Chapter 6

Color and Shading

The perception of color is very important for humans. Color perception depends upon both the physics of the light and complex processing by the eye-brain which integrates properties of the stimulus with experience. Humans use color information to distinguish objects, materials, food, places, and even the time of day. Figure 6.1 shows the same scene coded with different colors: even though all shapes are the same, the right image is quite different from the left and the viewer might interpret it as an indoor scene of a housecat rather than a tiger in grass.

With recent innovation in economical devices, color processing by machine has become commonplace: we have color cameras, color displays and software that processes color images. Color can also be used by machines for the same purposes humans use it. Color is especially convenient because it provides multiple measurements at a single pixel of the image, often enabling classification to be done without complex spatial decision-making.

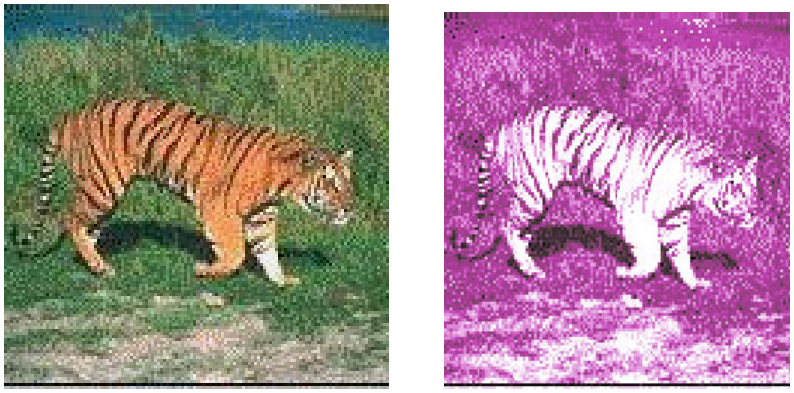


Figure 6.1: (Left) Naturally colored image of tiger in grass; (right) with transformed colors, recognition of a tiger is less secure – perhaps it's a cat on a rug?

Careful study of the physics and perception of color would require many pages: here we provide only a foundation that should be sufficient for beginning programming using color or as a guide to the literature. Some basic principles of the physics of color are given along with practical methods of coding color information in images. Then, we give some examples

and methods for using color in recognizing objects and segmenting images.

We also study the *shading* of objects, which depends not just on the color of objects and the light illuminating them, but also on many other factors. These factors include the roughness of surfaces, the angles made between the surface and both the light sources and the viewer, and the distances of the surface from both the light sources and the viewer. Color and shading, important elements of art for centuries, are also important for interpreting a scene by computer vision algorithms.

6.1 Some Physics of Color

Electromagnetic radiation with wavelength λ in the range of between about 400 and 700 nanometers stimulates human neurosensors and produces the sensation of color. (Figure 6.2). A nanometer is 10^{-9} meter: it is also referred to as a millimicron. For blue light, $400 \times 10^{-9} meters\ per\ wave$ means $2.5 \times 10^{6} waves\ per\ meter$ or 25000 $waves\ per\ cm$. The speed of light in a vacuum is $3 \times 10^{8} m/sec$, which is equivalent to a frequency of 0.75×10^{15} blue light waves per second. This frequency is one one thousandth of that for X-rays and one billion times that of broadcast radio waves.

For the rest of this chapter, we refer to wavelength or frequency only in the context of the qualitative color it produces. Machines can detect radiation well beyond the range of human neurosensors; for example, short ultraviolet waves and extremely short X-rays can be detected by special devices. Also, long infrared waves can be detected by many solid state cameras, and very long radio waves can be detected by a radio receiver. Science and engineering have developed many devices to sense and transduce pixel measurements into the visible spectrum: the X-ray machine and IR satellite weather scanner are two common examples.

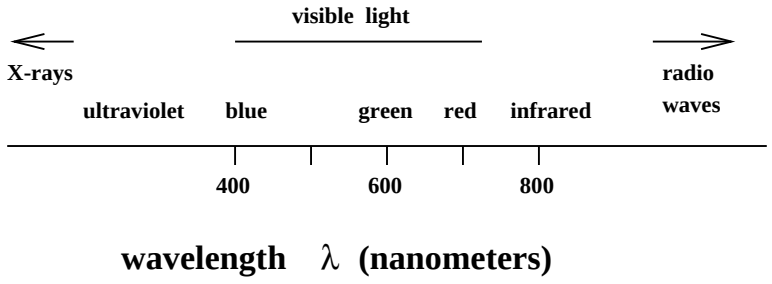


Figure 6.2: Visible part of the electromagnetic spectrum.

Exercise 1

Suppose a piece of paper is 0.004 inches thick. What is its thickness in terms of the equivalent number of waves of blue light?

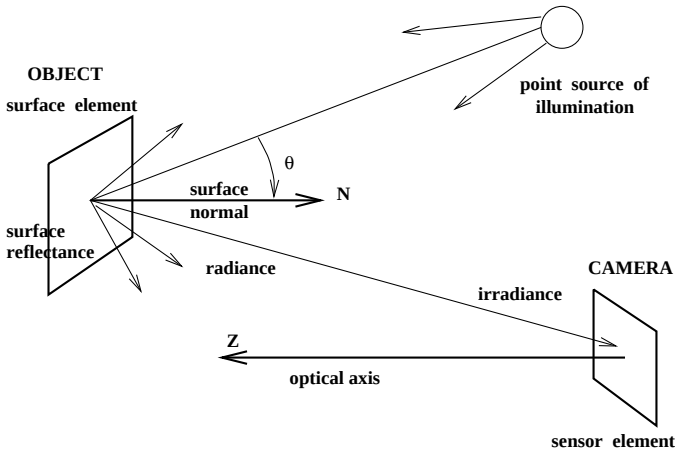


Figure 6.3: Light energy from a source reflects from an object surface and irradiates a sensor element.

6.1.1 Sensing Illuminated Objects

Figure 6.3 shows light from a point source illuminating an object surface. As a result of the illuminating energy interacting with molecules of the object surface, light energy, or radiance, is emitted from the surface, some of which irradiates, or stimulates, a sensor element in a camera or organism's eye. The sensation, or perception, of an object's color depends upon three general factors:

- the spectrum of energy in various wavelengths illuminating the object surface,
- the spectral reflectance of the object surface, which determines how the surface changes the received spectrum into the radiated spectrum,
- the spectral sensitivity of the sensor irradiated by the light energy from the object's surface.

An object that is "blue" has a surface material that appears blue when illuminated with *white light*.

1 Definition **White light** *is composed of approximately equal energy in all wavelengths of the visible spectrum.*

This same object should appear violet if illuminated by only red light. A blue car under intense (white) sunlight will become hot to the touch and radiate energy in the IR range, which cannot be seen by the human eye but can be seen by an IR camera.

6.1.2 Additional Factors

In addition to the three major factors given above, there are several complicating factors in both physics and human perception. Surfaces vary in *specularity*, that is, how much they act

like a mirror. *Matte* surfaces reflect energy equally in all directions. The energy or intensity of radiation depends upon distance – surface elements farther from a point source of white light will receive less energy than closer surface elements. The effect is similar between the radiating object and the sensor elements. As a result, image intensities received from the same surface material might be nonuniform due to the nonuniform distances along the imaging rays. The orientation θ of the surface element relative to the source is even more important than distance in determining the energy reflected toward the sensor. These issues are discussed in more detail toward the end of this chapter.

Exercise 2 variation of intensity with distance

Point your computer's camera perpendicularly at a sheet of uniform white paper that is illuminated from an incandescent bulb off to one side. Record the image and study the image intensities. How much variation is there? Is there a systematic decrease of intensity as the distance from some brightest pixel increases?

Exercise 3 variation of intensity with surface normal

Repeat the above experiment using a spherical volleyball rather than a flat sheet of paper. Record the image and study the image intensities. Report on the variations and regularities.

