx dy$$

where $b(x, y)$ is a function which is equal to 1 if (x, y) is on the shape boundary, otherwise 0. By integrating this expression with respect to r and θ , distribution of length $h_l(r)$ and orientation $h_o(\theta)$ are obtained:

$$h_l(r) = \int_0^{\pi} h(r, \theta)r d\theta \quad h_o(\theta) = \int_0^R h(r, \theta)dr$$

Distribution $h_l(r)$ is invariant to rotation and changes linearly with scaling. Distribution $h_o(\theta)$ is invariant to scaling and exhibits an offset, which is proportional to the rotation angle.

Representations that are invariant to scaling, rotation and translation are obtained by appropriately combining the two distributions.

Based on 1D Fourier transform

An effective representation of a curve boundary is obtained using 1D Fourier descriptors [24, 25]. The curve is represented as a function of the curvilinear abscissa t , and is expanded as a Fourier series. A finite number of coefficients of the series (typically the low-frequency coefficients) is employed

to describe the shape. Using rectangular coordinates $x(t)$ and $y(t)$, a point moving along the curve generates a complex function $r(u)$ which is periodic and therefore can be expanded as a Fourier series:

$$r(t) = x(t) + jy(t) = \sum_{k=-NC}^{NC} A(k) e^{j2\pi k t/L}$$

where L is the length of the curve. A translation of the curve determines a change only in the coefficient $A(0)$. Other transformations result in changes of $A(k)$ coefficients, according to:

$$A'(k) = S A(k) e^{j(\sigma + \rho k)}$$

where S is the change of scale, σ a rotation change and ρ is a change of the initial point taken on the curve (corresponding to $t = 0$). Such a representation has also been used for partially occluded shapes. Comparison of shapes using 1D Fourier descriptors has been discussed in [25]. In [26], it has been experimentally demonstrated that this representation provides better performance than autoregressive models (see below) for noisy images.

Based on autoregressive models

Autoregressive circular gaussian models (CGAR) have been used to represent boundaries of curves [27]. If a boundary is modelled as a single function, it can be expressed as:

$$u(k) = \alpha + \sum_{j=1}^m \phi_j u(k-j) + \beta w(k)$$

where α and β are constant values, ϕ_j are the autoregressive coefficients and $w(k)$ is a sequence of uncorrelated gaussian variables $\{w(1), \dots, w(N)\}$.

Use of boundary tokens

Methods that use local boundary properties break down the shape into a set of boundary segments, usually called tokens. The shape boundary is hence modelled as a sequence of tokens. Shapes are usually approximated through a simpler shape, such as a polygon with vertexes at the shape's salient points [7].

For each pair of segments, the coordinates of their intersection and their angle $\langle x_i, y_i, \alpha_i \rangle$ can be used as descriptors [28]. Techniques for resolving matching can be found in [5, 28, 29]. Visual information retrieval systems that use polygonalization for shape representation have been developed by Grosky and Mehrotra [30, 31], and Mehrotra and Gary [32, 33] (see section 4.2.1).

More sophisticated representations classify tokens according to their properties. The **codon-based representation** [34] is an example from this category. Codons are a set of primitive tokens which are used as basic descriptors of a generic curve. They are parts of a curve located between two consecutive minima of the curvature function. Therefore, between the extrema of each codon there is a maximum of the curvature function. Codons are classified into six different types, based on the analysis of their maximum and the number of zeroes of the curvature (Figure 4.2). A very complex shape can be described as a succession of codons. Codon-based symbolic descriptions of heart-wall shapes and their behaviour have been employed in a visual information retrieval system by Congiu *et al.* [35].

4.1.2 Shapes through transformations

The shape transformation model has been discussed in pattern recognition literature since the early work by Widrow [36], and successively developed in the well-known snake model by Kass, Witkin, and Terzopoulos [37]. According to the shape transformation model, a shape is regarded as a template and is deformed, in order to improve its match with a target image. A discrete set of parameters is used to model the deformation. The adaptation of these parameters is controlled through a cost function. The original shape is deformed until it finds a reasonable fitting with the target image.

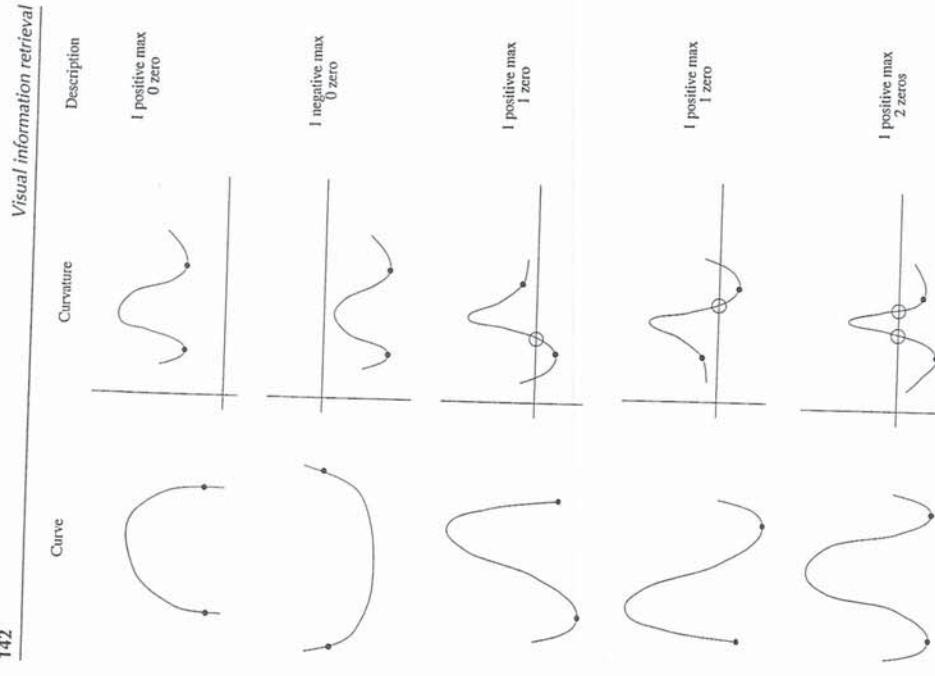
This model has been largely utilized to find correspondences between points of two different shapes [38, 39], or to classify shapes [40–42]. In image retrieval, this approach has been followed by Del Bimbo and Pala [43] and Scaroff and Pentland [39, 44], with different solutions. In [43], Del Bimbo and Pala have modelled the sketched template as a piecewise first-degree polynomial function, $\vec{r} = (r_x, r_y) : \mathbb{R} \mapsto \mathbb{R}^2$, parametrized with respect to arc length, and normalized. The deformation of the template is modelled as a function $\vec{\theta} = (\theta_x, \theta_y) : \mathbb{R} \mapsto \mathbb{R}^2$, so that the deformed template is expressed as:

