Chapter 15

Point Operations

Once the image data has been sampled, quantized, and stored in the computer, the next task is processing to improve the image, i.e., to extract some (or more) information from the data. The various image processing operations $O\{\}$ are applied to the digital input image $f_s[n \cdot \Delta x, \ m \cdot \Delta y]$ to obtain a (generally) different output $g_s[n' \cdot \Delta x, \ m' \cdot \Delta y]$. From this point on, all images may be considered as sampled data and thus the subscript $s$ will be ignored and the coordinates will be labeled by $[x, y]$. The general operator has the form

$$O\{f(x, y)\} = g(x', y')$$

The various operators $O$ can be grouped based on the number and location of pixels of the input image $f$ that affect the computation of a particular output pixel $g(x, y)$. One possible set of categories is:

1. **Point Operations on single images**: The gray value of the output image $g$ at a particular pixel $[x, y]$ depends ONLY on the gray value of the same pixel in $f$; examples of these operations include contrast stretching, segmentation based on gray value, and histogram equalization;

2. **Point Operations on multiple images**: The gray value of the output pixel $g(x, y)$ depends on the gray values of the same pixel in a set of input images $f(x, y, t_n)$ or $f(x, y, \lambda_n)$; examples are segmentation based on variations in time or color; multiple-frame averaging for noise smoothing, change detection, and spatial detector normalization;

3. **Neighborhood Operations on one image**: The gray value of $g$ at a particular pixel $[x, y]$ depends on the gray values of pixels in the neighborhood of the same pixel in $f(x, y)$; examples include convolution (as for image smoothing or sharpening), and spatial feature detection (e.g., line, edge, and corner detection);

4. **Neighborhood Operations on multiple images**: This is just a generalization of (3); the pixel $g(x, y)$ depends on pixels in the spatial and temporal
(or spectral) neighborhood of \([x, y, t_n \text{ or } \lambda_n]\). Spatial / temporal convolution or spatial / spectral convolution

5. Operations based on Object “Shape” (e.g., “structural” or “morphological”) operations: The gray level of the output pixel is determined by the object class to which a pixel belongs; examples include classification, segmentation, data compression, character recognition;

6. Geometrical Operations: The pixels \(f[x, y]\) are remapped to a new coordinate system to obtain \(g[x, y]\): image warping, cartography;

7. “Global” Operations: The gray value of the output image at a pixel depends on the gray values of all of the pixels of \(f[x, y]\); these include image transformations, e.g., Fourier, Hartley, Hough, Haar, Radon transforms

15.1 Point Operations on Single Images

The gray value of each pixel in the output image \(g[x, y]\) depends on the gray value of only the corresponding pixel of the input image \(f[x, y]\). Every pixel of \(f[x, y]\) with the same gray level maps to a single (usually different) gray value in the output image.

\begin{center}
\begin{tikzpicture}
\node (input) at (0,0) {
\begin{tabular}{c}
\textbf{Input} \\
\end{tabular}
};
\node (output) at (4,0) {
\begin{tabular}{c}
\textbf{Output} \\
\end{tabular}
};
\node (fxy) at (0,-2) {
\begin{tabular}{c}
\text{\(f[x, y]\)} \\
\end{tabular}
};
\node (gxy) at (4,-2) {
\begin{tabular}{c}
\text{\(g[x, y]\)} \\
\end{tabular}
};
\draw[->] (input.east) -- (output.west);
\end{tikzpicture}
\end{center}

\textit{Schematic of a point operation on a single image: the gray value of the output pixel is determined ONLY by the gray value of the corresponding input pixel.}

In a point operation, the only available parameter that determines the output pixel is the gray value of that one input pixel. Therefore, the point operation must affect all pixels with the same input gray level \(f_0\) in the same fashion; they all change to the same output gray value \(g_0\). When designing the action of the point operation, it often is very useful to know the pixel population \(H\) as a function of gray level \(f\):
the histogram $H[f]$ of the image. The amplitude of the histogram at gray value $f$ is proportional to the probability of occurrence of that gray value.

### 15.1.1 Image Histograms

The histogram of the 2-D image $f[x, y]$ plots the population of pixels with each gray level $f$. The histogram generally is represented as a 1-D function $H[f]$ where the independent variable is the gray value $f$ and the dependent variable is the number of pixels $H$ with that level. The histogram depicts a particular feature of the image: the population of gray levels. It arguably represents the simplest feature of the image, i.e., a measure of a useful characteristic of the image. The histogram may be called a feature space and its properties may be used to segment the image pixels into component groups.

The histogram often contains valuable global information about the image. For example, most pixels in a low-contrast image are contained in a narrow range of gray levels, so the histogram is concentrated within that small interval. An image with a bimodal histogram (the histogram of gray values exhibits two modes $\Rightarrow$ a histogram with two “peaks”) often consists of a foreground object (whose pixels are concentrated around a single average gray value) on top of a background object whose pixels have a different average gray value.

Because all pixels in the image must have some gray value in the allowed range, the sum of populations of the histogram bins must equal the total number of image pixels $N$:

$$\sum_{f=0}^{f_{\text{max}}} H[f] = N$$

where $f_{\text{max}}$ is the maximum gray value ($f_{\text{max}} = 255$ for an 8-bit quantizer). The histogram function is a scaled replica of the probability distribution function of gray levels in that image. The discrete probability distribution function $p[f]$ must satisfy
the constraint:

\[
\sum_{f=0}^{f_{\text{max}}} p[f] = 1
\]

and therefore the probability distribution and the histogram are related by the simple expression:

\[
p[f] = \frac{1}{N} H[f]
\]

As will be described later during the discussion of image compression and information theory, the probability distribution leads to a measure of the “quantity” of information in the image, which is the minimum number of bits of data that is required to store the image and generally is measured in bits per pixel. If the probability of gray level \( f \) in the image \( f[x,y] \) is represented as \( p[f] \), the definition of the quantity of information in the image is:

\[
I[f] = -\sum_{f=0}^{f_{\text{max}}} p[f] \log_2 (p[f]) \quad \text{[bits/pixel]}
\]

From this definition, it is easy to show that the maximum information content is obtained if each gray level has the same probability; in other words, a flat histogram corresponds to maximum information content.

### 15.1.2 Histograms of Typical Images

The form of the image histogram often indicates the character of the original image: a bitonal or binary image will have only two gray levels occupied; an image composed of a small dark object on a large bright background will have a bimodal histogram; a low-contrast image will have a small number of contiguous levels occupied; and an image with a large information content will have a flat histogram.
15.1 POINT OPERATIONS ON SINGLE IMAGES

15.1.3 Cumulative Histogram

Given an \( N \)-pixel image \( f[x, y] \) having gray values in the range \( 0 \leq f \leq f_{\text{max}} \) and histogram \( H[f] \), then the cumulative histogram evaluated at gray value \( f_0 \) is the number of pixels with gray value less than or equal to \( f_0 \):

\[
C[f_0] = \sum_{f=0}^{f_0} H[f] = \frac{1}{N} \sum_{f=0}^{f_0} p[f]
\]

The value of the cumulative histogram at the maximum gray value is the number of pixels in the image:

\[
C[f_{\text{max}}] = \sum_{f=0}^{f_{\text{max}}} H[f] = N
\]

In the case of a continuous probability distribution, the cumulative histogram is an integral over gray level:

\[
C[f_0] = \int_{f=0}^{f_0} H[f] \, df
\]

The cumulative histogram is used to derive the mapping that maximizes the global visibility of changes in image gray value (histogram equalization) and for deriving an output image with a specific histogram (histogram specification).
If $H[f]$ is flat, so that every gray level “has” the same number of pixels, then the associated cumulative histogram $C[f]$ increases by the same number of pixels for each gray value, and thus forms a linear “ramp” function at 45°. The cumulative histogram of a low-contrast image rises rapidly in the gray levels where most pixels lie and slowly over the other levels.

15.1.4 Histogram Modification for Image Enhancement

In point processing, the only parameter available in the pixel transformation is the gray value of that pixel; all pixels of the same gray level must be transformed identically by a point process. The mapping from input gray level $f$ to output level $g$ is called a \textit{lookup table}, or \textit{LUT}. Lookup tables can be graphically plotted as transformations $g[f]$ that relate the input gray level (plotted on the $x$-axis) to the output gray level (on the $y$-axis). One such operation is the “spreading out” of the compact
histogram from a low-contrast image over the full available dynamic range to make the image information more visible.

Examples of Point Operations

The output resulting from the first mapping below is identical to the input, while the output derived from the second mapping has inverted contrast, i.e., white→black.