6.1.3 Sensitivity of Receptors

Actual receptors react only to some wavelengths and are more sensitive to certain wavelengths than to others. Figure 6.4 shows sample sensitivity curves. Three of the curves correspond to three different kinds of cones in the human eye containing different chemical pigments sensitive to different wavelengths. The curve marked $human_1$ corresponds to a type of cone that is mildly sensitive to *blue* light between 400 and 500 nm. The curve marked $human_2$ corresponds to cones that are very sensitive to green light and mildly sensitive to shorter wavelengths of blue and longer wavelengths of red. The brain fuses the responses from a local neighborhood of several cones to produce the perception of any visible color. It is somewhat remarkable that only three kinds of receptors are needed to do this, even though there are an infinite number of possible wavelengths of light. Many other seeing animals have only one or two types of receptors and perhaps perceive less rich color as a result. Solid state sensing elements usually have good sensitivity above the range for humans. It's important to remember this, since sometimes as the workday warms up, a machine vision system will see a scene differently from what a human operator sees. This is primarily due to the different sensitivity to IR radiation.

Exercise 4 favorite color

Do you have a *favorite color*? Is so, what is it? Why is it your favorite? Ask 3 other people what their favorite color is. Assuming you have multiple answers, how can you explain it given the known physics of color?

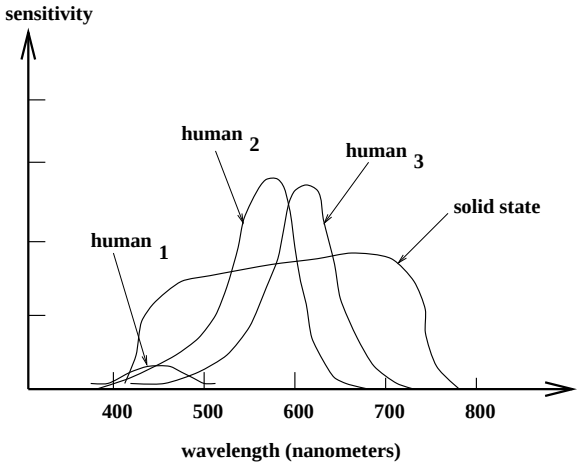


Figure 6.4: Comparison of relative sensitivities of 3 human pigments in cones and solid state sensor element.

6.2 The RGB Basis for Color

Using only three types of receptors, humans can distinguish among thousands of colors; a more exact number is subject to argument. The *trichromatic* RGB (red-green-blue) encoding in graphics systems usually uses three bytes enabling $(2^8)^3$ or roughly 16 million distinct color codes. To be precise, we say 16 million *codes* and not 16 million *colors* because humans cannot actually perceive that many distinct colors. Machines can distinguish between any pair of different bit encodings, but the encodings may or may not represent differences that are significant in the real world. Each 3-byte or 24-bit RGB pixel includes one byte for each of red, green, and blue. The order in which each appears in memory can vary; order is irrelevant to theory but important for programming. Display devices whose color resolution matches the human eye are said to use *true color*. At least 16 bits are needed: a 15-bit encoding might use 5 bits for each of R,B,G, while a 16-bit encoding would better model the relatively larger green sensitivity using 6 bits.

The encoding of an arbitrary color in the visible spectrum can be made by combining the encoding of three *primary colors* (RGB) as shown in Figure 6.5. Red:(255,0,0) and green:(0,255,0) combined in equal amounts create yellow:(255,255,0). The amount of each primary color gives its intensity. If all components are of highest intensity, then the color white results. Equal proportions of less intensity create shades of grey:(c,c,c) for any constant $0 < c < 255$ down to black:(0,0,0). It is often more convenient to scale values in the range 0 to 1 rather than 0 to 255 when making decisions about color in our algorithms: use of such a range is *device-independent*.

The RGB system is an *additive color system* because colors are created by adding components to black:(0,0,0). This corresponds well to RGB displays (monitors) which have three types of phosphors to emit light. Three neighboring elements of phosphor corresponding to a pixel are struck by three electron beams of intensity c_1, c_2 and c_3 respectively: the human

	RGB	CMY	HSI
RED	(255, 0, 0)	(0,255,255)	(0.0 , 1.0, 255)
YELLOW	(255,255, 0)	(0, 0,255)	(1.05, 1.0, 255)
	(100,100, 50)	(155,155,205)	(1.05, 0.5, 100)
GREEN	(0,255, 0)	(255, 0,255)	(2.09, 1.0, 255)
BLUE	(0, 0,255)	(255,255, 0)	(4.19, 1.0, 255)
WHITE	(255,255,255)	(0, 0, 0)	(-1.0, 0.0, 255)
GREY	(192,192,192)	(63, 63, 63)	(-1.0, 0.0, 192)
	(127,127,127)	(128,128,128)	(-1.0, 0.0, 127)
	(63, 63, 63)	(192,192,192)	(-1.0, 0.0, 63)
	...		
BLACK	(0, 0, 0)	(255,255,255)	(-1.0, 0.0, 0)

Figure 6.5: Different digital trichromatic color encoding systems. It is often more convenient to scale values in the range 0 to 1 when making decisions in algorithms. HSI values are computed from RGB values using Algorithm 1: $H \in [0.0, 2\pi)$, $S \in [0.0, 1.0]$ and $I \in [0, 255]$. Byte codings exist for H and S.

eye integrates their luminance to perceive "color":(c_1, c_2, c_3). The light of 3 wavelengths from a small region of the CRT screen is thus physically added or mixed together.

Suppose that a color sensor encodes a pixel of a digital image as (R, G, B), where each coordinate is in the range $[0, 255]$, for example. The computations shown in Equation 6.1 are one way to normalize image data for interpretation by both computer programs and people and for transformation to other color systems as discussed below. Imagine a color camera staring at a scene with variations in illumination; for example, object surface points are at varying distances from illumination sources and may even be in shadow relative to some of the light sources. An algorithm to aggregate green pixels corresponding to the image of a car would perform poorly unless the normalization for intensity were done first.

$$\begin{aligned} intensity\ I &= (R + G + B)/3 \\ normalized\ red\ r &= R/(R + G + B) \\ normalized\ green\ g &= G/(R + G + B) \\ normalized\ blue\ b &= B/(R + G + B) \end{aligned} \tag{6.1}$$

Using the normalization of Equation 6.1, the normalized values will always sum to 1. There are alternative normalizations; for instance, we could use $max(R, G, B)$ as the divisor rather than the average RBG value. By using $r + g + b = 1$, the relationship of coordinate values to colors can be conveniently plotted via a 2D graph as in Figure 6.6. Pure colors

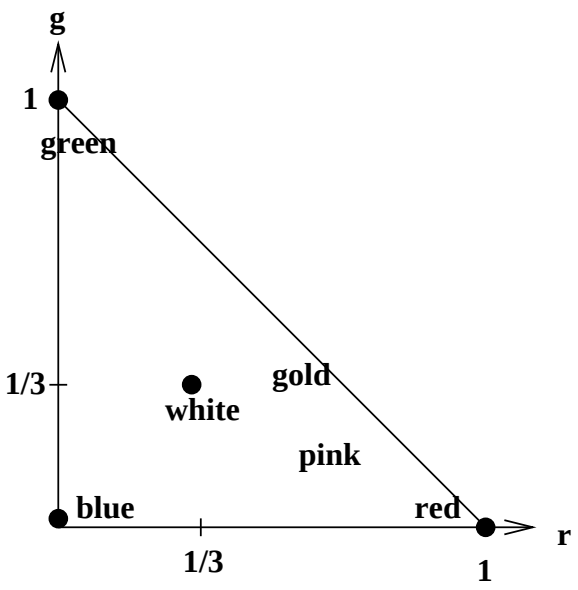


Figure 6.6: Color triangle for normalized RGB coordinates. The blue ('b') axis is out of the page perpendicular to the 'r' and 'g' axes. Thus, the triangle is actually a slice through the points [1,0,0], [0,1,0] and [0,0,1] in 3D. The value for blue can be computed as $b = 1 - r - g$ for any pair of r-g values shown in the triangle.

are represented by points near the corners of the triangle. For example, a "fire-engine-red" will be near the lower right corner with coordinates (1,0) and a "grass-green" will be at the top with coordinates (0,1) while "white" will be at the centroid (1/3,1/3). In Figure 6.6, the blue ('b') axis is out of the page perpendicular to the 'r' and 'g' axes, and thus the triangle is actually a slice through the points [1,0,0], [0,1,0] and [0,0,1] in 3D. The value for blue can be computed as $b = 1 - r - g$ for any pair of r-g values shown inside the triangle.