$$\vec{\phi}(s) = \vec{r}(s)\vec{\theta}(s)$$

Deformation of the template is calculated by taking into account two opposite requirements:

1. The deformed template has to follow (as closely as possible) the edges of the image. The match between the deformed template and the edges in the image is measured as:

$$\mathcal{M} = \int_0^l [\nabla l[\vec{\phi}(s)]]^2 ds;$$

*Image retrieval by shape similarity*

143

2. The elastic deformation energy of the template has to be minimized. This energy is modelled as:

$$\mathcal{E} = \mathcal{S} + \mathcal{B}$$

$$\begin{aligned} \mathcal{E} &= \int_0^1 \left[\left(\frac{d\theta_x}{ds} \right)^2 + \left(\frac{d\theta_y}{ds} \right)^2 \right] + \beta \int_0^1 \left[\left(\frac{d^2\theta_x}{ds^2} \right)^2 + \left(\frac{d^2\theta_y}{ds^2} \right)^2 \right] ds \end{aligned}$$

The quantity \mathcal{S} depends on the first derivative and measures the stretching of the template $\vec{\tau}$ caused by the deformation $\vec{\theta}$. The quantity \mathcal{B} depends on the second derivative and is an approximate measure of the energy spent in bending the template.

An adaptation of the sketched template over the target image is obtained by minimizing the compound functional:

$$\mathcal{F} = \int_0^1 \mathcal{E} - M ds$$

In [39, 44], Pentland and Sclaroff have represented shapes as modal deformations of a prototype object [45]. Two objects are regarded as similar if they have undergone the same set of physical transformations. In order to build the shape finite element model, the mass and stiffness matrices are computed for each shape in the database. Positions of image feature points are used as finite element nodes. Both matrices are broken down into eigenvectors and eigenvalues. A modal matrix is defined as the ordered set (in ascending eigenvalue order) of mode vectors used to build the finite element model:

$$\Phi = [\phi_1 | \phi_2 | \dots | \phi_{2m}] = \begin{bmatrix} \vdots \\ \mathbf{u}_1^T \\ \vdots \\ \mathbf{v}_1^T \\ \vdots \\ \mathbf{u}_m^T \\ \vdots \\ \mathbf{v}_m^T \end{bmatrix} \quad (4.5)$$

This corresponds to a frequency-ordered shape description. Each mode vector ϕ_i describes the modal displacement of the i -th mode, at each feature point. Generalized feature vectors \mathbf{u} and \mathbf{v} describe the j -th feature displacement. Dimension m is the number of nodes used to build the model. Low-frequency modes describe global deformations that are due to rigid object movements. Intermediate-frequency modes describe local shape deformations. Highest-frequency modes are very sensitive to noise and are not useful for comparison.

Figure 4.2 Different types of codons.

Only p -intermediate mode vectors are actually used to compute correspondences between mode vectors. In this way comparison is performed between feature spaces of the same dimensionality.

Image retrieval with these solutions is discussed in detail in section 4.2.2.

4.1.3 Multi-scale shape descriptions

Behaviour of shapes at multiple scales is a very important signature to detect structural similarities between shapes [46]. Therefore, it can also be used effectively in shape retrieval [47–49].

If a shape is smoothed progressively, its details are filtered out while prominent features still remain. A good technique for analysing a shape at different resolutions is **scale space filtering** [46]. In this approach, the second derivative of the generic curve $f(t)$ is convoluted with a gaussian of given variance σ . Zeroes of this function correspond to the extrema of the $f'(t)$ curvature and represent points at which the curve is partitioned. Tokens are defined as those parts of the curve located between two consecutive partition points. As σ varies, the position and shape of these tokens change with continuity, and some partition points disappear (see Figure 4.3).

A pyramidal representation is obtained. Only boundary tokens that are stable at different resolutions are considered as representative of the structure of the shape.

4.2 SHAPE-BASED RETRIEVAL

Systems that employ **feature-based** representations treat shapes as feature vectors and measure similarity as a distance in the multidimensional feature space. They are particularly suited for large image databases. If the dimension of the feature vectors is not too large, vectors are organized in index structures (multidimensional point access methods – PAM) which filter the access to the database (see section 1.6.2). Given a query shape, the index is searched to find matching shape models, in a data-driven way.

Feature-based approaches tend to be brittle when shapes are a distorted version of the query template. For most features, there is no guarantee that the human notion of “closeness” corresponds to the topological closeness in the feature space. Shapes usually require the extraction of a great number of features in order to be reliably modelled. In order to perform indexing, this representation space must be reduced to a lower dimension.