First row: identity lookup table \(g[f] = f\), the resulting image \(g[x,y] = f[x,y]\) and its histogram. Second row: the “negative” lookup table \(g[f] = 255 - f\), the resulting image, and its histogram, showing that the histogram is “reversed” by the lookup table.
First row: lookup table that decreases the contrast, the resulting image, and the histogram showing the concentration of pixels in the middle third of the range.  
Second row: lookup table for linear contrast enhancement, its result when applied to the low-contrast image, and the “spread-out” histogram.

Remapping Histograms

As already mentioned, the image histogram is proportional to the probability distribution of gray levels in the image. The action of any lookup table on an image may be modeled as a transformation of probabilities. Recall that the area under any continuous probability density $p[f]$ or discrete probability distribution $p_f$ is unity:

$$\int_{-\infty}^{\infty} p[f] \, df = 1 \implies \sum_{n=0}^{\infty} p_n = 1$$

For histograms, the corresponding equations are:

$$\int_{0}^{f_{\text{max}}} H[f] \, df = \sum_{f=0}^{f_{\text{max}}} H[f] = N \text{ (total number of pixels)},$$

which merely states that every image pixel has some gray level between 0 and $f_{\text{max}}$. Similarly for the output image:

$$\sum_{g=0}^{f_{\text{max}}} H[g] = N.$$ 

The input and output histograms $H[f]$ and $H[g]$ are related to the lookup table transformation $g[f]$ via the basic principles of probability theory. The fact that the number of pixels must be conserved requires that incremental areas under the two histograms must match, i.e., if input gray level $f_0$ becomes output level $g_0$, then:

$$H[f_0] \, df = H[g_0] \, dg \text{ (continuous gray levels)}$$

$$H[f_0] = H[g_0] \text{ (discrete gray levels)}$$

These equations merely state that all input pixels with level $f_0$ are mapped to level $g_0$ in the output.

15.1.5 Jones Plots

It may be useful to plot the histogram of the input image, the lookup table, and the histogram of the output image on the same Jones plot that shows the relationship among them. The input histogram is upside down at the lower-right; the output histogram (rotated $90^\circ$ counterclockwise) is at the upper left; the lookup table is at the upper right. The new gray level $g_0$ is determined by mapping the value $f_0$
through the curve $g[f]$ onto the vertical axis. In this case, the gray levels of the original low-contrast image are spread out to create a higher-contrast image.

15.2 Histogram Equalization (“Flattening”)

The quantity of information in an image $f[x,y]$ was defined by Shannon to be:

$$I[f] = -\sum_{f=0}^{f_{\text{max}}} p[f] \log_2 (p[f]) \quad [\text{bits/pixel}]$$

The meaning of information will be considered in more detail in the discussion of image compression. The information content in an image is maximized if all gray levels are equally populated. This ensures that the differences in gray level within the image are spread out over the widest possible range and thus maximizes the ability to distinguish differences in gray values. Therefore, the act of maximizing image information results in the histogram with gray levels populated as uniformly as possible; the process is called histogram equalization or flattening. The appropriate lookup table is proportional to the cumulative histogram of the input image $C[f]$. The mathematical derivation of the appropriate $g[f]$ is straightforward.

Assume the point operation (lookup table) $\mathcal{O}\{f[x,y]\} = g[x,y]$ equalizes the output histogram, i.e., $H[g]$ is flat. For simplicity, assume that gray levels are continuous.
and that the lookup transformation \( g[f] \) is monotonically increasing:

\[
\begin{align*}
    g[f_0] &= g_0 \\
g[f_0 + \Delta f] &= g_0 + \Delta g
\end{align*}
\]

Since the lookup table \( g[f] \) must be a monotonically increasing function, then the corresponding inverse operation must exist (call it \( g^{-1} \)), so that \( g^{-1}[g_0] = f_0 \). Because the number of pixels must be conserved (each pixel in \( f[x,y] \) is also in \( g[x,y] \)), then the continuous probabilities must satisfy the relation:

\[
\begin{align*}
p[f] \ df &= p[g] \ dg \\
\implies \frac{H[f]}{N} \ df &= \frac{H[g]}{N} \ dg \\
\implies H[f] \ df &= H[g] \ dg
\end{align*}
\]

but \( H[g] \) is constant by assumption (flat histogram), so substitute \( H[g] = k \), a constant:

\[
H[f] \ df = k \ dg
\]

Integrate both sides over the range of allowed levels from 0 to \( f_0 \). The integral evaluated a gray level \( f_0 \) is:

\[
k \cdot \int_0^{f_0} dg = \int_0^{f_0} H[f] \ df = C[f_0] \\
\implies k \cdot g[f_0] - k \cdot g[f = 0] = C[f_0] \\
\implies g[f_0] = \frac{1}{k} \cdot C[f_0] + g[f = 0]
\]

The proportionality constant \( k \) may be evaluated for the number \( R \) of available gray levels (dynamic range) and the image “area”

\[
k = \frac{A}{R}
\]

In the discrete case of an \( N \times N \) image, the proportionality constant is \( k = \frac{N^2}{M} \), where \( M \) is the number of available gray levels and \( N^2 \) is the number of image pixels.

The lookup table that equalizes the image histogram is:

\[
g_{flat}[f_0] = \frac{M}{N^2} C[f_0] + g[f = 0]
\]

Since all pixels with the same discrete gray level \( f_0 \) are treated identically by the transformation, the histogram is “flattened” by spreading densely occupied gray values into “neighboring,” yet sparsely occupied, gray levels. The resulting histogram is as “flat” as can be obtained without basing the mapping on features other than gray level.

The local areas under the input and flattened histograms must match; where \( H[f] \)
is large, the interval $\Delta f$ is spread out to a larger $\Delta g$, thus enhancing contrast. Where $H[f]$ is small, the interval $\Delta f$ maps to a smaller $\Delta g$, thus reducing contrast.

Jones plot for histogram equalization in the continuous case. The areas under the original and “flattened” histograms must match: where $H[f]$ is large, the interval $\Delta f$ is spread out into a larger $\Delta g$, thus enhancing contrast. Where $H[f]$ is small, the interval $\Delta f$ is compressed into a smaller range of $\Delta g$, thus reducing contrast.

Adjacent well-populated gray levels are spread out, thus leaving gaps (i.e., unpopulated levels) in the output histogram. Pixels in adjacent sparsely populated gray levels of $f[x,y]$ often are merged into a single level in $g[x,y]$. In practice, neighboring values with few pixels may be combined into single levels, thus eliminating the gray-level differences of those pixels.
CHAPTER 15 POINT OPERATIONS

Discrete Case

15.2.1 Example of Histogram Equalization — 1-D “Image”

Because the equalization operation acts on the histogram, and thus only indirectly on the image, the mathematical operation does not depend on the number of spatial dimensions in the input image; the process applies to images with any number of dimensions. For simplicity of presentation, consider first the equalization of a 1-D function. The “image” has the form of a decaying exponential with 256 pixels quantized to 6 bits (values $0 \leq f \leq 63$). The object is shown in (a) its histogram in (b), and its cumulative histogram in (c). Note that the histogram is significantly clustered; there are more “dark” than “light” pixels. The lookup table for histogram equalization is a scaled replica of the cumulative histogram and is shown in (d). The cumulative histogram of the equalized output image is $C[g]$ in (e), and the output histogram $H[g]$ in (f). Note that the form of the output image in (g) is approximately linear, significantly different from the decaying exponential object in (a). In other words, the operation of histogram equalization changed BOTH the spatial character as well as the quantization. The gray levels with large populations (dark pixels) pixels have been spread apart in the equalized image, while levels with few pixels have been compressed together.

Jones plot for contrast enhancement in the discrete case.
Illustration of histogram flattening of a 1-D function: (a) 256 samples of \( f[n] \), which is a decaying exponential quantized to 64 levels; (b) its histogram \( H[f] \), showing that the smaller population of larger gray values; (c) cumulative histogram \( C[f] \); (d) Lookup table, which is scaled replica of \( C[f] \); (e) Cumulative histogram of output \( C[g] \), which more closely resembles a linear ramp; (f) histogram \( H[g] \), which shows the wider “spacing” between levels with large populations; (g) output image \( g[n] \) after quantization.
15.2.2 Example of Histogram Equalization – 2-D “Image”

First row: low-contrast image and its concentrated histogram. Second row: nonlinear cumulative histogram which is proportional to the lookup table for histogram equalization, the resulting histogram (showing the “spreading out” of well-occupied levels and the “smooshing” of levels with small populations), and the resulting image. Third row: the histogram and image resulting from linear contrast enhancement.