Exercise 5 experimenting with color codes

Aquire an RGB color image and view it with some image tool. Exchange the green and blue bytes and report on the results. Double all and only the low blue values and report on the results.

6.3 Other Color Bases

Several other color bases exist which have special advantages relative to devices that produce color or relative to human perception. Some bases are merely linear transformations of others and some are not.

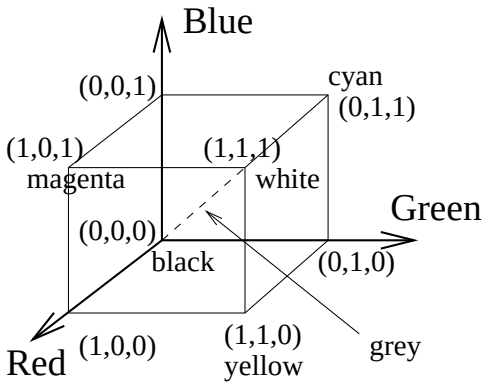


Figure 6.7: Color cube for normalized RGB coordinates: the triangle in Figure 6.6 is a projection of the plane through points [1, 0, 0], [0, 1, 0], and [0, 0, 1].

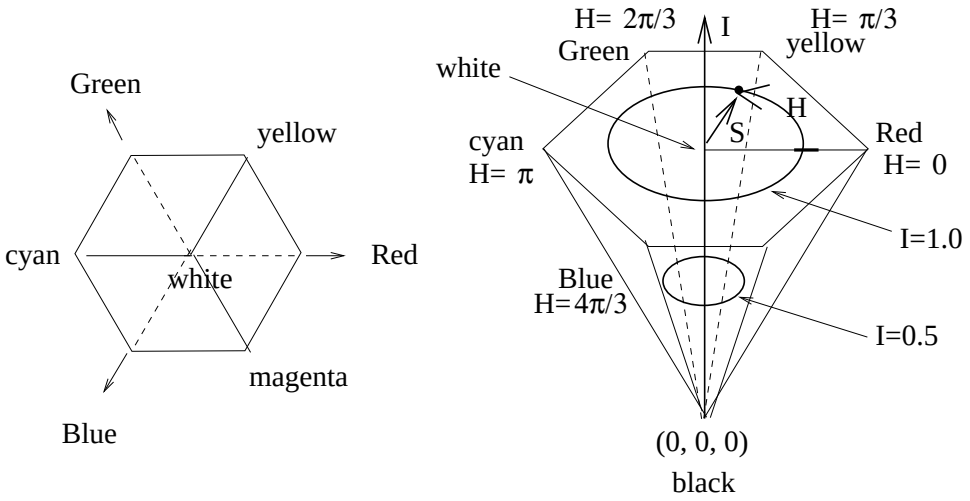


Figure 6.8: Color hexacone for HSI representation. At the left is a projection of the RGB cube perpendicular to the diagonal from (0, 0, 0) to (1, 1, 1): color names now appear at the vertices of a hexagon. At the right is a hexacone representing colors in HSI coordinates: intensity (I) is the vertical axis; hue (H) is an angle from 0 to 2π with *RED* at 0.0; saturation (S) ranges from 0 to 1 according to how pure, or unlike white, the color is with S=0.0 corresponding to the **I**-axis.

6.3.1 The CMY Subtractive Color System

The CMY color system models printing on white paper and *subtracts from white* rather than adds to black as the RGB system does. CMY coding is shown next to RGB in Figure 6.5. CMY is an abbreviation of *Cyan-Magenta-Yellow*, which are its three primary colors corresponding to three inks. Cyan absorbs red illumination, magenta absorbs green and yellow absorbs blue, thus creating appropriate reflections when the printed image is illuminated with white light. The system is termed *subtractive* because of the encoding for absorption. Some trichromatic encodings are as follows; white:(0,0,0) because no white illumination should be absorbed, black:(255,255,255) because all components of white light should be absorbed and yellow:(0,0,255) because the blue component of incident white light should be absorbed by the inks, leaving the red and green components to create the perception of yellow.

6.3.2 HSI: Hue-Saturation-Intensity

The HSI system encodes color information by separating out an overall intensity value I from two values encoding *chromaticity* — hue H and saturation S. The color cube in Figure 6.7 is related to the RGB triangle shown in Figure 6.6. In the cube representation, each r, g, b value can range independently in $[0.0, 1.0]$. If we project the color cube along its major diagonal, we arrive at the hexagon at the left of Figure 6.8. In this representation, shades of grey that were formerly along the color cube diagonal now are all projected to the center "white" point while the "red" point [1,0,0] is now at the right corner and the green point [0,1,0] is at the top left corner of the hexagon. A related 3D representation, called a "hexacone", is shown at the right in Figure 6.8: the 3D representation allows us to visualize the former cube diagonal as a vertical intensity axis **I**. Hue **H** is defined by an *angle* between 0 and 2π relative to the "red"-axis, with pure "red" at an angle of 0, pure "green" at $2\pi/3$ and pure "blue" at $4\pi/3$. Saturation **S** is the 3rd coordinate value needed in order to completely specify a point in this color space. Saturation models the purity of the color or hue, with 1 modeling a completely pure or saturated color and 0 modeling a completely unsaturated hue, i.e. some shade of "grey".

The HSI system is sometimes referred to as the "HSV" system using the term "value" instead of "intensity". HSI is more convenient to some graphics designers because it provides direct control of brightness and hue. Pastels are centered near the **I** axis, while deep or rich colors are out at the periphery of the hexacone. HSI might also provide better support for computer vision algorithms because it can normalize for lighting and focus on the two chromaticity parameters that are more associated with the intrinsic character of a surface rather than the source that is lighting it.

Derivation of HSI coordinates from RGB coordinates is given in Algorithm 1. The algorithm can convert input values (r, g, b) from the 3D color cube, or those normalized by Equation 6.1, or even byte-coded RGB values as in the left column of Figure 6.5. Intensity I is returned in the same range as the input values. Saturation S is not defined when intensity $I = 0$ and hue H is not defined when $S = 0$. H is in the range $[0,\ 2\pi)$. Whereas one might use a square root and inverse cosine to define mathematical conversion formulas, Algorithm 1 uses very simple computational operations so that it will run fast when converting an entire image of pixels from one encoding to another. Samples of the output of

Algorithm 1 are given at the right in Figure 6.5.

Conversion of RGB encoding to HSI encoding.

```
R,G,B : input values of RGB all in range [0,1] or [0,255];
I : output value of intensity in same range as input;
S : output value of saturation in range [0,1];
H : output value of hue in range [0,2π), -1 if S is 0;
R,G,B,H,S,I are all floating point numbers;

    procedure RGB_to_HSI( in R,G,B; out H,S,I)
    {
    I := max ( R, G, B );
    min := min ( R, G, B );
    if (I ≥ 0.0) then S := (I - min )/I else S := 0.0;
    if (S ≤ 0.0) then { H := -1.0; return; }
       "compute the hue based on the relative sizes of the RGB components"
    diff := I - min;
    "is the point within +/- 60 degrees of the red axis?"
    if (r = I) then H := (π/3)*(g - b)/diff;
    "is the point within +/- 60 degrees of the green axis?"
    else if (g = I) then H := (2 * π/3) + π/3 *(b - r)/diff;
    "is the point within +/- 60 degrees of the blue axis?"
    else if (b = I) then H := (4 * π/3) + π/3 *(r - g)/diff;
    if (H ≤ 0.0) H := H + 2π;
    }
```

Algorithm 1: Conversion of RGB to HSI.