Shape transformation approaches model shape similarity more closely to human perception. These methods are typically more robust in order to cope

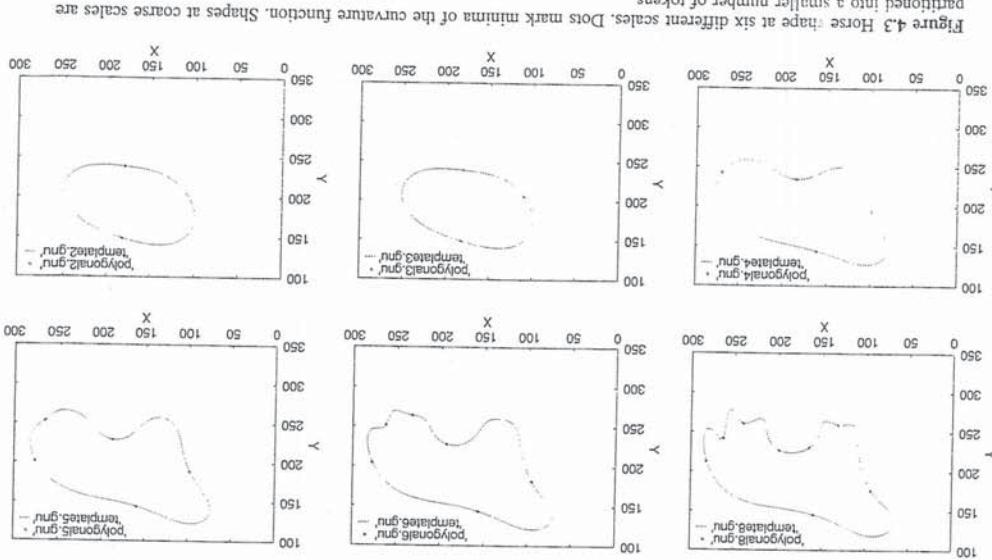


Figure 4.3 House shape at six different scales. Dots mark minima of the curvature function. Shapes at coarse scales are partitioned into a smaller number of tokens.

with shape distortions, support a higher precision in retrieving similar images and allow comparison with partially occluded shapes. However, indexing with these approaches is impossible. Therefore they are only suited to small-size image databases and run-time indexing. A convenient use is for search refinement after a pruning has been performed. Filtering modules (based on shape or other features such as colours, textures or spatial relationships) are used to discard uninteresting images before shape similarity is computed.

4.2.1 Feature-based methods

In feature-based retrieval methods, the similarity evaluation mechanism and indexing are dependent on the shape representation which is employed.

Hirata and Kato [50] have adopted the space of pixel values as the representation space. They do not use any multidimensional point access method. Instead, they follow the template matching approach to evaluate similarity between the query sketch and database edge images. However, with this technique it is possible to compare edge images of any complexity and not only closed contours of shapes.

Representations based on the closed boundary of the shape have been employed by several authors. Local boundary features modelling a part of the whole boundary have been employed by Grosky and Mehrotra [31], Mehrotra and Gary [32] and Jagadish [51]. Global boundary features capturing the global object appearance have been used in the QBIC system [19]. All these systems use multidimensional point access methods to speed up searching in the database. Califano and Mohan use hash tables in the main memory to index target shapes [52].

Image retrieval based on shape and other features like colour, texture and spatial relationships has been addressed by several authors, in [19, 49, 53–56], among others.

Based on edge image correlation

The Query by Visual Example (QVE) system by Hirata and Kato [50], performs retrieval by shape similarity, by computing the correlation between the query sketch and database edge images (see Figure 4.4).

In order to reduce storage size and computational costs, the RGB intensity value matrix is scaled up to 64×64 . Edge images are extracted by computing, for each pixel p_{ij} , the gradient ∂_{ij} in the four directions, and by considering those pixels where:

$$|\partial_{ij}| \geq \mu + \sigma$$

Edge images are thinned, scaled down and divided into 8×8 local blocks (they are referred to as abstract images).

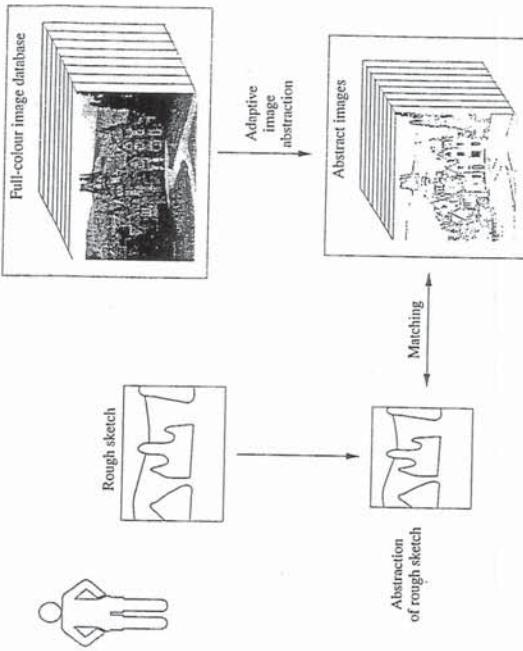


Figure 4.4 Sketch of QVE system.

The μ and σ are the average and deviation respectively of the gradient of the whole image and simulate the colour contrast sensitivity of the human eye. Edge images are subjected to a refinement procedure. For each local pixel, those pixels are considered where the gradient is greater than the sum of the local average and the local deviation of gradient values, in a 7×7 local window:

$$|\partial_{ij}| \geq \mu_{ij} + \sigma_{ij}$$

$$\mu_{ij} = \frac{1}{4(2m+1)(2n+1)} \sum_{r=i-m}^{i+m} \sum_{s=j-n}^{j+n} \sum_{k=1}^4 |\partial_{rs}|$$

$$\sigma_{ij} = \sqrt{\frac{1}{4(2m)(2n)} \sum_{r=i-m}^{i+m} \sum_{s=j-n}^{j+n} \sum_{k=1}^4 |\partial_{rs}|^2 - \mu_{ij}^2} \quad (4.6)$$

Edge images are thinned, scaled down and divided into 8×8 local blocks (they are referred to as abstract images).

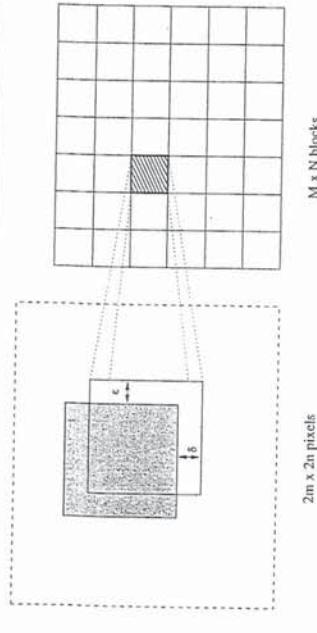


Figure 4.5 Local correlation shifting in [50].