15.2.3 Nonlinear Nature of Histogram Equalization

The equalization lookup table in the 1-D example just considered is not a straight line, which means that the gray value $g$ of the “output” pixel is NOT proportional to $f$, and thus the mapping of histogram equalization clearly is NOT linear. For subjective applications, where the visual “appearance” of the output image is the only concern, the nonlinearity typically poses no problem. However, if two images with different histograms are to be compared in a quantitatively meaningful way (e.g., to detect seasonal changes from images taken from an airborne platform), then independent
15.3 HISTOGRAM SPECIFICATION

Histogram equalization of the two images before comparison is not appropriate because the images are generally different nonlinear operations. Nonlinear operations produce unpredictable effects on the spatial frequency content of the scene, as you saw in the linear mathematics course. Images should either be compared after applying linear mappings based on pixels of known absolute “brightness”, or after the histogram specification process discussed next.

Nonlinear mappings are used deliberately in a “compandor”, which is a composite word blending “compressor” and “expandor”. The process of companding is used to maintain the signal dynamic range and thus improve the “signal-to-noise” ratio in a noise reduction system. A common companding system used in audio systems is the well-known Dolby noise reduction system that is still used for recording analog audio signals on magnetic tape. Analog signals are recorded on tape by aligning appropriate percentages of magnetic domains beneath the recording head. Unavoidable statistical variations in the percentage of aligned domains generates an audible noise signal called tape “hiss” even if no signal is recorded. The Dolby system boosts the amplitude of low-level high-frequency input signals before recording; this is called “pre-emphasis.” The amount of amplification decreases with increasing level of the input signal. The complementary process of “de-emphasis” is performed on playback. The annoying tape hiss is attenuated while the recorded signal is faithfully reproduced. Compandors are also used in digital imaging systems to preserve highlights and shadow detail in digital imaging systems.

15.3 Histogram Specification

It is often useful to transform the histogram of an image to create a new image whose histogram “matches” that of some reference image \( f_{\text{ref}} [x, y] \). This process of histogram specification is a generalization of histogram equalization and allows direct comparison of images perhaps taken under different conditions, e.g., LANDSAT images taken through different illuminations or atmospheric conditions. The required transformation of the histogram of \( f_1 \) to \( H[f_{\text{ref}}] \) may be derived by first equalizing the histograms of both images:

\[
O_{\text{REF}} \{ f_{\text{REF}} [x, y] \} = e_{\text{REF}} [x, y]
\]

\[
O_1 \{ f_1 [x, y] \} = e_1 [x, y]
\]

where \( e_n [x, y] \) is the image of \( f_n [x, y] \) with a flat histogram obtained from the operator \( O \{ \} \); the histograms of \( e_{\text{REF}} \) and \( e_1 \) are “identical” (both are flat). The inverse of the lookup table tranformation for the reference image is \( O^{-1} \{ g_{\text{REF}} \} = f_{\text{REF}} \). The lookup table for histogram specification of the input image is obtained by first deriving the lookup tables that would flatten the histograms of the input and reference image. It should be noted that some gray levels will not be specified by this transformation.
and so must be interpolated. The functional form of the operation is:

\[
g_1[x,y] \text{ (with specified histogram)} = O_{REF}^{-1} \{O_1 \{f_1\}\}
\]

\[
= [O_{REF}^{-1} \cdot O_1] \{f_1\} \propto C_{REF}^{-1} \{C_1 \{f_1\}\}
\]

---

**15.4 Application of Histograms to Tone-Transfer Correction**

Histogram specification may be used to compensate for a nonlinear tone-transfer curve to ensure that the overall tone scale is linear. The recorded image \(g_1[n \cdot \Delta x, m \cdot \Delta y]\) is obtained from the sampled input image \(f[n \cdot \Delta x, m \cdot \Delta y]\) through the transfer curve (lookup table) \(g_1[f]\), which may be measured by digitizing a linear step wedge. The inverse of the transfer curve may be calculated and cascaded as a second lookup table.
15.5 APPLICATION OF HISTOGRAMS TO IMAGE SEGMENTATION

\( g_2 \) to linearize the total transfer curve:

\[
g_2 \left[ g_1 \left( f [n \cdot \Delta x, m \cdot \Delta y] \right) \right] = f [n \cdot \Delta x, m \cdot \Delta y]
\]

\[
\Rightarrow g_2 \left[ g_1 \right] = 1 \\
\Rightarrow g_2 \left[ x \right] = g_1^{-1} \left[ x \right]
\]

Note that the display may be linearized in similar fashion. Consider a nonlinear digitizer transfer curve of the form \( g_1 [f] = \sqrt{f} \). The correction curve necessary to linearize the system is:

\[
g_2 \left[ f_1 \left[ f \right] \right] = g_2 \left[ \sqrt{f} \right] = f
\]

\[
\Rightarrow g_2 \left[ x \right] = x^2
\]

15.5 Application of Histograms to Image Segmentation

Obviously, histograms may be used to distinguish among objects in the image that differ in gray level; this is the simplest example of segmentation in a feature space. Consider the bimodal histogram that often indicates the presence of a brighter object on a darker background. A gray value \( f_T \) may be determined from the histogram and used as a threshold to segment the “foreground” object. If the histogram clusters overlap (as they seemingly always do), then there are bound to be some false identifications. If background pixels that should be thresholded to black appear as white, we speak of “false positives”, whereas foreground pixels that are classified as background are “false negatives.”

Bimodal histogram, showing the intermixing of the “tails” of the two object classes, which produces false identifications in the image created by the thresholding lookup table.

The threshold lookup table maps all pixels with gray levels greater than \( f_T \) to white and all others to black. If the histogram clusters are disjoint and the threshold is well chosen (and if the image really contains a bright foreground object), a binary image
of the foreground object will result. In this case, the histogram likely is composed of
two overlapping gaussian clusters, and thus some pixels likely will be misclassified by
the threshold. Segmentation based on gray level only will be imperfect; there will be
false positive pixels (background pixels classified as foreground), and false negative
( foreground classified as background). Consider the crude \( 64 \times 64 \) 5-bit image, which
shows several distinguishable objects even though the histogram exhibits only two
obvious clusters. Segmentation based on this histogram will be unsatisfactory. A
theme of the study of image processing operations will be to improve segmentation by
gathering or processing data to create histograms with compact and distinguishable
clusters.

Segmentation of noisy from histogram; the histogram contains four obvious
"clusters"; the lookup table segmented the clusters at level 158, which segmented the
sky, clouds, and door from the grass, house, and tree, but some white pixels appear
in the grass ("false positives") and some black pixels in the sky ("false negatives").

This result illustrates the goal of histogram segmentation; to find some “feature space”
(histogram) where the clusters of pixels from the various objects are “compact” and
“far apart.”

Other nonlinear mappings may be used for segmentation. For example, the upper
LUT on the left maps background pixels to black and foreground pixels to their
original gray level. The other is a level slicer; gray levels below \( f_1 \) and above \( f_2 \) map
to zero while those with \( f_1 < f[x, y] < f_2 \) are thresholded to white.
15.6 Point Operations on Multiple Images

\[
g[x, y] = \mathcal{O}\{f[x, y, t_n]\}
g[x, y] = \mathcal{O}\{f[x, y, \lambda_n]\}
g[x, y] = \mathcal{O}\{f[x, y, z_n]\}
\]

The output pixel \(g[x, y]\) is a function of the gray value of that pixel in several input images. The input frames may differ in time, wavelength, depth (if they are slices of a 3-D scene), resolution, etc. Most commonly, the gray values of the multiple inputs are combined by arithmetic operations (e.g. addition, multiplication); binary images (i.e. two gray values) may be combined via logical operators (e.g. AND, XOR, etc.). It is also very common to generate a multidimensional histogram from the multiple inputs and use the interpreted data to segment the image via multispectral thresholding.

**Applications:**

1. Image segmentation using multispectral information
2. Averaging multiple frames for noise reduction
3. Change detection by subtraction
4. Windowing images by mask or template multiplication
5. Correct for detector nonuniformity by division

We begin this discussion by immediately digressing to the most common class of multiple-image system, that of color vision where the images differ in the wavelength \(\lambda\).
15.7 Digression: Introduction to Vision and Color

One of the more common systems in this category is the combination of three mono-
ochrome images to create a color image. To understand this principle, we need to
introduce the human visual system (HVS).

15.7.1 The Eye

The eye is (obviously, and no pun intended) the human sensor that collects radiant
energy and forms an image. It contains two positive lens arrangement that generates
a real image on the light-sensitive retina. Kepler (1604) described vision in terms
of the image projected onto the retina. Sheiner confirmed Kepler’s description by
looking at the image created by an eye in 1625.