Exercise 6

Using Algorithm 1, (a) convert the RGB code (100,150,200) into an HSI code and (b) convert the rgb code (0.0, 1.0, 0.0) to HSI.

Returning to Figure 6.6, we see how HSI values relate to the color triangle. Hue is related to the dominant wavelength of the light and corresponds approximately to a point on the sides of the triangle in Figure 6.6 with the lower values of λ near 400 nm starting at the origin and increasing along the $g - axis$ to about 520 nm and further increasing toward 800 nm down along the hypotenuse. Hue corresponds to the angle from the centroid corresponding to "white" toward some point (r, g) on a side of the triangle. The H and S values for 50% saturated gold is midway between the points marked "white" and "gold" in Figure 6.6. Figure 6.6 is an approximation to the *painters color palette*.

Figure 6.9 shows the transformation of an image by changing its saturation. The original input image is at the left. The center image is the result of decreasing the saturation S of all individual pixels by 20% and the right image is the result of a 20% increase in S. Relative

Figure 6.9: (Left) Input RGB image; (center) saturation S increased by 40%; (right) saturation S decreased by 20%. (Photo by Frank Biocca.)

to our experience, colors in the center image look washed out while those in the right image appear overdone. It is important to note that hue H is unchanged in the three images and should thus be a reliable feature for color segmentation despite variations in intensity of white light under which a machine vision system might have to operate.

Exercise 7

Develop an algorithm to convert r,g,b color coordinates in $[0, 1]$ to H,S,I using the following approach based on analytical geometry. Construct a perpendicular from point $[r, g, b]$ to the color cube diagonal through $[0, 0, 0]$ to $[1, 1, 1]$ and compute H,S,I accordingly.

6.3.3 YIQ and YUV for TV signals

The NTSC television standard is an encoding that uses one luminance value Y and two chromaticity values I and Q; only luminance is used by black and white TVs, while all three are used by color TVs. An approximate linear transformation from RGB to YIQ is given in Equation 6.2. In practice, the Y value is encoded using more bits than used for the values of I and Q because the human visual system is more sensitive to luminance (intensity) than to the chromaticity values.

$$\begin{aligned} luminance \;\; Y &= 0.30R + 0.59G + 0.11B \\ R - cyan \;\; I &= 0.60R - 0.28G - 0.32B \\ magenta - green \;\; Q &= 0.21R - 0.52G + 0.31B \end{aligned} \tag{6.2}$$

YUV encoding is used in some digital video products and compression algorithms such as JPEG and MPEG. The conversion of RGB to YUV is as follows.

$$\begin{aligned} Y &= 0.30R + 0.59G + 0.11B \\ U &= 0.493 * (B - Y) \\ V &= 0.877 * (R - Y) \end{aligned} \tag{6.3}$$

YIQ and YUV have better potential for compression of digital images and video than do other color encoding schemes, because luminance and chrominance can be coded using different numbers of bits, which is not possible with RGB.

Exercise 8 color code conversion

Suppose a color camera encodes a given pixel in RGB as (200,50,100), where 255 is the highest (most energy) value. (a) What should be the equivalent triple in the HSI system? (b) What should be the equivalent triple in the YIQ system?

Exercise 9

Is the transformation from RGB to YIQ invertible? If so, compute the inverse.

Exercise 10 recoding images

Assuming that you have a display and software to view an RGB image, perform the following experiment. First, create an HSI image such that the upper right quarter is saturated red, the lower left corner is saturated yellow, the upper left quarter is 50% saturated blue and the lower right quarter is 50% saturated green. Invert the RGB to HSI conversion of Algorithm 1 and convert the HSI image to RGB. Display the image and study the colors in the 4 image quarters.

6.3.4 Using Color for Classification

The color of a pixel contains good information for classifying that pixel in many applications. In Section 6.5 a color model for human skin color is described that goes a long way toward finding a human face in a color image. Confusion is possible, however. For example, pixels from a brown cardboard box can pass the skin color test and region shape might be needed to distinguish a polyhedral box face from an elipsoidal human face. Figure 6.10 shows the result of extracting "white regions" of an image by passing pixels that are close to some sample pixel from training. Sample pixels were obtained from the symbols on the sign. Several unwanted regions are also formed by other white objects and by specular reflections. Character recognition algorithms could recognize many of the characters and discard most of the unwanted components.

In general, interpretation of the color of an individual pixel is error prone. The image at the left in Figure 6.9 was taken with a flash from the camera and some of the faces of the pineapple chunks appear white because of specular reflection (described below in Section 6.6.3). A classifier that broadens the definition of yellow to include these white pixels is also likely to include pixels from specular reflections off a blue cup, for instance. Interpetation problems occur in particular regions of the color space: when saturation is close to zero computation and interpretation of hue is unreliable and when intensity is low interpretation of saturation is also unreliable.

Figure 6.10: "White pixels" are segmented from the color image at the left. Individual connected components of white pixels are arbitrarily labeled by a coloring algorithm as decribed in Chapter 3. (Analysis contributed by David Moore.)

Exercise 11

Show that conversion from RGB to HSI is unstable when either saturation or intensity is close to 0 by performing the following experiments. Implement Algorithm 1 as a program. (a) Convert the RGB codes $(L + \delta L_R, L + \delta L_G, L + \delta L_B)$ to HSI for L large and $\delta L_X \in$ {-2, -1, 1, 2}. Are the values of H consistent? (b) Repeat this experiment for L small (about 10) and δ as above. Are the values for S consistent?

6.4 Color Histograms

A histogram of a color image can be a useful representation of that image for the purpose of image retrieval or object recognition. A histogram counts the number of pixels of each kind and can be rapidly created by reading each image pixel just once and incrementing the appropriate bin of the histogram. Retrieval of images from image databases using color histograms is treated in Chapter 8. Color histograms are relatively invariant to translation, rotation about the imaging axis, small off-axis rotations, scale changes and partial occlusion. Here, we sketch the method of color histogram matching originally proposed by Swain and Ballard (1991) for use in object recognition.

A simple and coarse method of creating a histogram to represent a color image is to concatenate the higher order two bits of each RGB color code. The histogram will have $2^6 = 64$ bins. It is also possible to compute three separate histograms, one for each color, and just concatenate them into one. For example, separate RGB histograms quantized into 16 levels would yield an overall $k = 48$ bin histogram as used by Jain and Vailaya (1996). Two color images and histograms derived from them are shown in Figure 6.4.

The *intersection* of image histogram $h(I)$ and model histogram $h(M)$ is defined as the sum of the minimum over all K corresponding bins as denoted in Equation 6.4. The intersection value is normalized by dividing by the number of pixels of the model to get a match value. This match value is a measure of how much color content of the model is present

Figure 6.11: Color images and their 64-bin histograms (obtained from A. Vailaya).

in the image and is not diminished due to background pixels of the image that are not in the model. Other similarity measures can be defined; for example, we could normalize the histograms into frequencies by dividing each bin count by the total number of pixels and then use the Euclidean distance to compare two images.

$$\begin{aligned} intersection(h(I), h(M)) &= \sum_{j=1}^{K} min\{h(I)[j], h(M)[j]\} \\ match(h(I), h(M)) &= \frac{\sum_{j=1}^{K} min\{h(I)[j], h(M)[j]\}}{\sum_{j=1}^{K} h(M)[j]} \end{aligned} \tag{6.4}$$

Experiments have shown that histogram match values can be good indicators of image similarity under variations mentioned above and also using different spatial quantizations of the image. Swain and Ballard also developed a *backprojection* algorithm which could locate a region in an image approximately the size of the model object that best matched the model histogram. Thus, they have developed two color-based algorithms, one to recognize if an image contains a known object and another to determine where that object is located. If images are taken under different lighting conditions, then intensity should be factored out first. One should also consider smoothing the histograms so that a good match will still be obtained with minor shifts of the reflectance spectrum. An alternate method is to match the cumulative distributions rather than the frequencies themselves.