In order to cope with shifts and distortions between the database images and the user's sketch, the local correlation between the two blocks (of the abstract image and the user's sketch) is the maximum value among the five correlation values $C_{\delta\epsilon}^{ab}$ between the local blocks P^{ab} of the abstract image and the local blocks Q^{ab} of the sketch, shifted by (δ, ϵ) . Each local block is shifted in four directions. A 4-pixel shifting is used, essentially to keep the cost of computation low (see Figure 4.5):

$$C^{ab} = \max(C_{\delta\epsilon}^{ab}) \quad \text{for } -m/2 \leq \delta \leq m/2, -n/2 \leq \epsilon \leq n/2 \quad (4.7)$$

$$C_{\delta\epsilon}^{ab} = \sum_{r=m}^{m(\alpha+1)\beta(h+1)} \sum_{s=nh}^7 (\alpha p_{rs} \odot q_{r+\delta s+\epsilon} \beta \overline{p_{rs}} \odot \overline{q_{r+\delta s+\epsilon}} + \gamma p_{rs} \oplus q_{r+\delta s+\epsilon}) \quad (4.8)$$

The global image correlation is the sum of the local correlation values:

$$C = \sum_{\alpha=0}^7 \sum_{h=0}^7 C^{ab} \quad (4.9)$$

This approach does not allow indexing. To filter out uninteresting images, Hirata and Kato suggest that images be clustered in the database with respect to a set of basic templates. Basic templates are, for example, horizontal or vertical lines (to select images that have prevalent horizontal or vertical directionality), but could also take into account other characteristic attributes.

Use of local boundary features

Grosky and Merhota have approximated shapes through a polygonal curve [30, 31, 57]. For each vertex v , a local feature is defined as:

$$(A_v, D_v, X_v, Y_v) \quad (4.10)$$

where A_v is the internal angle at the vertex v , D_v is the distance from the adjacent vertex, clockwise, and (X_v, Y_v) are the vertex coordinates. A fixed number of these local features is extracted from each shape. To improve insensitivity to noise, they suggest taking into account the three sharpest internal angles and the successive angles in ascending order.

A shape is thus represented by an attributed string. Similarity between two shapes is computed as the editing distance between the two strings of the boundary features (i.e., the number of changes required to transform one string into the other). Features of database images are organized in an index tree. A binary tree is used to access main memory. Access to secondary memory is accomplished through a m -way tree. The algorithm used to derive the binary tree index organization is based on the following steps:

1. Find a reference feature for the collection of imaged objects. This is the root of the binary index tree. Reference features may be features from the feature collection or artificial features. The distinguishing property of such feature is that the sum of values of the similarity measure with the other features must be minimal, or alternatively must have the minimal maximum value of the similarity between this feature and the others in the collection.
2. List the features in the collection according to their similarity with the root and split the set in the middle.
3. For each subcollection, find the reference feature according to the same procedure as in step 1.
4. Move features from one subcollection to another so that members of any subtree are more similar to the reference feature of the subtree than to the reference features of other subtrees.
5. Follow iteratively steps 2, 3, 4, until the size of a collection becomes equal to 1.

In this index, leaf nodes always hold true object features. They also hold a list of all the objects and their locations in the images. Intermediate node labels may identify object features but, more frequently, they identify artificial features. Small deviations of feature values may impair this approach. If features at the root of two subtrees at the same level have a very small difference, and a slightly different version of one of them is compared with both, the wrong path can be chosen leading to incorrect retrieval. To cope with this fact, Grosky and Merhota allow ranges for each feature value in the index. When the database is populated, if a feature has a similarity difference below

$$(A_v, D_v, X_v, Y_v) \quad (4.10)$$

where A_v is the internal angle at the vertex v , D_v is the distance from the adjacent vertex, clockwise, and (X_v, Y_v) are the vertex coordinates. A fixed number of these local features is extracted from each shape. To improve insensitivity to noise, they suggest taking into account the three sharpest internal angles and the successive angles in ascending order.

A shape is thus represented by an attributed string. Similarity between two shapes is computed as the editing distance between the two strings of the boundary features (i.e., the number of changes required to transform one string into the other). Features of database images are organized in an index tree. A binary tree is used to access main memory. Access to secondary memory is accomplished through a m -way tree. The algorithm used to derive the binary tree index organization is based on the following steps:

1. Find a reference feature for the collection of imaged objects. This is the root of the binary index tree. Reference features may be features from the feature collection or artificial features. The distinguishing property of such feature is that the sum of values of the similarity measure with the other features must be minimal, or alternatively must have the minimal maximum value of the similarity between this feature and the others in the collection.
2. List the features in the collection according to their similarity with the root and split the set in the middle.
3. For each subcollection, find the reference feature according to the same procedure as in step 1.
4. Move features from one subcollection to another so that members of any subtree are more similar to the reference feature of the subtree than to the reference features of other subtrees.
5. Follow iteratively steps 2, 3, 4, until the size of a collection becomes equal to 1.

In this index, leaf nodes always hold true object features. They also hold a list of all the objects and their locations in the images. Intermediate node labels may identify object features but, more frequently, they identify artificial features. Small deviations of feature values may impair this approach. If features at the root of two subtrees at the same level have a very small difference, and a slightly different version of one of them is compared with both, the wrong path can be chosen leading to incorrect retrieval. To cope with this fact, Grosky and Merhota allow ranges for each feature value in the index. When the database is populated, if a feature has a similarity difference below

a predefined threshold (*distortion allowance*) with features in two subtrees at the same level, the feature is duplicated in the two subtrees. In this case, multiple tree branches include the same feature. This results in a larger index tree and more computational effort in spanning the index at deletion and retrieval time. An interesting point of this approach is that, with this system, a boundary feature modelling only a part of the boundary can be used in the query instead of the complete object boundary.