![Structures in the eye](http://www.tedmontgomery.com/the_eye/)

The eye is nearly spherical (2.4 cm by 2.2 cm across). The vitreous humor (behind
the lens) “supports” the eyeball, much as air inflates a balloon. It contains micro-
particles of cellular debris floating within, that produce entoptic perception. Within
the sclera is the choroid, a dark layer that absorbs stray light in the same manner as
the black coating inside a camera. The retina is a thin layer of light receptor cells
that cover the inner surface of the choroid. The retinas of (at least most) human eyes
have four kinds of receptors: rods and three kinds of cones. The receptors work via
a photochemical reaction in a photopigment.

Humans have ≈ 75 – 150 million rods distributed over the retinal surface. They
are arrayed in groups of several rods connected to a single nerve ending. This feature
increases the sensitivity of the eye, but decreases the spatial resolution discernible by
these receptors. Rods increase in density from the center to about 20° off axis and then
decrease in density out to extreme periphery. Thus they provide an overall picture
of the field of view. The rods contain rhodopsin, which is a “blue-green” pigment,
but they are not sensitive to color and are used at low levels of illumination (scotopic
vision). Rod vision is better in low light situations because a ganglion cell will fire
when certain threshold signal for all the sensors is reached. It is easier to reach the
threshold with more receptors collecting light. Objects that appear brightly colored in daylight are seen as colorless in dim light because only the rods are stimulated.

Humans have $\approx 6 - 7$ million of cones in each eye that are concentrated in the central portion of the retina, the *fovea*. The cones are the color receptors. Each cone is connected to its own nerve ending, thus ensuring a better spatial resolution than the ganged-together rods. Vision generated by cones is called *photopic* and is applicable at “normal” (daylight) levels of illumination. Under these conditions, the eye motion muscles rotate the eyeball until the image of an object falls on the fovea, where the cones are located give color and high resolution. The eye motion “jiggles” the image on the retina; if the image would fade out if kept stationary on a given spot of photoreceptors. Without the fovea the eye would lose 90% of its capability, retaining only peripheral vision.

*Angular distribution of rods and cones across the retina, also showing the location of the “blind spot” due to the optic nerve.*

For simplicity, we often think of the three cone receptors as sensitive to red, green, and blue light, though this is not strictly correct. The English scientist Thomas Young (a contributor to several areas of optics) hypothesized in 1801 that the eye has three color receptors. This was the basis for his theory of *trichromacy*, which observed that the three independent attributes of colored light (hue, saturation, and lightness) suggests that the eye is sensitive to three independent color input signals. Helmholtz hypothesized that the three types of cones are primarily sensitive to short, medium, and long wavelengths ($S, M, L$), though the response curves were assumed to overlap. The three cones contain pigments with peak response at $\lambda \approx 447$ nm ("short", or blue), 540 nm ("medium", or green), and 577 nm ("long", or red). Approximate sensitivity curves are shown in the figure:

![Approximate sensitivity curves for three cone receptors.](image)
Of course, some humans (usually males) are born without the use of one of the cone types (due to missing connections to some nerves or connections to the wrong nerves), or a cone type contains an incorrect photopigment. Such folks suffer from color blindness.

Eye Sensors

![Schematic of a rod (top) and a cone (bottom); light is incident from the left. The shape of the “outer sections” of the receptors (to the right) leads to the names. The photochemical is contained in these outer sections.]

The transduction (transformation) of light energy into electrical energy occurs through a chemical reaction via a photosensitive dye. The photochemical in rods is called rhodopsin (“visual purple”), which is derived from vitamin A (hence the parental enticement to “eat your carrots”). The interaction of light with the molecules of visual purple causes the electrons to oscillate and change the shape of the molecule. The shape change creates an electrical signal that is transmitted through the nerve synapse to the brain. The “new” molecular shape is not stable and it returns to the original shape after some time delay. The process in cones is similar, but not identical. Because cones work under bright illumination, their absorption process is less efficient; some light is “discarded” by scattering from the cones themselves.

Latency

Because the absorption process is chemical, the response of receptor cells is not instantaneous when light arrives or is removed. In the first case, the result is the “latency effect”, while the continuation in response after light is removed is the “persistence response”. The latter effect is the reason why movies and video can convey the illusion of continuous motion from time-sampled data.

Brightness adaptation and discrimination:

Digital images are displayed as a discrete set of intensities. Important to understand how the eye differentiate between different intensity levels. The human visual system adapts to a huge irradiance range of $10^{10}$, from the scotopic threshold (dimmest light) to the bright glare limit. It is described by a logarithmic function of the incident light irradiance.
In photonic vision alone, the range of irradiances is about $10^6$. The transition from scotopic to photopic is gradual from 0.001 – 0.1 millilamberts (or -3 to -1 mL in log scale). This huge dynamic range is accomplished by changing the overall sensitivity of the eye to the average brightness via the process of adaptation. It also is important to understand is the ability of the eye to discriminate between irradiance changes at any specific adaptation level:

Assume that the retina is flooded with a uniform field of illumination of level $\ell$. Then imagine a short, localized flash in the center of the field of view with level $\ell + \Delta \ell$:

![Field used to evaluate the weber ratio.](image)

The Weber ratio is:

$$\text{Weber ratio} = \frac{\Delta \ell_c}{\ell}$$

where $\Delta \ell_c$ is the increment that a subject detects 50% of the time. A large Weber ratio implies poor sensitivity, meaning that a large percentage change in intensity is required for perception.
At low levels of \( \log [\ell] \) (i.e., in the dark range), \( \log \left[ \frac{\Delta I_f}{I_f} \right] \) is large, which means that the ability to discriminate brightnesses is poor. As \( \log [\ell] \) increases, so that the background illumination increases), the Weber Ratio decreases, which means that the ability to discriminate brightnesses improves.

Visual system tends to undershoot or overshoot around the boundary of regions of different intensities.

Note that Weber ratio is large at low levels of illumination. The two branches mean that at low levels of illumination vision is carried out by the rods, at high levels is a function of the cones.

### 15.7.2 Convergence

The large number of sensors must be connected to the brain. Since the processing capacity of the brain is limited (though prodigious!), the signals from multiple sensors are combined into a smaller number of nerves by the process of convergence. The “degree” of convergence is very different for rods and cones: \( \approx 120 \) rods and \( \approx 6 \) cones converge together; in the fovea. Some cones may have their own ganglia.

**Spatial Effect of Convergence**

The individual sensors are tied together by a neural network of ganglion cells within the retina.
Schematic of neural net for lateral inhibition; a strong signal on receptor A inhibits the signal at the neuron for receptor B.

The lateral connections between cells are weighted to diminish the response from neighbors if one cell is stimulated by a strong signal. This lateral inhibition has the effect of introducing an analogue of an impulse response to the imaging system, where the response “on axis” is positive and those of immediate neighbors are negative. If we think of the system as linear (which it most certainly is not, but is an effective model if the eye is well adapted to a small range of “brightnesses”) and shift-invariant, then we can construct a corresponding transfer function via the Fourier transform. A measurement of the eye response under these conditions leads to some valid conclusions.
Campbell-Robson chart for estimating the contrast sensitivity function.

The Campbell-Robson chart shown in the figure consists of a “chirped” sinusoidal grating whose frequency increases linearly to the right and whose modulation decreases vertically. When the chart is viewed at a fixed distance, the observer typically can draw a line on the chart where the grating modulation “disappears;” this line typically has a peak value of the modulation at a nonzero spatial frequency.
Approximate contrast sensitivity function of the HVS, showing peak monochromatic ("luminance") response at a spatial frequency of approximately 6 cycles per angular degree ($\cong 1.7 \cdot 10^{-3}$ cycles per arcsecond).

This observation demonstrates that the response (i.e., the output modulation) of the HVS is larger than zero at DC, increases to its peak response at a frequency of about six cycles per angular degree ($\cong 0.003$ cycles per radian), and falls off at larger spatial frequencies, as shown in (b). The corresponding impulse response resembles that shown below in (a).

The model of the "impulse response" of the eye (a) and the corresponding transfer function (b). The impulse response shows the lateral inhibition.