Exercise 12 matching images with permuted pixels

Suppose that given image A, we create image B by randomly permuting the locations of the pixels in A. (This could be done as follows. First, copy A to B. Then, for each pixel I[r, c] in B, choose pixel I[x, y] at random and swap I[r, c] with I[x, y].) What would be the resulting match value between the histograms from A and B?

Exercise 13 recognizing produce

Obtain 3 bananas, 3 oranges, 3 red apples, 3 green apples, 3 green peppers and 3 red tomatoes. For each of these six sets of produce, take three images, varying the arrangement of the 3 different produce items each time. This will result in 18 images. Construct a color histogram for each image. Use the first histogram of each set to be the model (6 models in all), then compute the histogram match value between each of the 6 models and the 12 other histograms. Report the results: do your results support the possibility of a supermarket produce recognition system to recognize produce placed on the cashier's scale?

6.5 Color Segmentation

We now describe work on finding a face in a color image taken from a workstation camera. The ultimate goal of the work is better man-machine communication. An algorithm is sketched which finds the main region of the image corresponding to the user's face. First, a training phase is undertaken to determine the nature of face pixels using samples from different people. Figure 6.12 shows a plot of pixels (r, g) taken from different images containing faces: normalized red and green values are used as computed from Equations 6.1. Six classes of pixels are easy to define by decision boundaries using the methods of Chapter 4; three

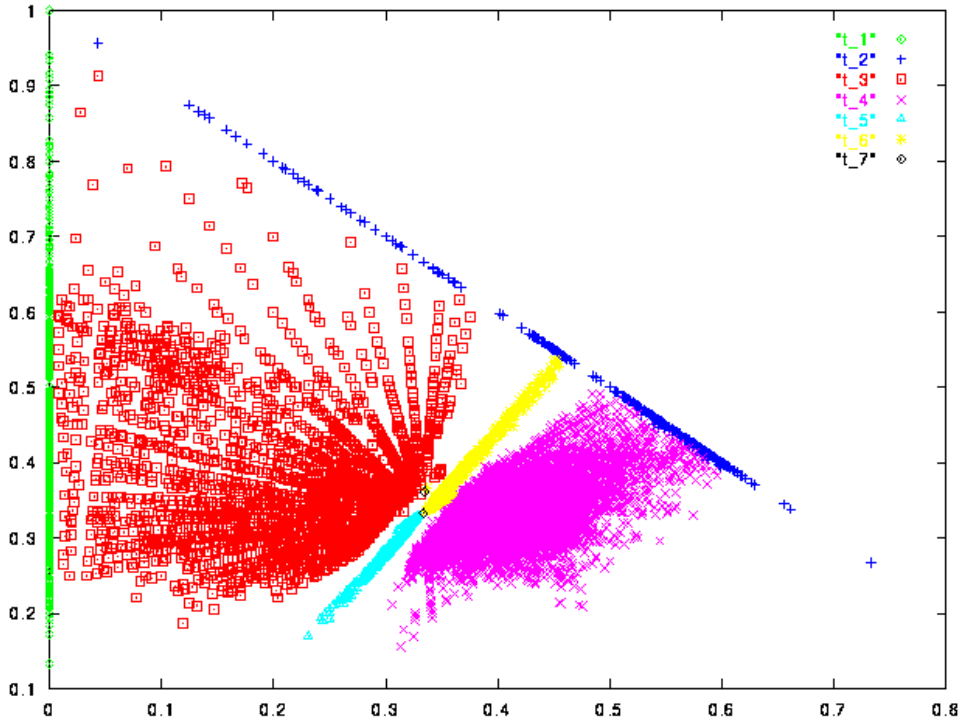


Figure 6.12: Skin color clusters obtained from training: the horizontal axis is R_{norm} and the vertical axis is G_{norm}. The cluster labeled as `t_4` is the primary face color, clusters `t_5` and `t_6` are secondary face clusters associated with shadowed or bearded areas of a face. (Figure from V. Bakic.)

of them contain face pixels, a primary class and two classes of pixels from shadows or beards.

Three major steps are used to identify the face region. The input to the first step is a labeled image using the labels 1, 2,...,7 which result from classification according to the training data (label 7 is used for a pixel that is not in any of the other six classes). The middle images of Figure 6.13 show labeled images for two different faces: most of the face pixels are correctly labeled as are most of the background pixels; however, there are many small areas of error. Components are aggregated and then merged or deleted according to their size and location relative to the major face region. First, connected component processing is done as described in Chapter 3 with only pixels labeled 4, 5, or 6 as foreground pixels. The second step selects as the face object the largest suitable component. This step also discards components that are too small or too big using heuristics learned from processing many examples. Typically, less than 100 components remain, the majority of which are in the shadow classes. The third step discards remaining components or merges them with the selected face object. Several heuristics using knowledge of the human face are applied; also, it is assumed that there is only one face in the scene. Example results are shown at the right in Figure 6.13. The program is fast enough to do these computations roughly 30 times per second (real-time), including computing the locations of the eyes and nose, which has not been described. This example generalizes to many other problems. A

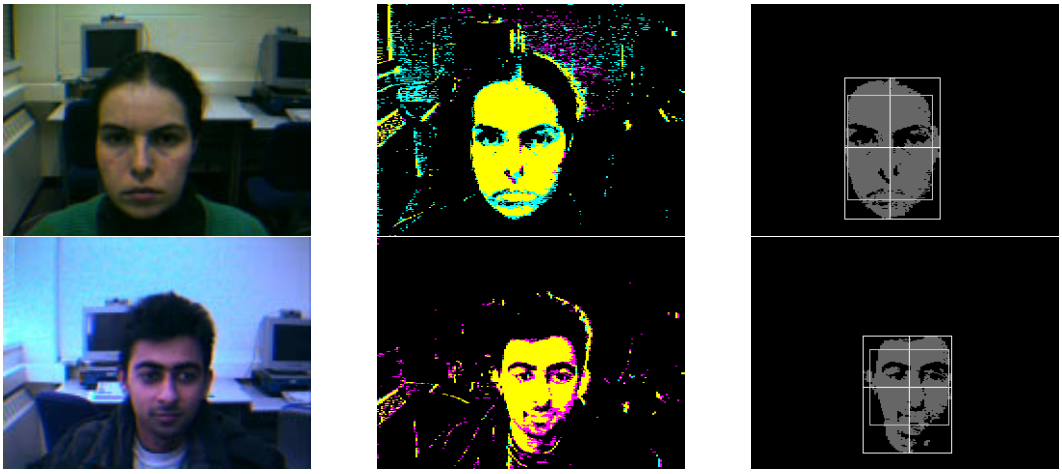


Figure 6.13: Face extraction examples: (left) input image, (middle) labeled image, (right) boundaries of the extracted face region. (Images from V. Bakic.)

key stage is the clustering of the thousands of color codes in the original image to obtain the labeled image with only a few labels. In the face extraction example, clusters were decided "by hand", but sometimes automatic clustering is done. Segmentation is covered thoroughly in Chapter 10.

6.6 Shading

There are several complicating factors in both the physics of lighting and human perception. Surfaces vary in *specularity*, that is, how much they act like a mirror. Highly specular surfaces reflect a ray of incident energy in a restricted cone about a ray of reflection. *Matte* surfaces reflect energy equally in all directions. Thus, a surface not only has a wavelength dependent bias in reflecting incident radiation, but it also has directional bias. Moreover, the energy or intensity of radiation depends upon distance – surface elements farther from a point source of white light will receive less energy than closer surface elements. The effect is similar between the radiating object and the sensor elements. As a result, image intensities will be nonuniform due to the nonuniform distances along the imaging rays. The orientation θ of the surface element relative to the source is also very important.