Mehrotra and Gary have developed a retrieval technique called FIBSSR (Feature Index-Based Similar-Shape Retrieval) [32, 33], based on similar ideas. Like [31], they use a polygonal approximation of the shape boundary. In this case, each boundary feature models a segment of the shape boundary and is defined as a finite number n of adjacent vertices of the polygon boundary. The shape boundary is thus represented as an ordered finite collection of boundary features. In order to obtain invariance to scale rotation and translation, a couple of adjacent vertices are selected as a basic vector. All the other vertices belonging to the same feature are transformed in the new space (see Figure 4.6).

To derive a complete description, every pair of adjacent vertices of the polygonal approximation is selected as a basis vector. Each feature is therefore defined by the set of points in the new normalized space and by a transformation vector which holds the scale, translation and angle of the basis vector. Feature vectors are organized in a k - d - B tree. Leaves of the tree contain a list indicating in which shapes the feature stored at the leaf appears. This approach permits the use of a single boundary feature, modelling a part of the boundary, instead of the complete object boundary, in the query.

The choice of features of the query influences the index performance. Less sensitive features are those with a long distance between adjacent points and sharp angles. A degree of tolerance on feature matching can be declared at

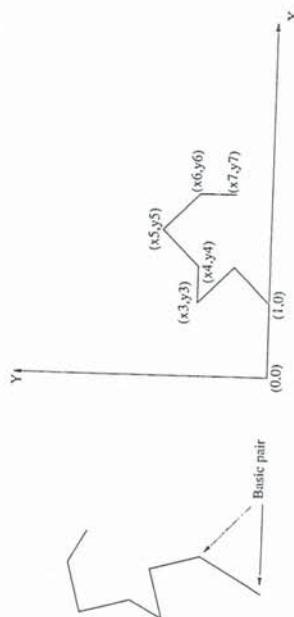


Figure 4.6 Shape boundary features used in [32].

querying time. The matching of one or more features does not guarantee full shape matching. Once shapes with similar features are retrieved, shape similarity is checked by overlaying each retrieved shape on the query shape and evaluating the amount of overlap between the two. This overlay is carried out by the *simplex method*.

Use of global features

In the QBIC system, retrieval by shape similarity is supported in two ways [19]. On the one hand, sketch-based retrieval is provided according to the method proposed by Hirata and Kato [50]. On the other hand, retrieval is performed by comparing a set of global features representing the query shape with features modelling database shapes. In this latter approach, all shapes are represented as binary images. A set of 22 features is used for their representation:

- area, computed as the number of pixels, set in the binary image.
- circularity, expressed as the ratio between the squared perimeter and the area.
- eccentricity, computed as the ratio between the smallest and the largest eigenvalues.
- major-axis orientation, derived as the direction of the largest eigenvector of the second order covariance matrix.
- a set of algebraic moment invariants, computed from the first central moments up to degree 8. They are derived as the eigenvalues of predefined matrices: $(M_{[2,2]}, M_{[2,3]} \times M_{[3,2]}, M_{[3,3]}, M_{[3,4]} \times M_{[4,3]}, M_{[4,4]}, M_{[5,4]})$, whose elements are scaled factors of the central moments.

In order to permit indexing, this high-dimensional feature vector is reduced to a lower-dimensionality vector, by using the distance-preserving Karhunen-Loeve transform. Transformed vectors retain most of the information in the first few coefficients. An R^* tree is used as the indexing method for these low-dimensional features. Shape similarity is checked by computing the weighted Euclidean distance between the transformed feature vectors of the query and the target shapes. Feature weights are the inverse variances of the features. In QBIC, the quality of retrieval is highly dependent on the nature of the template. Generally speaking, it is effective when considering global shape properties (for example, roundness or squareness of a bottle shape). But, it is not suited to evaluating similarity according to local properties [43].

Use of multiple scale representation

A few authors have considered the possibility of using scale-space representations of shape boundaries for image retrieval.

Among them, Mokhtarian, Abbasi and Kittler [48] have used curvature scale space images of the curve to represent the boundary at multiple resolutions. Every curvature scale space contour represents a concavity or convexity of the original boundary. Every object is represented by the locations of the maxima of its curvature scale space image. Matching of curvature scale space images is obtained by shifting curvature scale space contours so that the major maximum of one image covers the maximum of the other [58]. Del Binbo and Pala have used a three-layered representation of the boundary for each scale [49]. At the lowest level, a numeric description of each boundary token in which the curve is partitioned by the curvature minima is stored. Each token is described by its orientation, its symmetry and its relative length with respect to the overall shape boundary. At higher levels, descriptions are symbolic and retain for each token a bucketed value of its orientation, symmetry and length (at the second level) and orientation only (at the highest level). In this system, an index is used to group shapes which share similar tokens at a certain scale. The index is organized as a graph.

Faloutsos uses a shape representation based on mathematical morphology which is inherently multiscale [59]. He uses size-distribution, related to the pattern spectrum to extract shape features. Features extracted are organized into a *R*-Tree index.

4.2.2 Shape transformation based methods

Examples of shape transformation based solutions for retrieval are Elastic Template Matching by Del Binbo and Pala [43, 49, 60] and Modal Matching, in the Photobook system, by Sclaroff and Pentland [39].