In this model, the impulse response of the HVS is positive at the origin, but becomes negative within a short distance. In words, this indicates that the response of an eye receptor to a bright source actually subtracts from the neighboring receptors; we actually say that the response of one receptor inhibits the responses of those nearby. In other words, these neighboring receptors must be stimulated more strongly to elicit the same response as the first. The HVS "processor" that generates this response is the network of neural connections behind the retina, the so-called visual neural net. We can use this linear shift-invariant model of the HVS to calculate the visual response to a specific input image. If the stimulus is a simple "stairstep" input irradiance, with regular steps of increasing brightness, the output exhibits "overshoots" at the edges. This is the phenomenon of Mach Bands, which result in a nonuniform visual appearance of uniform areas in the vicinity of a transition in gray level. The eye response enhances edges.
The effect on the image of lateral inhibition: (a) the input is a “stairstep” function; (b) the output shows the “overshoot” characteristic of edge enhancement.

15.7.3 Eye Motions

The photoreceptors do not respond to luminance, but rather to changes in luminance; in other words, the eye is an “AC system.”

Saccadic motion

Ballistic point-to-point jumps that the eye executes 1 to 3 times per second when viewing a scene.

Drift, tremor, flicker

Minute, involuntary motions that the eye executes continuously. Necessary to maintaining vision. If you hold eyes in a fixed position image would fade.

15.7.4 Eye Lens

The power of the eye is due to the refractive effect of the cornea and of the eye lens. As we saw during our discussion of optics, the cornea actually provides more of the power, while the lens varies the power of the system.

Accommodation

*Accommodation* (fine focusing) is performed by the crystalline lens, which is suspended behind the iris by ligaments connected to the ciliary muscles. When relaxed, these muscles pull outward to bring the lens into its “flattest” configuration, so that the radii of the surfaces lengthen, producing a longer focal length. For a perfect eye, from an object at infinity will be focused on the retina if the ciliary muscles are completely relaxed. If the object is positioned closer to the eye, the ciliary muscles
contract, relieving the external tension on the periphery of lens. The radii of the lens surfaces decreases, and thus so does the focal length. This ensures that the image distance remains unchanged. The closest location of an object where the eye can accommodate is the near point, which tends to increase with age (70 mm for a teenager, $\approx 120$ mm for young adult, $\approx 350$ mm at middle age, and $\approx 80$ mm at 60 years of age). People whose near points are abnormally close to the eye are nearsighted. This trait was actually valuable in ancient times because nearsighted people could see fine detail up close, as when working on jewelry. Since the ability of the eye to accommodate has some hereditary basis, it was common for families to have a tradition of this kind of precision work.

Normal wavelength range of human vision is in the wavelength $390 \text{ nm} \lesssim \lambda \lesssim 780 \text{ nm}$ (which actually seems to be too large, particularly on the red end). The limitation on sensitivity to ultraviolet wavelengths is due to absorption by crystalline lens.

**Nearsightedness (myopia)**

Myopic eyes bring parallel rays to focus in front of the retina; in other words, the power of the lens is too large. Myopia is corrected by placing another lens in front of the eye such that the focal point of the combined power lens-eye system is on the retina without changing the power of the eye+auxiliary lens system. We saw in the section on optics that the system power is unchanged if an additional lens is placed at the front focal point of the original system. This can be seen by applying the two-lens equation with the separation distance $\tau = (\varphi_{\text{eye}})^{-1}$

\[
\varphi_{\text{system}} = \varphi_{\text{corrector}} + \varphi_{\text{eye}} - \varphi_{\text{corrector}} \cdot \varphi_{\text{eye}} \cdot (\varphi_{\text{eye}})^{-1} = \varphi_{\text{corrector}} + \varphi_{\text{eye}} - \varphi_{\text{corrector}}
\]

\[
\varphi_{\text{system}} = \varphi_{\text{eye}}
\]

If you are nearsighted and wear glasses, note that the images are the same size with and without correction. Since corrective contact lenses are placed directly upon the cornea, we can assume that the distance between the corrective lens and the power due to the first surface of the cornea is zero, thus the power of the corrected cornea is the sum of the powers of the two lenses:

\[
\varphi'_{\text{cornea}} = \varphi_{\text{corrector}} + \varphi_{\text{cornea}}
\]

**Farsightedness (hyperopia)**

In this condition, the second focal point lies behind the retina, i.e., the lens power is insufficient, or is placed too close to the retina. To increase the bending of the rays a positive lens is placed in front of the eye. A farsighted person can see distant objects sufficiently well, but cannot bring objects close to our eye to increase the angular subtense of fine structure on the retina because the near point is farther away than normal.
Astigmatism
This is an aberration of the eye system due to different radii of curvature of the cornea along different axial directions, thus resulting in different focal lengths for horizontal and vertical objects.

15.7.5 Spatial resolution
The metric of “visual acuity”

- Seek the ability to characterize the resolving power of the eye.
- Snellen tests used by optometrists
- Test for visual acuity where the patient reads Snellen’s chart at a certain distance with one eye, then with the other, and then with both eyes. Acuity is given by the longest distance at which the patient is able to read the letters, divided with the distance at which he should normally be able to read them.

15.8 Color Vision
The three cones contain pigments with peak response at $\lambda \approx 447\text{ nm}$ (“short”, or blue), 540 nm (“medium”, or green), and 577 nm (“long”, or red). Approximate sensitivity curves are shown:

![Spectral response curves for short (S), medium (M), and long (L) cones.](image)

The response of short, medium, and long cones.

Curves have been determined through physiological measurements of absorption spectra of individual cones in vitro, and psychophysical color matching experiments. The curves are normalized.

The responses of the cones differ, and also the number of short, medium, and long (SML) cones are not equal. There are significantly fewer S cones than the others, so that the S cones contribute little to the overall brightness sensation.

The sensitivity curves provide the basis for color vision: Assume two lights with two wavelengths ($\lambda_1 = 500\text{ nm}$ and $\lambda_2 = 600\text{ nm}$). The first generates a response from M cones that is twice as large as its L response. The second light produces
approximately twice the response in \( L \) cones than \( M \). Now change relative intensities of the two lights: the individual \( L \) and \( M \) responses will be different (corresponding to difference in luminance) but their ratios will be constant, which means that we can separate changes in intensity from changes in color. One single type of detector could not accomplish this!

### 15.8.1 Color Matching, Metamers

A consequence of color matching is that two colors can appear identical to the eye even though they have different spectral compositions. These are called metamers. They appear to match under one illumination but mismatch when viewed under another. This is because color is a sensation rather than a property of an object; the cones can register the same sensation from an infinite variety of combinations of different light frequencies and amplitudes.

**Grassman’s Law**

The eye can distinguish only three attributes of a light: hue, brightness, and saturation. Hue is that psychological dimension of color which roughly corresponds to wavelength. Brightness is the psychological dimension of color which most closely relates to physical intensity and saturation is the amount of hue a color possesses and is most closely correlated with spectral purity. Two lights with two different colors added to two other lights with the same colors produce mixtures of the same color: if \( a = b \) and \( c = d \), then \( a + c = b + d \). Two lights of the same color each subtracted respectively from two other light of the same color will leave mixtures of the same color: if \( a = b \) and \( c = d \) then \( a - c = b - d \). If one unit of light \( a \) has the same color as one unit of \( b \), then \( ka = kb \). Luminance produced by the additive mixture of a number of lights is the sum of luminances produced separately by each light.

Mathematically Grassman’s laws define a vector space \( \Rightarrow \) we can represent any color as a vector in a 3-D space, where the basis vectors are the primary colors. It’s obvious that the primary colors have to be linearly independent, meaning that any one cannot be created as a weighted sum of the others. In this vector space, an arbitrary color \( C \) can be matched by appropriate quantities of the three primaries \( R, G, B \):

\[
C = r_1 R + g_1 G + b_1 B
\]

where \( r_1, g_1, b_1 \) are the weights (number of units) applied to each primary to make the match.

**Trichromancy**

The weighted sum means that every colored light \( C \) is the physical sum of a number of essentially pure spectral components and can be described by a function \( E[\lambda] \) (radiant power per unit wavelength). The result of the experiment across a wide range of test wavelengths produces the so-called color matching functions for primaries at 436 nm, 546 nm, and 700 nm. The color matching functions reveal the fundamental fact that
human color vision is *trichromatic*. Trichromacy simply means that any visible color can be matched by a mixture of three primary colors.