6.6.1 Radiation from One Light Source

Consider radiation from a single distant light source reaching an object surface as shown in Figure 6.14. Currently, there is no view position from which we observe the surface; we are only considering how the surface is irradiated by the light source. We assume that the light source is far enough away so that the direction from all surface elements of the illuminated object to the light source can be represented by a single unit length direction vector $\mathbf{s}$. The light energy per unit area (*intensity* i) that reaches each surface element A_j is proportional to the area of the surface element times the cosine of the angle that the surface element

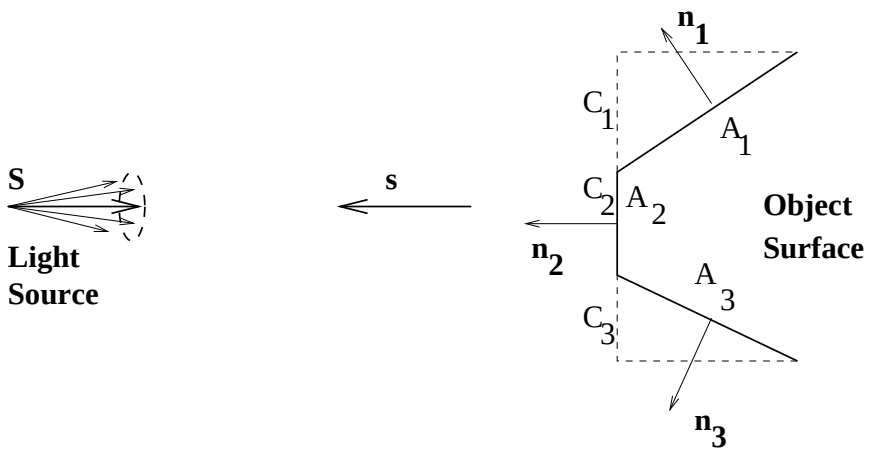


Figure 6.14: Object surface elements A_j irradiated by light source **S** receive energy proportional to the cross section $C_j = A_j \, cos\theta_j$ presented to the source. Intensity of received radiation is thus $i \sim \mathbf{n} \circ \mathbf{s}$ where $\mathbf{n}$ is the unit normal to the surface and $\mathbf{s}$ is the unit direction toward the source. θ_j is the angle between the surface normal $\mathbf{n_j}$ and $\mathbf{s}$.

makes with the illumination direction $\mathbf{s}$. The cosine of the angle is $\mathbf{n} \circ \mathbf{s}$, where $\mathbf{n}$ is the unit vector normal to the surface element A_j. Thus our mathematical model for the intensity of radiation received by a surface element is

$$\text{received} \quad i \sim \mathbf{n} \circ \mathbf{s}. \tag{6.5}$$

The radiation received is directly proportional to the power of the light source, which may or may not be known. The light source may radiate energy in all directions, or may be like a *spotlight* radiating light only in a small cone of directions. In either case, the power of the light source is expressed in watts per steradian, or energy per unit area of a conical sector of a unit sphere centered at the source. This simple model for irradiating surface elements extends readily to curved surfaces by considering that the rectangular surface elements become infinitesimally small in the limit. The fraction of the incident radiation that a surface element reflects is called its *albedo*.

2 DEFINITION *The* **albedo** *of a surface element is the ratio of the total reflected illumination to the total received illumination.*

We have assumed that albedo is an intrinsic property of the surface: for some surfaces this is not the case because the fraction of illumination reflected will vary with the direction of lighting relative to the surface normal.

6.6.2 Diffuse Reflection

We now extend our model to consider the reflection from object surfaces; moreover, we model how the surface element appears from some viewing position $\mathbf{V}$. Figure 6.15 shows *diffuse* or *Lambertian* reflection. Light energy reaching a surface element is reflected evenly

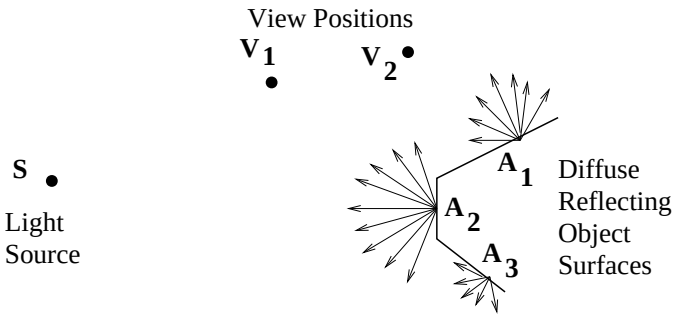


Figure 6.15: Diffuse, or Lambertian, reflection distributes energy uniformly in all directions of the hemisphere centered at a surface element. Thus an entire planar patch will appear to be of uniform brightness for all viewpoints from which that surface is visible.

in all directions of the hemisphere centered at that surface element. Diffuse reflections occur from surfaces that are rough relative to the wavelength of the light. The intensity of the reflected illumination is proportional to the intensity of the received illumination: the constant factor is the *albedo* of the surface, which is low for dark surfaces and high for light surfaces.

$$\text{diffuse reflected} \quad i \sim \mathbf{n_j} \circ \mathbf{s} \tag{6.6}$$

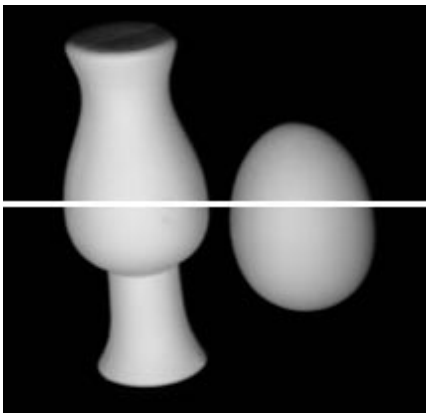
3 DEFINITION *A* **diffuse** *reflecting surface reflects light uniformly in all directions. As a result, it appears to have the same brightness from all viewpoints.*

The critical characteristic is that the surface element will have the same brightness when viewed from the entire hemisphere of directions, because its brightness is independent of the viewer location. From Figure 6.15, surface element A_1 will have the same brightness when viewed from positions V_1 or V_2; similarly, surface A_2 will also appear to be of the same brightness when viewed from either V_1 or V_2. If all three surface elements are made of the same material, they have the same albedo and thus A_2 will appear brighter than A_1, which will appear brighter than A_3, due to the angles that these surfaces make with the direction of the illumination. Surface element A_3 will not be seen at all from either position V_1 or V_2. (A surface element will not be visible if $\mathbf{n} \circ \mathbf{v} < 0$ where $\mathbf{v}$ is the direction toward the viewer.)

A convincing example of diffuse reflection is given in Figure 6.16, which shows intensity of light reflected off an egg and a blank ceramic vase. Intensities from a row of the image are much like a cosine curve, which demonstrates that the shape of the object surface is closely related to the reflected light as predicted by Equation 6.6.

Exercise 14

Consider a polyhedral object of diffusely reflective material such that face F directly faces a distant light source S. Another face A adjacent to F appears to be half as bright as F. What is the angle made between the normals of faces A and F?



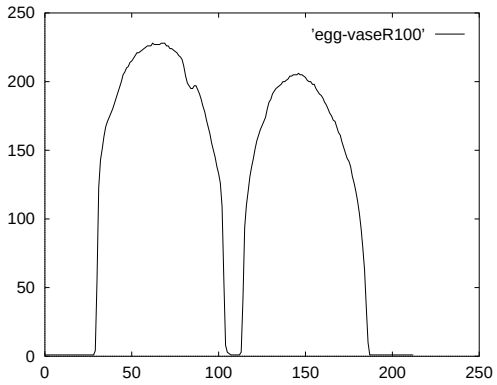


Figure 6.16: Diffuse reflection from Lambertian objects — a vase and an egg — and a plot of intensities across the highlighted row. The intensities are closely related to the object shape.