Elastic template matching

In this approach, the query sketch is warped in order to adapt to shapes in the database images [41, 61, 62]. Measures of the elastic deformation and the degree of matching achieved are used to evaluate similarity rankings of database images. Similarity matching is regarded as a minimization problem. An adaptation of the sketched template over the target images is obtained by minimizing the compound functional (see section 4.1.2):

$$\mathcal{F} = \int_0^1 \varepsilon - \mathcal{M} ds \quad (4.11)$$

A numerical solution for the deformation $\vec{\theta}$ in Eq. 4.11 is provided by piecewise fourth order splines. Experiments carried out show that 20 knots

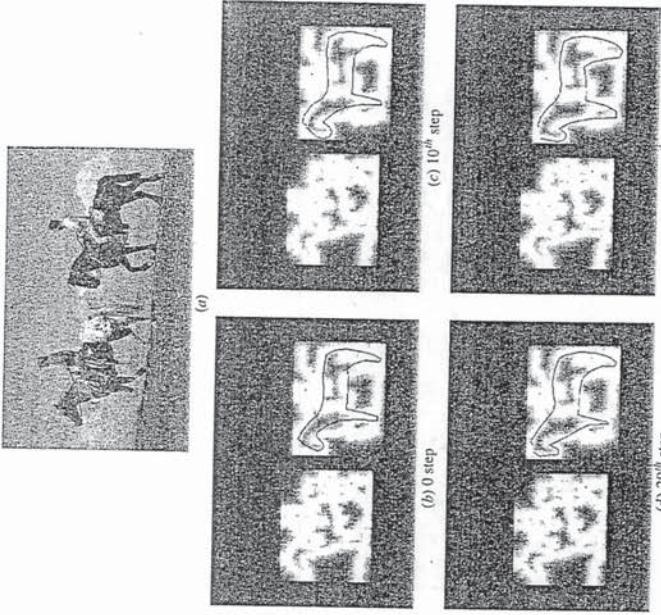


Figure 4.6 Iterative matching of a horse-like template over an edge image: (a) the original image; (b) the original template superimposed to the edge image; (c), (d), (e) different steps of the deformation process of the original template [43]. © 1997 IEEE.

are sufficient to model the warping of quite complex shapes, such as the one reported in Figure 4.7. If $B_i(s)$ are the fourth order B-splines, the deformation $\vec{\theta}$ can be written as:

$$\vec{\theta}(\vec{C}, s) = \sum_{i=1}^N \vec{c}^i B_i(s) \quad (4.12)$$

where

$$\vec{C} = (\vec{c}^1, \dots, \vec{c}^N) \quad \text{with} \quad \vec{c}^i = (c_x^i, c_y^i)$$



Figure 4.7 Elastic deformation of a horse-like template over an edge image: (a) the original image; (b) the original template superimposed to the edge image; (c), (d), (e) different steps of the deformation process of the original template [43]. © 1997 IEEE.

The minimization of the functional $\mathcal{F}(\tilde{\mathbf{C}})$ (obtained by substituting (4.12) in (4.11)), with respect to the variables $(c_x^1, \dots, c_x^N, c_y^1, \dots, c_y^N)$ is achieved through a gradient descending technique [63]. The higher α and β are, the less the template can be deformed. The values of α and β are kept low initially and are increased during the deformation process.

Shape similarity is computed through a back-propagation neural network. The network uses as an input the three values of matching, stretching and bending, and two additional parameters: a qualitative measure of template deformation, calculated as the correlation between the curvature of the original template and that of the deformed one; and a qualitative measure of the complexity of the template, measured through the number of zeroes of the shape curvature function.

Elastic matching is expensive from a computational point of view and does not allow indexing. It is therefore suited to small databases, or in being employed as a refinement tool. On the other hand it provides similarity ranking very close to human ranking. Figure 4.8 shows an example of retrieval obtained with this system. This technique is inherently translation invariant, but it is not rotation invariant, since rotation contributes to the elastic energy. However,

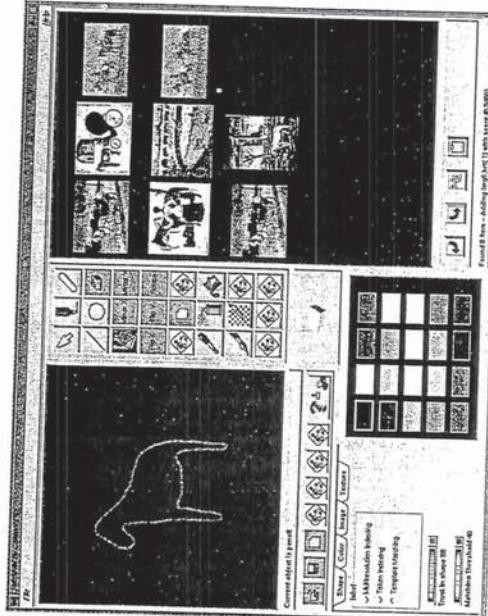


Figure 4.8 Retrieval results with the elastic matching method (Picasso system) [43].

small rotations, of the order of 12–15 degrees, are coped with by the elastic matching algorithm. Comparison of Elastic Template Matching with QBIC and QVE is reported in [43].

Modal matching

Sclaroff and Pentland, in the Photobook system, have represented shapes as modal deformations of a prototype object [45]. Once modal matrices are computed (see section 4.1.2), affinities between intermediate mode vectors of modal matrices are analysed by building an affinity matrix:

$$z_{ij} = \|\bar{u}_{Q,i} - \bar{u}_{D,j}\|^2 + \|\bar{v}_{Q,i} - \bar{v}_{D,j}\|^2.$$

The affinity matrix is used to determine the corresponding mode features in the query and target shapes. These are the minimum entries in the matrix. Normally, one-to-many correspondences exist between features. Modal vibrations are used to identify the amount of deformation (both rigid and non-rigid) needed to align the source object to the target. Mode amplitudes indicate the amount of energy required to align two objects and thus provide a similarity measure.

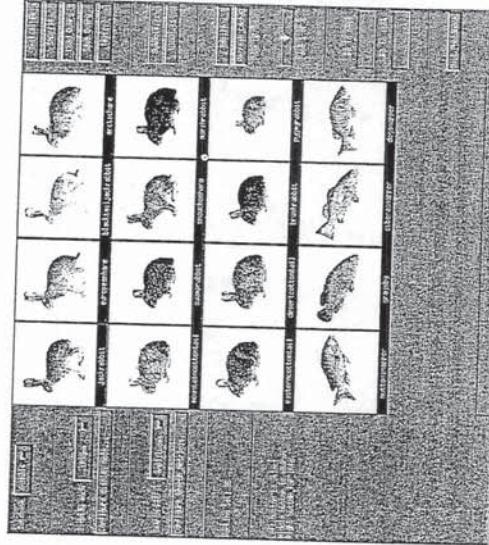


Figure 4.9 Retrieval by shape using modal matching: the user selected the jackrabbit (shown in upper left corner) [45] – courtesy of S. Sclaroff.