Note that the color matching functions include negative weights. For example, this means that $\lambda = 500\,\text{nm}$ cannot be matched by the three primaries unless some red is subtracted from the mixture. No colors exist that can be used as primaries such that the color matching functions are wholly positive. In 1931, CIE adopted a set of fictitious primaries that result in wholly positive functions:

15.8.2 **XYZ Color Space**

To find the $X$, $Y$, $Z$ values corresponding to a particular color object let $R[\lambda]$ be the spectral transmittance or reflectance of the object, and let $P[\lambda]$ be the spectral power per unit wavelength interval of the illuminating source. Also let $\bar{x}_\lambda$, $\bar{y}_\lambda$, and $\bar{z}_\lambda$
be the CIE positive-definite color matching functions:

\[
\begin{align*}
X &= k \int_{380}^{780} P(\lambda) \ R(\lambda) \ \bar{x}_\lambda \ d\lambda \\
Y &= k \int_{380}^{780} P(\lambda) \ R(\lambda) \ \bar{y}_\lambda \ d\lambda \\
Z &= k \int_{380}^{780} P(\lambda) \ R(\lambda) \ \bar{z}_\lambda \ d\lambda
\end{align*}
\]

where \(k\) is a normalizing factor such that \(Y = 100\) when the object is a perfect white diffuser or a perfect transmitter of light for the entire visible band. \(Y\) is therefore proportional to total luminance and matches the photopic response of the eye. The discrete forms of the equations are:

\[
\begin{align*}
X &= k \sum_{380}^{780} P(\lambda) \ R(\lambda) \ \bar{x}_\lambda \\
Y &= k \sum_{380}^{780} P(\lambda) \ R(\lambda) \ \bar{y}_\lambda \\
Z &= k \sum_{380}^{780} P(\lambda) \ R(\lambda) \ \bar{z}_\lambda
\end{align*}
\]

**Determination of \(X,Y,Z\)**

\[
\begin{bmatrix}
X \\
Y \\
Z
\end{bmatrix} = \begin{bmatrix}
r_1 & g_1 & b_1 \\
r_2 & g_2 & b_2 \\
r_3 & g_3 & b_3
\end{bmatrix} \begin{bmatrix}
R \\
G \\
B
\end{bmatrix}
\]

The color can be characterized independent of its luminance via:

\[
\begin{align*}
x &= \frac{X}{X + Y + Z} \\
y &= \frac{Y}{X + Y + Z} \\
z &= 1 - (x + y) = \frac{Z}{X + Y + Z}
\end{align*}
\]

which are known as *chromaticity coordinates*. The normalization reduces the 3D color space to a 2D plane that satisfies the constraint \(x + y + z = 1\). Purely additive combinations are found inside the triangle:
CHAPTER 15 POINT OPERATIONS

Plane that satisfies the constraint $x + y + z = 1$.

All colors that are possible by a combination of two primaries will lie on a straight line connecting the primaries in this space. Any color that can be derived from any three primaries will lie inside the triangle whose apexs are the primaries; this triangle represents the gamut of the primaries.

15.8.3 CIE Chromaticity Diagram

Additive mixtures of colors lie along a straight line connecting those colors. The complement of any color is found by extending a straight line form that color through white to the opposite side of the CIE diagram.
Definition 15.1 Chromaticity Diagram: A two-dimensional Cartesian plot that depicts the multidimensional subjective relationship among colors perceived by the normal human visual system (eyes and nervous system, including the brain) when additively stimulated by (two or more; usually three) discrete monochromatic visible sources (wavelengths). Note: The familiar CIE chromaticity diagram depicts perceived colors plotted as a function of the normalized relative intensity of a defined red (increasingly red with increased "X", or abscissa, value) versus the normalized relative intensity of a defined green (increasingly green with increased "Y", or ordinate, value). With respect to a given perceived color, as plotted on the chromaticity diagram, the normalized relative intensity of a defined blue at any point is obtained by adding the normalized relative intensities of the red and green, and subtracting the total from 1.

Additive mixtures of colors lie along a straight line connecting those colors. The complement of any color is found by extending a straight line form that color through white to the opposite side of the CIE diagram

http://www.atis.org/tg2k/_chromaticity_diagram.html

In 1931, the CIE defined three standard primaries \((X, Y, Z)\). The Y primary was intentionally chosen to be identical to the average luminous-efficiency function of
human eyes. The three standard CIE primary colors are ($\lambda = 435.8$ nm, $546.1$ nm and $700$ nm).

CIE has recommended the use of other color spaces, all derived from $XYZ$. $L^*a^*b^*$ is used for non-luminous objects such as textiles, paints, plastics, etc. $L^*u^*v^*$ is helpful in the registration of color differences experienced with flashes, photography, television screen, etc. These systems are useful in specifying small differences between color stimuli. The motivation was to find coordinates that relate in a linear fashion to the perceptual attributes of color (perceptually uniform color spaces)

$L^*a^*b^*$

- $L^*$ is the lightness axis and extends from 0 (black) to 100 (white).
- $a^*$ represents the *redness-greenness*
- $b^*$ represents the *yellowness-blueness*

\[
L^* = 116 \left( \frac{Y}{Y_n} \right)^{\frac{1}{3}} - 16
\]
\[
a^* = 500 \left[ \left( \frac{X}{X_n} \right)^{\frac{1}{3}} - \left( \frac{Y}{Y_n} \right)^{\frac{1}{3}} \right]
\]
\[
b^* = 200 \left[ \left( \frac{Y}{Y_n} \right)^{\frac{1}{3}} - \left( \frac{Z}{Z_n} \right)^{\frac{1}{3}} \right]
\]

where $X_n, Y_n, Z_n$ are the coordinates of the reference white.

### 15.8.4 Color Reproduction

A color imaging system usually goes through three steps in order to reproduce an image:

1. **Color separation**: Derive R, G, B signals from the input scene, just like the eye would do. System need more than one kind of detector

2. **Processing**: The system must convert the R, G, B signals from the detectors into output signals that are suitable for the color reconstruction stage. For example a current that controls the amount of red dye to a nozzle. Signal processing may be electronic (TV), chemical (photography), or more complex combination.

3. **Reconstruction**: Image must be produced by whatever means are appropriate: dyes, phosphors, etc. There are two basic methods of color reproduction: subtractive and additive.
**Subtractive Color Reproduction**

Color printing uses three colored dyes: cyan, magenta and yellow. These can be thought of as complementary to the additive primaries red, green and blue. In other words, cyan absorbs the red part of the spectrum so by varying its concentration we can vary the absorption of red while having small effect on other parts of the spectrum.

**Additive Color Reproduction**

Used with lights (e.g., phosphors) of the three additive primary colors. red, green, and blue. There are phosphors with smooth spectral distributions in the blue and green, but red phosphors tend to have “spikey” spectral distributions. As a consequence the full range of human color perception cannot be reproduced.

According to Hunt, there are 6 different types of color reproduction:

1. **Spectral**: Spectral reflectance or transmittance of the image matches the original exactly. This is not possible in normal reproductions since the color gamut does not encompass the entire “horseshoe”.

2. **Colorimetric**: Reproduction matches the original in chromaticity and relative luminance, i.e., the original and reproduced colors are metamers. Of course, whether a reproduction is colorimetric varies with the observer.

3. **Exact**: In addition to colorimetric reproduction we have equality of absolute luminances, i.e., the appearance of colors does not depend on the illuminant intensity.

4. **Equivalent**: Chromaticities, relative luminance, and absolute luminance are adjusted to achieve equality of appearance, i.e., we assume the viewing conditions and adjust by making the reproduction colorimetrically incorrect but perceptually correct.

5. **Corresponding**: Similar to “equivalent” but does not require that absolute luminances match, i.e., adjust chromaticity and relative luminance so as to achieve equality of appearance as if the original was lit by the reproduction illuminant.

6. **Preferred**: Issue with Caucasians who don’t like their skin tone.

**15.8.5 Color Spaces**

**Red, Green, Blue**

This representation is usually graphed in a Cartesian system analogous to $[x, y, z]$, as shown:
RGB coordinates (8 bit) displayed as a Cartesian system. Locations with the same value in each coordinate are “neutral”, e.g., $[0,0,0]$ is “black”, $[255,255,255]$ is “white”, and others are “gray”. Pure colors appear along the axes.

**Hue, Saturation, Lightness (or Brightness, or Value):**

This is the representation that led to Young’s theory:

- **Hue** corresponds to the common definition of color, e.g., “red”, “orange”, “violet” etc., specified by the *dominant wavelength* in a spectrum distribution, though a “dominant” may not actually be present.