6.6.3 Specular reflection

Many smooth surfaces behave much like a mirror, reflecting most of the received illumination along the ray of reflection as shown in Figure 6.17. The ray of reflection($\mathbf{R}$) is coplanar with the normal($\mathbf{N}$) to the surface and the ray of received illumination($\mathbf{S}$), and, makes equal angles with them. A perfect mirror will reflect all of the light energy received from source $\mathbf{S}$ along ray $\mathbf{R}$. Moreover, the reflected energy will have the same wavelength composition as the received light regardless of the actual color of the object surface. Thus a *red* apple will have a *white* highlight, or twinkle, where it reflects a white light source. Equation 6.7 gives a mathematical model of specular reflection commonly used in computer graphics. Equation 6.8 defines how to compute the reflected ray $\mathbf{R}$ from the surface normal and the direction toward the source. The parameter α is called the *shininess* of the surface and has a value of 100 or more for very shiny surfaces. Note that as α increases, $cos\phi^{\alpha}$ decreases more sharply as ϕ moves away from 0.

$$\text{specular reflected} \quad i \sim (\mathbf{R} \circ \mathbf{V})^{\alpha} \tag{6.7}$$

$$\mathbf{R} = 2\mathbf{N}(\mathbf{N} \circ (-\mathbf{S})) \oplus \mathbf{S} \tag{6.8}$$

4 Definition **Specular reflection** *is mirrorlike reflection. Light reflected off the surface is radiated out in a tight cone about to the ray of reflection. Moreover, the wavelength composition of the reflected light is similar to that of the source and independent of the surface color.*

5 Definition *A* **highlight** *on an object is a bright spot caused by the specular reflection of a light source. Highlights indicate that the object is waxey, metalic, or glassy, etc.*

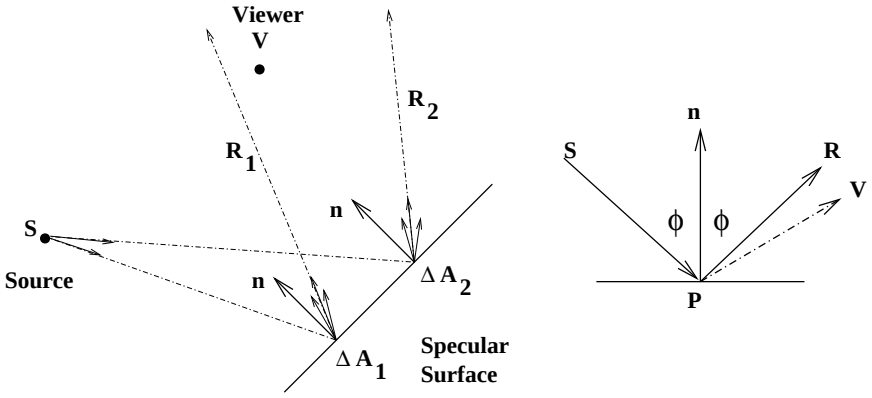


Figure 6.17: Specular, or mirrorlike, relection distributes energy in a narrow cone about the ray of reflection $\mathbf{R}$. The viewpoint $\mathbf{V}$ receives some reflected energy from surface element ΔA_1 but very little from surface element ΔA_2. The intensity received at $\mathbf{V}$ is $e_i \sim (\mathbf{R} \circ \mathbf{V})^{\alpha}$, where $\mathbf{R}$ is the ray of reflection, $\mathbf{V}$ is the direction from the surface element toward the viewpoint and α is the *shininess* parameter.

6.6.4 Darkening with Distance

The intensity of light energy reaching a surface decreases with the distance of that surface from the light source. Certainly our Earth receives less intense radiation from the sun than does Mercury. A model of this phenomena is sketched in Figure 6.18. Assuming that a source radiates a constant energy flux per unit of time, any spherical surface enclosing that source must intercept the same amount of energy per unit time. Since the area of the spherical surface increases proportional to the square of its radius, the energy per unit area must decrease proportional to the inverse of the radius squared. Thus, the intensity of light received by any object surface will decrease with the square of its distance from the source. Such a distance is labeled d_1 in Figure 6.18. The same model applies to light energy reflected from the object surface elements: thus, a viewer at position V in space will observe a surface brightness (d_2) inversely proportional to the square of the distance from that surface element. This inverse square model is commonly used in computer graphics to compute the shading of rendered surfaces so that 3D distance, or depth, can be communicated to a user.

Exercise 15

An inventor wants to sell the traffic police the following device for detecting the speed of cars at night. The device emits a very short flash of light at times t_1 and t_2 and senses the reflection back from the car. From the intensities of the reflections, it computes the distances d_1 and d_2 for the two time instants using the principle in Figure 6.18. The speed of the car is simply computed as the change in distance over the change in time. Critique the design of this instrument. Will it work?

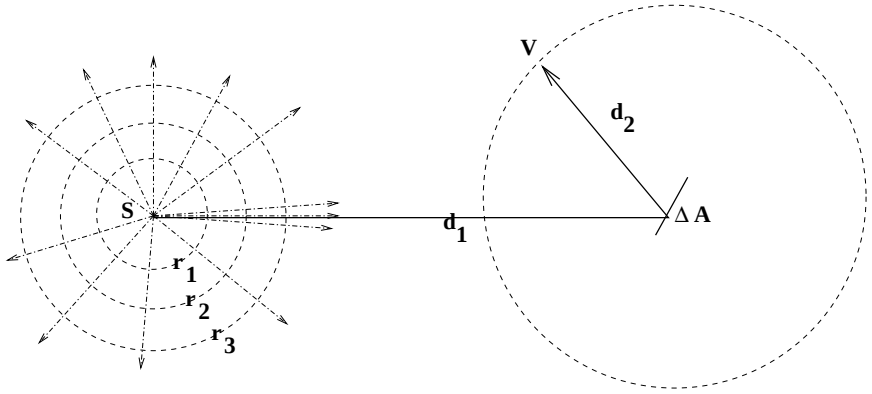


Figure 6.18: The total energy radiating from a point source through any enclosing spherical surface is the same: thus the energy per unit area of surface, or intensity, must decrease inversely with the square of the radius of the enclosing sphere (d_1). Similarly, light energy reflecting off a surface element must descrease in intensity with the distance (d_2) from which that surface is viewed.

6.6.5 Complications

For most surfaces, a good reflection model must combine both diffuse and specular reflections. If we view an apple with a flashlight, we will actually see a reddish object with a whitish highlight on it: the reddish reflections are from diffuse reflection, while the highlight is from specular reflection. Were the apple entirely specular, then we wouldn't be able to observe most of its surface.

Often there are many light sources illuminating a scene and many more surface elements reflecting light from these sources. We might not be able to account for all the exchanges of energy, except by saying that there is *ambient* light in the scene. In computer graphics, it is common to use an ambient light factor when shading a surface.

6 Definition **Ambient light** *is steady state light energy everywhere in the scene resulting from multiple light sources and the interreflections off many surfaces.*

Some surfaces actually emit light. These might be light bulbs or perhaps an object that absorbs one kind of energy and then emits it in the visible band. Such objects will reflect light as well as emit it. Finally, all of our emitting or reflecting phenomena are wavelength dependent. A source emits an entire spectrum of different wavelengths (unless it is a monchromatic laser) and a surface reflects or absorbs energy in some wavelengths more than others. Machines can be built to be sensitive to these wavelength phenomena; for example, multispectral scanners can produce 200 values for reflection from a single surface element. For humans, however, we can summarize a sample of visible light using a combination of only three values, such as RGB or HSI. Computer graphics commonly describes both illumination and surface reflection in terms of only RGB components.

Exercise 16

An amateur photographer took a photo of friends at the rim of the Grand Canyon just after sunset. Although a flash was used and the images of the friends were good, the beautiful canyon wall background was almost black. Why?