Deformation parameters can be evaluated in a number of ways. A simple method is to solve, in closed form, an equilibrium problem with $2m$ equations in the $2m$ unknown modal displacements. Since it is required that strain energy is minimized, an iterative technique is needed to obtain non-zero modal displacements.

Shape silhouettes are extracted and thinned down to a maximum of 120 points per contour. Mode amplitudes are evaluated for the first 22 modes of each prototype object. Figure 4.9 shows an example of retrieval by shape using this approach.

4.3 REFERENCES

- [1] Huttenlocher, D.P., G.A. Klanderman and W.J. Rucklidge. 1993. Comparing images using the Hausdorff distance. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15(9): 850–863.
- [2] Duda, R. and P. Hart. 1973. *Pattern Classification and Scene Analysis*. New York: J. Wiley & Sons.
- [3] Pavlidis, T. 1977. *Structural Pattern Recognition*. New York: Springer.
- [4] Ballard, D. and C. Brown. 1982. *Computer Vision*. Englewood Cliffs, NJ: Prentice-Hall.
- [5] Haralick, R.M. and L.G. Shapiro. 1992. *Computer and Robot Vision, Vol. I-II*. Reading MA, USA: Addison Wesley.
- [6] Montanari, U. 1969. Continuous skeletons from digitized images. *Journal of the ACM*, 4: 534–549.
- [7] Montanari, U. 1970. A note on minimal length polygonal approximation to a digitized contour. *Communications of the ACM*, 13(January): 41–47.
- [8] Arcelli, C., L.P. Cordella and S. Levialdi. 1981. From local maxima to connected skeletons. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 3: 134–143.
- [9] Blum, H. and R. Nagel. 1978. Shape description using weighted symmetric axis features. *Pattern Recognition*, 10: 167–180.
- [10] Shapiro, L.G. 1979. A structural model of shape. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 1: 10–20.
- [11] Shapiro, L.G. and R. Haralick. 1979. Decomposition of two-dimensional shapes by graph theoretic clustering. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 1: 10–20.
- [12] Avis, D. and G.T. Toussaint. 1981. An efficient algorithm for decomposing a polygon into star-shaped polygons. *Pattern Recognition*, 13: 395–398.
- [13] Alt, F.L. 1962. Digital pattern recognition by moments. *Journal of the ACM*, 11: 240–258.
- [14] Hu, M.K. 1962. Visual pattern recognition by moment invariants. *JRE Transactions on Information Theory*, 8: 351–364.
- [15] Teh, C.H. and R.T. Chin. 1986. On digital approximation of moment invariants. *Computer Vision, Graphics and Image Processing*, 33: 318–326.
- [16] Teh, C.H. and R.T. Chin. 1988. On image analysis by the methods of moments. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 10: 496–513.
- [17] Khotanzad, A. and Y.H. Hong. 1988. Rotation invariant pattern recognition using Zernike moments. *Proc. 9th Int. Conf. on Pattern Recognition*, Rome, pp. 326–328.
- [18] Hsia, T.C. 1981. A Note on Invariant Moments in Image Processing. *IEEE Trans. on Systems, Man and Cybernetics*, 11: 831–834.
- [19] Niblack, W., R. Barber, W. Equitz, M. Flickner, E. Glasman, D. Petkovic and P. Yanker. 1993. The QBIC project: querying images by content using colour, texture and shape. *Proc. SPIE Int. Symp. on Electronic Imaging: Science & Technology*, Vol. 1908, *Storage and Retrieval of Image and Video Databases*, February.
- [20] Niblack, W. et al. 1993. *The QBIC project: querying images by content using colour, texture and shape*. Res. Report 9203, IBM Res. Div., Almaden Res. Center, February.
- [21] Daubechies, I. 1988. Orthonormal bases of compactly supported wavelets. *Communications on Pure and Applied Mathematics*, XL1: 999–996.
- [22] Chui, C.K. 1992. *An Introduction to Wavelets*. San Diego, CA: Academic Press.
- [23] Freeman, H. 1974. Computer processing of line drawing images. *Computer Surveys*, 6: 57–97.
- [24] Granlund, G.H. 1972. Fourier preprocessing for hand print character recognition. *IEEE Trans. on Computers*, 21: 195–201.
- [25] Pearson, E. and K.S. Fu. 1977. Shape discrimination using Fourier descriptors. *IEEE Transactions on Systems, Man and Cybernetics*, 7: 170–179.
- [26] Kauppinen, H., T. Seppänen and M. Pietikäinen. 1995. An experimental comparison of autoregressive and Fourier-based descriptors in 2D shape classification. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 17(2): 201–207.
- [27] Kayshap, R.L. and R. Chellappa. 1981. Stochastic models for closed boundary analysis: representation and reconstruction. *IEEE Trans. on Information Theory*, 27(5): 627–637.
- [28] Davis, L.S. Shape matching using relaxation techniques. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 1: 60–72.
- [29] Bunke, H. and A. Sanfeliu. 1990. *Syntactic and Structural Pattern Recognition: Theory and Applications*. Singapore: World Scientific.
- [30] Grossky, W.I., P. Neo and R. Mehrotra. 1989. A pictorial index mechanism for model-based matching. *Proc. 5th Int. Conf. on Data Engineering*, Los Angeles, CA, February, pp. 180–187.
- [31] Grossky, W.I. and R. Mehrotra. 1990. Index-based object recognition in pictorial data management. *Computer Vision, Graphics and Image Processing*, 52: 416–436.
- [32] Mehrotra, R. and J.E. Gary. 1995. Similar-shape retrieval in shape data management. *IEEE Computer*, (September): 57–62.