- **Saturation** (also called *chroma*): an expression for the “strength” or “purity” of a color. The intensity of a very saturated color is concentrated near the dominant wavelength. Looked at another way, saturation is measured as the amount of “gray” in proportion to the hue. All colors formed from one or two primaries have 100% saturation; if some amount of the third primary is added to a color formed from the other two, then there is at least some “gray” in the color and the saturation decreases. The saturation of a pure white, gray, or black scene (equal amounts of all three primaries) is zero. A mixture of a purely saturated color (e.g., “red”) and white produces a “desaturated red”, or
“pink”. Saturation is reduced if surface reflections are present.

\[
\begin{align*}
R &= 0 & H &= 116 \\
G &= 204 & S &= 255 & \Rightarrow S = \frac{255}{255} = 100% \\
B &= 153 & L &= 102 \\
R &= 25 & 25 & 0 & H &= 115 \\
G &= 204 = 25 + 179 & S &= 199 & \Rightarrow S = \frac{199}{255} = 78% \\
B &= 153 & 25 & 128 & L &= 115 \\
S &= \frac{\text{hue} - \text{gray}}{\text{hue}} = \frac{115 - 25}{115} = 78%
\end{align*}
\]

In more scientific terms, the saturation is the relative bandwidth of the visible output from a light source. A source with a large saturation has a narrow bandwidth, and vice versa. As saturation increases, colors appear more “pure.” As saturation decreases, colors appear more “washed-out”.

- **Brightness**: sensation of intensity of a light, from dark through dim to bright.

- **Lightness**: a relative expression of the intensity of the energy output reflected by a surface; “blackness”, “grayness”, “whiteness” of a visible light source. It can be expressed as a total energy value or as the amplitude at the wavelength where the intensity is greatest.

HSB is often represented in a cylindrical coordinate system analogous to \((r, \theta, z)\). The saturation coordinate is plotted along the radial axis, the hue as the azimuthal coordinate, and the lightness as the vertical \((z)\) axis. The hue determines the frequency of light, the position in the spectrum, or the relative amounts of red, green and blue. It is a continuous and periodic scale that often is measured in an “angle” in angular degrees (e.g., the “hue angle”), though it also may be normalized to be compatible with 8-bit numerical representations. Hues located at the extrema (e.g., angles of \(\pm180^\circ\)) are identical, as shown in the figure taken from the hue adjustment in Adobe Photoshop™. A pure hue is 50% luminosity, 100% saturation. The hue angles are shown, where red corresponds to an angle of 0°.

![Hue Representation](image)

*The hue representation used in Adobe Photoshop™. The hue at angle 0° is “red”*
To illustrate the continuity of the hue circle, it may be rotated about the azimuth and to show how it “wraps around” at cyan ($\theta = 180^\circ$), the complementary color to red.

![Hue Circle Diagram]

*The hue axis after rotation by 180°, showing the “wraparound” at the edge of the axis.*

Hue is represented by 8-bit integer values in other applications, such as Powerpoint\textsuperscript{TM}. A list of the primary colors for different hue angles is shown in the table. Note that the additive primaries are located at $0^\circ$ and $\pm 120^\circ$, while the subtractive primaries are at $\pm 60^\circ$ and $180^\circ$. Colors at opposite sides of the hue circle (separated by $180^\circ$) are complementary, so that the sum of two complementary colors produces white.

The sum of monochromatic yellow ($\lambda = 580$ nm) and monochromatic blue ($\lambda = 480$ nm) produces white light that looks just as while as the sum of all visible wavelengths.

<table>
<thead>
<tr>
<th>Color</th>
<th>Photoshop\textsuperscript{TM} Hue Angle $\theta$ ($^\circ$)</th>
<th>Powerpoint\textsuperscript{TM} Hue $[0, 255]$</th>
</tr>
</thead>
<tbody>
<tr>
<td>cyan</td>
<td>$\pm 180^\circ$</td>
<td>127 $\Rightarrow \frac{1}{2}$</td>
</tr>
<tr>
<td>green</td>
<td>$+120^\circ$</td>
<td>85 $\Rightarrow \frac{1}{3}$</td>
</tr>
<tr>
<td>yellow</td>
<td>$+60^\circ$</td>
<td>42 $\Rightarrow \frac{1}{6}$</td>
</tr>
<tr>
<td>red</td>
<td>$0^\circ$</td>
<td>0</td>
</tr>
<tr>
<td>magenta</td>
<td>$-60^\circ$</td>
<td>213 $\Rightarrow \frac{5}{6}$</td>
</tr>
<tr>
<td>blue</td>
<td>$-120^\circ$</td>
<td>170 $\Rightarrow \frac{2}{3}$</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Color</th>
<th>R</th>
<th>G</th>
<th>B</th>
<th>H</th>
<th>S</th>
<th>L</th>
</tr>
</thead>
<tbody>
<tr>
<td>cyan</td>
<td>0</td>
<td>255</td>
<td>255</td>
<td>127</td>
<td>255</td>
<td>128</td>
</tr>
<tr>
<td>green</td>
<td>0</td>
<td>255</td>
<td>0</td>
<td>85</td>
<td>255</td>
<td>128</td>
</tr>
<tr>
<td>yellow</td>
<td>255</td>
<td>255</td>
<td>0</td>
<td>42</td>
<td>255</td>
<td>128</td>
</tr>
<tr>
<td>red</td>
<td>255</td>
<td>0</td>
<td>0</td>
<td>0</td>
<td>255</td>
<td>128</td>
</tr>
<tr>
<td>magenta</td>
<td>255</td>
<td>0</td>
<td>255</td>
<td>213</td>
<td>255</td>
<td>128</td>
</tr>
<tr>
<td>blue</td>
<td>0</td>
<td>0</td>
<td>255</td>
<td>170</td>
<td>255</td>
<td>128</td>
</tr>
</tbody>
</table>
Microsoft Windows color dialogs also use HSB but call the third dimension “luminosity” or “lightness”. It ranges from 0% (black) to 100% (white).

The RGB model is quite simple, so it is natural to consider the advantages of the HSL color model:

1. You can generate grey scales using only one parameter – the luminosity (set saturation to 0).

2. You can vary the color without changing the brightness – vary the hue alone.

3. You can fade or darken several colors, or whole bitmaps, such that the lightness (or darkness) stay in step.

The HSL model is easier to use visually because it suits the eye, whereas the RGB model is easier to use in programming.

15.8.6 Conversion from RGB to HSL

Recall the transformations between Cartesian coordinates \([x, y, z]\) and cylindrical coordinates \((r, \theta, z)\):

\[
\begin{align*}
  r &= \sqrt{x^2 + y^2} \quad x = r \cos[\theta] \\
  \theta &= \tan^{-1} \left( \frac{y}{x} \right) \quad y = r \sin[\theta] \\
  z &= z \quad z = z
\end{align*}
\]

The transformation from Cartesian to cylindrical coordinates is nonlinear and thus cannot be written as a matrix-vector product. The scheme for computing HSL from RGB is:

1. Normalize the three values \([R, G, B]\) to \([0, 1]\)

2. Find the maximum and minimum of the three values; these are \(color_{\text{max}}\) and \(color_{\text{min}}\)

3. If all three RGB values are identical, then the hue and saturation are both 0
4. Compute the lightness $L$ via

$$L = \frac{\text{color}_{\text{max}} + \text{color}_{\text{min}}}{2}$$

5. Test $L$:

If $L < 0.5$ then $S = \frac{\text{color}_{\text{max}} - \text{color}_{\text{min}}}{\text{color}_{\text{max}} + \text{color}_{\text{min}}}$

If $L < 0.5$ then $S = \frac{\text{color}_{\text{max}} - \text{color}_{\text{min}}}{2 - (\text{color}_{\text{max}} + \text{color}_{\text{min}})}$

6. Compute hue $H$ via:

(a) If $\text{color}_{\text{max}} = R$ then

$$H = \frac{G - B}{\text{color}_{\text{max}} - \text{color}_{\text{min}}}$$

(b) If $\text{color}_{\text{max}} = G$ then

$$H = 2 + \frac{B - R}{\text{color}_{\text{max}} - \text{color}_{\text{min}}}$$

(c) If $\text{color}_{\text{max}} = B$ then

$$H = 4 + \frac{R - G}{\text{color}_{\text{max}} - \text{color}_{\text{min}}}$$

7. Convert $L$ and $S$ back to percentages, and $H$ into an angle in degrees (i.e., scale it from 0-360).

Example:

$$R = 25 \rightarrow \frac{25}{255} \approx 0.098$$

$$G = 204 \rightarrow \frac{204}{255} = 0.8$$

$$B = 53 \rightarrow \frac{53}{255} \approx 0.208$$
\[ \text{color}_{\text{max}} = \frac{G}{255} = 0.800 \]

\[ \text{color}_{\text{min}} = \frac{R}{255} = 0.098 \]

\[ L = \frac{0.8 + 0.098}{2} = 0.449 < 0.5 \]

\[ L = 0.449 \cdot 255 = 115 \]

\[ S = \frac{0.8 - 0.098}{0.8 + 0.098} \cdot 255 = 0.782 \cdot 255 = 199 \]

\[ H = 2 + \frac{0.208 - 0.098}{0.8 - 0.098} \approx 2.157 \text{ radians} \approx +123.6^\circ \]