6.6.6 * Phong Model of Shading

A popular shading model used in computer graphics is the Phong shading model, which accounts for several phenomena; (a) ambient light, (b) diffuse reflection, (c) specular reflection, and (d) darkening with distance. Components (b),(c), and (d) are summed for each separate light source. We assume that we know the details of the surface element imaging at image point $I[x, y]$ and the position and characteristics of all light sources. The reflective properties of this surface element are represented by $K_{d\lambda}$ for diffuse reflectivity and $K_{s\lambda}$ for specular reflectivity, where $K_{q\lambda}$ is a vector of coefficients of reflection for different wavelengths λ — usually three of them for RGB.

$$I_\lambda[x, y] \;=\; I_{a\lambda}K_{d\lambda} \;+\; \sum_{m=1}^{M}(\frac{1}{cd_m^2}I_{m\lambda}[K_{d\lambda}(\mathbf{n} \circ \mathbf{s}) \;+\; K_{s\lambda}(\mathbf{R}_m \circ \mathbf{V})^\alpha]) \tag{6.9}$$

Equation 6.9 uses ambient illumination $I_{a\lambda}$ and a set of M light sources $I_{m\lambda}$. The equation can be thought of as a vector equation treating each wavelength λ similarly. $I_{a\lambda}$ is the intensity of ambient light for wavelength λ, $I_{m\lambda}$ is the intensity of the light source m for wavelength λ. The $m-th$ light source is a distance d_m from the surface element and makes reflection ray $\mathbf{R}_m$ off the surface element.

6.6.7 Human Perception using Shading

There is no doubt that human perception of three-dimensional object shape is related to perceived surface shading. Moreover, the phenomena described account for shading that we perceive, although the above models of illumination and reflection are simplified. The simplified models are of central importance in computer graphics and various approximations are used in order to speed up rendering of lit surfaces. In controlled environments, computer vision systems can even compute surface shape from shading using the above formulas: these methods are discussed in Chapter 13. We could, for example, compute surface normals for the surface points shown in Figure 6.16 by calibrating our formulas. In uncontrolled scenes, such as outdoor scenes, it is much more difficult to account for the different phenomena.

6.7 * Related Topics

6.7.1 Applications

Color features make some pattern recognition problems much simpler compared to when only intensity, texture, or shape information are available. Color measurements are local; aggregation methods and shape analysis may not be needed. For example, as indicated in Exercise 13, pixel level color information goes a long way in classification of fruits and vegetables for automatic charging at the grocery store or for quality sorting in a distribution center. A second example is the creation of a filter to remove pornographic images from the

WWW. The face detection algorithm as described above first detects skin color according to the training data: regions of skin pixels can then be aggregated and geometric relations between skin regions computed. If it is probable that bare body parts fill a significant part of the image, then that image could be blocked. Color is useful for access to image databases, as described in Chapter 8, and for understanding of biological images taken through a microscope.

6.7.2 Human Color Perception

Characteristics of human color perception are important for two reasons; first, the human visual system is often an efficient system to study and emulate, secondly, the main goal of graphic and image displays is to communicate with humans. The machine vision engineer often wants to learn how to duplicate or replace human capabilities, while the graphic artist must learn how to optimize communication with humans.

Humans in general have biased interpretations of colors. For example, wall colors are generally unsaturated pastels and not saturated colors; reds tend to stimulate, while blues tend to relax. Perhaps 8% of humans have some kind of color blindness, meaning that color combinations should be chosen carefully for communication. In the human retina, red and green sensitve receptors greatly outnumber blue receptors; this is accentuated in the high resolution fovea where blue receptors are rare. As a result, much color processing occurs in neurons that integrate input from the receptors. Various theories have been proposed to explain color processing in terms of processing by neurons. This higher level processing is not fully understood and human visual processing is constantly under study. The color of single pixels of a display cannot be accurately perceived, but humans can make good judgements about the color of an extended surface even under variations of illumination, including illumination by only two principal wavelengths. Often, separate edge-based processing of intensity (Chapter 5) that is faster than color processing yields object recognition before color processing is complete. Theories usually address how human color processing might have evolved on top of more primitive intensity processing. The reader can pursue the vast area of human visual perception by referring to the references and following other references given there.

6.7.3 Multispectral Images

As discussed in Chapter 2, a sensor that obtains 3 color measurements per pixel is a multispectral sensor. However, sensing can be done in bands of the electromagnetic spectrum that are not perceived as color by humans; for example, in infrared bands of the spectrum. In IR bands of a satellite image, hot asphalt roads should appear bright and cold bodies of water should appear dark. Having multiple measurements at a single pixel is often useful for classifying the surface imaged there using simple procedures. The scanning system can be expensive, since it must be carefully designed in order to insure that the several frequency bands of radiation are indeed collected from the same surface element. The parameters of MRI scanning (refer to Chapter 2) can be changed to get multiple 3D images, effectively yielding m intensities for each voxel of the volume scanned. These n measurements can be used to determine whether the voxel material is fat, blood, muscle tissue, etc. The reader

might be alarmed to learn that it can take a full hour to obtain a 3D volume of MRI data, implying that some noise due to motion will be observed, particularly near the boundaries between different tissues where the material element sampled is most likely to change during the scanning process due to small motions caused by circulation or respiration.

6.7.4 Thematic Images

Thematic images use *pseudo color* to encode material properties or use of space represented in an image. For example, pixels of a map or satellite image might be labeled for human consumption so that rivers are blue, urban areas are purple and roads are red. These are not the natural colors recorded by sensors but communicate image content well in our culture. Weather maps might show a temperature theme with red for hot and blue for cold. Similarly, thematic images can encode surface depth, local surface orientation or geometry, texture, density of some feature or any other scalar measurement or nominal classification. The two center images in Figure 6.13 are thematic images: the yellow, blue and purple colors are just labels for three clusters in the real color space. It is important to remember that thematic images do not show actual physical sensor data but rather transduced or classified data for better visualization by a human.

6.8 References

For a detailed treatment of light and optics, one can consult the text by Hecht and Zajac (1974). Some of the treatment of practical digital encoding of color was derived from Murray and VanRiper (1994): the reader can consult that book for many details on the many file formats used to store digital images. Details of the design of color display hardware, especially the shadow-mask technology for color displays, can be found in the graphics text by Foley *et al* (1996). The book by Levine (1985) contains the discussion of several different biological vision systems and their characteristics as devices. More detail is given in the book by Overington (1992), which takes a technical signal processing approach. Livingston (1988) is a good start in the psychology literature. The discussion of matching color histograms was drawn from Swain and Ballard (1991) and Jain and Vailaya (1996). More details on the face extraction work can be found in a technical report by Bakic and Stockman (1999). Work on multispectral analysis of the brain using MRI can be found in the paper by Taxt and Lundervold (1994).

1. V. Bakic and G. Stockman (1999), *Menu Selection by Facial Aspect*, Proceedings of Vision Interface '99, Trois Rivieres, Quebec, Canada (19-21 May 99)

2. M. Fleck, D. Forsyth and C. Pregler (1966) *Finding Naked People*, Proceedings of the European Conference on Computer Vision, Springer-Verlag, 593-602.

3. J. Foley, A. van Dam, S. Feiner and J. Hughes (1996) **Computer Graphics: Principles and Practice, 2nd Ed in C**, Addison-Wesley.

4. E. Hecht and A. Zajac (1974) **Optics**, Addison-Wesley.

5. A. Jain and A. Vailaya (1996) *Image retrieval using color and shape*, Pattern Recognition, Vol 29, No. 8, 1233-1244.

6. M. Levine (1985) **Vision in Man and Machine**, McGraw-Hill.

7. M. Livingstone, (1988) *Art, Illusion and the Visual system*, Scientific American, Jan. 1988, 78-85.

8. J. Murray and W. VanRyper (1994) **Encyclopedia of Graphical File Formats**, O'Reilly and Associates, Sebastopol, CA.

9. I. Overington (1992) **Computer Vision: A unified, biologically-inspired approach**, Elsevier, Amsterdam.

10. M. Swain and D. Ballard (1991) *Color Indexing*, International Journal of Computer vision, Vol. 7, No. 1, 11-32.

11. T.Taxt and A. Lundervold (1994) *Multispectral Analysis of the Brain in Magnetic Resonance Imaging*, in Proceedings of the IEEE Workshop on Biomedical Image Analysis, Seattle,WA (24-25 June 1994) 33-42.