- [33] Mehrotra, R. and W.I. Grosky. 1989. Shape matching utilizing indexed hypothesis generation and testing. *IEEE Trans. on Robotics and Automation*, 5(1): 70–77.
- [34] Richards, W. and D.D. Hoffman. 1984. *Coden Constraint on Closed 2D Shapes*. Computer Vision II, Natural Computation Group, MIT, Cambridge.
- [35] Congiu, G., A. Del Bimbo and E. Vicario. 1995. Iconic retrieval by content from databases of cardiological sequences. In *Visual Database Systems III*, S. Saccapreti and R. Jain (eds), Chapman & Hall.
- [36] Widrow, B. 1973. The rubber mask technique-II – pattern storage and recognition. *Pattern Recognition*, 5: 199–211.
- [37] Kass, M., A. Witkin and D. Terzopoulos. 1988. Snakes: active contour models. *International Journal of Computer Vision*, 3(6): 321–331.
- [38] Burr, D.J. 1981. A dynamic model for image registration. *Computer Graphics and Image Processing*, 15(2): 102–112.
- [39] Sclaroff, S. and A. Pentland. 1993. A modal framework for correspondence and recognition. *Proc. ICCV*, May.
- [40] Burr, D.J. 1981. Elastic matching of line drawing. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 3(6): 708–713.
- [41] Youille, A.L., K. Honda and C. Peterson. 1991. Particle tracking by deformable templates. *Proc. Int. Joint Conf. on Neural Networks*, pp. 7–11.
- [42] Youille, A.L., P.W. Hallinan and D.S. Cohen. 1992. Feature extraction from faces using deformable templates. *Int. Journal of Computer Vision*, 8(2): 99–111.
- [43] Del Bimbo, A. and P. Pala. 1997. Image retrieval by elastic matching of user sketches. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 19(2): 121–132.
- [44] Sclaroff, S. 1997. Deformable prototypes for encoding shape categories in image databases. *Pattern Recognition*, 30(4): 627–642.
- [45] Pentland, A. and S. Sclaroff. 1991. Closed-form solutions for physically-based shape modelling and recognition. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 13(7): 715–729.
- [46] Witkin, A. 1983. Scale space filtering. *Proc. 7th Int. Conf. Artificial Intelligence*, pp. 1010–1021.
- [47] Mokhtarian, F. and A.K. Mackworth. 1992. A theory of multiscale, curvature-based shape representation for planar curves. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 14: 789–805.
- [48] Mokhtarian, F., S. Abbasi and J. Kittler. 1996. Efficient and robust retrieval by shape content through curvature scale space. *Proc. First Int. Workshop on Image Databases and Multi-Media Search*, Amsterdam, The Netherlands, August, pp. 35–42.
- [49] Del Bimbo, A. and P. Pala. 1996. Shape indexing by multi-scale representation. *Proc. First Int. Workshop on Image Databases and Multimedia Search*, Amsterdam, The Netherlands, August, pp. 35–42.
- [50] Hirata, K. and T. Kao. 1992. Query by Visual Example: Content-Based Image Retrieval. In *Advances in Database Technology – EDBT'92*, Pirotte, A., C. Delobel and G. Gottlob (eds), *Lecture Notes on Computer Science*, 580: 56–71.
- [51] Jagadish, H.V. 1991. A retrieval technique for similar shapes. *Proc. ACM SIGMOD*, New York, pp. 208–217.
- [52] Califano, A. and R. Mohan. 1994. Multidimensional indexing for recognizing visual shapes. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 16(4): 373–392.
- [53] Dubuisson, M.P. and A.K. Jain. 1994. Fusing colour and edge information for object matching. *Proc. First Int. Conf. on Image Processing*, Austin TX, November.
- [54] Jain, A.K. and A. Vallaya. 1994. Image retrieval using colour and shape. *Pattern Recognition*, 29(8): 1233–1244.
- [55] Pentland, A., R. Picard and S. Sclaroff. 1994. Photobook: tools for content based manipulation of image databases. *Proc. of the SPIE: Storage and Retrieval for Image and Video Databases II*, San Jose, CA, vol. 2185, February, pp. 34–47.
- [56] Liang, K.C. and C.C.J. Kuo. 1997. Progressive image indexing and retrieval based on embedded wavelet coding. *Proc. ICIP'97, Int. Conf. on Image Processing*, Santa Barbara, CA, October, pp. 1572–1575.
- [57] Grossky, W.I., P. Neo and R. Mehrotra. 1992. A pictorial index mechanism for model-based matching. *Data and Knowledge Engineering*, 8: 309–327.
- [58] Mokhtarian, F., S. Abbasi and J. Kittler. 1996. Robust and efficient shape indexing through curvature scale space. *Proc. BMVC96*, Edinburgh, Scotland, September.
- [59] Faloutsos, C. 1998. Efficient and effective nearest neighbor search in a medical image database of tumor shapes. In *Image Description and Retrieval*, Vicario E. (ed.), Plenum Press.
- [60] Del Bimbo, A., P. Pala and S. Santini. 1994. Visual image retrieval by elastic deformation of object shapes. *Proc. IEEE VL'94, Int. Symp. on Visual Languages*, St Louis MI, October.
- [61] Durbin, R. and D. Willshaw. 1987. An analogue approach to the travelling salesman problem using an elastic net method. *Nature*, 326, April.
- [62] Ohlsson, M. 1992. *Extensions and Explorations of the Elastic Arms Algorithm*. Technical Report LU TP 92-28, Department of Theoretical Physics, University of Lund, Sjövegatan 14A S-22362 Lund, Sweden, December. Also submitted to *Computer Physics Communications*.
- [63] Anseline, P.M. and P.J. Laurent. 1968. A general method for the construction of interpolating or smoothing spline-functions. *Numerische Mathematik*, 12: 66–82.

Further reading

- [a] Basri, R. and D. Weinshall. 1992. *Distance metric between 3D models and 2D images for recognition and classification*. AI Memo 1373, AI Laboratory, MIT, July.
- [b] Cohen, L.D. and I. Cohen. 1993. Finite element methods for active contour models and balloons for 2D and 3D images. *IEEE Trans. on Pattern Analysis and Machine Intelligence*, 15(11): 1131–1137.