**Conversion from HSL to RGB**

1. If \( S = 0 \), define \( L = R = G = B \), otherwise, test \( L \):
   - If \( L < 0.5 \), then \( \alpha = L \cdot (S + 1) \)
   - If \( L \geq 0.5 \), then \( \alpha = L + S - L \cdot S \)

2. Set

   \[ \beta = 2.0 \cdot L - \alpha \]

3. Normalize hue angle \( H \) to the range \([0, 1]\) by dividing by \(360^\circ\)

4. For each of \( R, G, B \), compute \( \gamma \) as follows:
   - for \( R \), \( \gamma = H + \frac{1}{3} \)
   - for \( G \), \( \gamma = H \)
   - for \( B \), \( \gamma = H - \frac{1}{3} \)

5. If \( \gamma < 0 \), then \( \gamma = 1 + \gamma \)

6. For each of \([R, G, B]\), do the following test:
   - If \( \gamma < \frac{1}{6} \), then \( \text{color} = \beta + (\alpha - \beta) \cdot 6 \cdot \gamma \)
   - If \( \frac{1}{6} \leq \gamma < \frac{1}{2} \), then \( \text{color} = \alpha \)
   - If \( \frac{1}{2} \leq \gamma < \frac{2}{3} \), then \( \text{color} = \beta + (\alpha - \beta) \cdot (4 - 6\gamma) \)
15.8 COLOR VISION

7. Scale back to the range \([0, 100]\)

8. Repeat steps 6 and 7 for the other two colors.

Other Descriptors of Colors

- Hue: a “pure” color, i.e., one containing no black or white.
- Shade: a “dark” color, i.e., one produced by mixing a hue with black
- Tint: a “light” color, i.e., one produced by mixing a hue with white
- Tone: color produced by mixing a hue with a shade of grey.

15.8.7 Phenomena of Color Vision

Afterimages

This is a well-known optical phenomenon due to the photochemical sensitivity of the eye. If exposed to a stationary scene for an extended period of time, the photochemical dyes that are sensitive to that color become “depleted.” If exposed to a neutral white background, the eye “sees” the complement to that color, because only the dyes sensitive to that color are available. The phenomenon often is illustrated by viewing a representation of the American flag rendered in the colors that lie opposite on the hue angle (cyan for red, yellow for blue, and black for white). Afterimages may be created in Adobe Photoshop™ by using the “inverse” option that cascades a rotation of the hue by \(180^\circ\) and an “inversion” (complementing of) the lightness.

Constructing a specimen for demonstrating the “afterimage:” rotate the hue by \(180^\circ\) and complement the lightness, or just use the “inverse” operation in Adobe Photoshop™. Test it by staring at the processed image for 30+ seconds and then look at a blank sheet of white paper.
15.9 Color and Multispectral Image Processing

We now return to consider pixel processing of multiband images, that may include images differing in color, time, or other parameter. The gray value of each pixel in the output image is determined from the gray values of the corresponding input pixel in the various bands. For example we can decompose a color or monochromatic luminance (lightness) image from the gray values of the image in each of the three additive primary colors via:

\[
g[x, y] = \alpha f[x, y, \lambda_r] + \beta f[x, y, \lambda_g] + \gamma f[x, y, \lambda_b]
\]

\[
\equiv \alpha f_R[x, y] + \beta f_G[x, y] + \gamma f_B[x, y],
\]

where the coefficients \([\alpha, \beta, \gamma]\) are functions of the spectral filtration and sensitivity of the recording process. The gray values of the same pixel in the three images are generally different but correlated. For example, a red object will be bright in the red image and darker in the green and blue. The decomposition of a crude color image into its component \(RGB\) images is shown below; note that the histograms do not exhibit easily distinguishable clusters and thus it will be difficult to segment the objects from the image effectively.

Simple 3-band color image, the individual monochromatic images, and their histograms. Note that the gray values of the “tree” are approximately equal to those of “sky” in the red band and approximately equal to those of both “sky” and “grass.”
in the green band. For this reason, the tree is difficult to segment from a single color channel.

The three 1-D histograms do not exhibit distinct pixel clusters for each of the five object colors (red house, green tree, pale green grass, blue sky, and white clouds and door). In other words, the clusters of pixels belonging to these five classes overlap. This observation forces the conclusion that the five objects cannot be segmented successfully from a single band. That said, we can observe that the spectral reflectance of pixels belonging to the “house” is significantly different from the other objects, and those pixels may be segmented from the other objects fairly easily, (e.g., by a simple threshold in the blue image, as shown after selecting a threshold level of 112). However, the reflectance of the pixels belonging to the tree in the blue image is only slightly different from those belonging to sky. Attempts to segment pixels belonging to the “tree” from the blue image are shown, which lead to either a “noisy” segmentation, or the misidentification of the “tree” and the “grass.”

Two attempts to segment “tree” from blue image alone. The first sets all pixels to black in the range $112 \leq b \leq 140$, which produces a very noisy “tree.” Pixels in the second image are thresholded to black if they lie in the range $112 \leq b \leq 167$. The segmented pixels include the “grass.”
15.10 Multispectral Histograms for Feature Extraction

As already described, visible-light color images are represented by triads of monochrome images. The decomposition of color images into RGB bands is very common. Within the neural network of the eye, the three cone signals are weighted and combined to generate the three channels that are transmitted to the brain. Roughly speaking, one channel corresponds to the lightness or luminance of the scene (i.e., the black-and-white video signal), and is a weighted sum (integral) of $S, M,$ and $L$. This information is transmitted to the brain with full resolution. The other two transmitted signals are weighted differences (derivatives) of the $S, M,$ and $L$ and describe the chrominance of the scene; these are transmitted with reduced resolution, thus preserving information deemed more important during evolutionary development. Broadcast transmission of color video signals is roughly modeled on the weighted summing of cone signals for transmission to the brain.

In digital imaging, the three raw cone signals generated by the human visual system can be represented as three digital images: roughly the brightness in blue, green, and red light. The weighted summing of the cone signals for transmission to the brain may be modeled as a linear operation applied to the 3-element vector $[R, G, B]$. The operation is implemented by an invertible $3 \times 3$ matrix. The requirement that the matrix be invertible ensures that the 3-element output vector may be computed from the output vector. However, note that if the output values are quantized, as required before subsequent digital processing, then the cascade of forward and inverse transformations may not yield the identical triplet of $[R, G, B]$.

The variations of colors of the objects in an image may be used to segment a multispectral image $f [n, m, \lambda]$. To successfully segment the “tree” pixels from the color image, it is necessary to use the multispectral information simultaneously. One way is to use a multidimensional (2-D or 3-D) histogram generated from two or three of the gray values at each pixel. Often only two colors are used to generate a 2-D histogram because of the difficulty of displaying three dimensions of information by conventional means. For example, pixels having a particular gray level $f_R$ in the red image and a level $f_G$ in the green image are counted in bin $[f_R, f_G]$. The resulting matrix is the image of a 2-D feature space, i.e., the bin with the largest number of pixels can be displayed as white and unpopulated bins as black. The histogram can be used for image segmentation as before, but the extra information obtained from the second color usually ensures better results.
15.10 MULTISPECTRAL HISTOGRAMS FOR FEATURE EXTRACTION

15.10.1 2-D Histograms of Simple Color Image

The three 2-D histograms of each pair of channels of the three color bands. You should be able to identify the objects in each of the histograms. Four clusters can be identified in the R-G and five in the G-B histograms.

These 2-D histograms (also called “scatterplots”) of pairs of grayscale images were plotted using the Hypercube Image Analysis Software (available free for many platforms from http://www.tec.army.mil/Hypercube/).

The cluster of pixels with large gray values in all images due to the white pixels in the original are easily identified in the three 2-D histograms. You should be able to identify the clusters that belong to the house, tree, sky, etc. We can segment the image by thresholding those pixels within certain intervals of red, green, and blue (to white, say), and thresholding the others to black.

Note that histogram concept can be extended easily to more than three dimensions, though visual representation is more difficult. This is the basis for multispectral segmentation in many areas of image processing.

Segmentation from 3-D Histogram

It is often easier to identify distinct and well-separated clusters from the multidimensional histogram rather than from the individual images. The segmented image of the tree obtained from the 3-D histogram is:
Segmentation of “tree” pixels directly from the 3-D histogram. The black pixels satisfy the three constraints: $r \geq 106$, $g \geq 195$, and $b \leq 167$.

### 15.10.2 Principal Component Analysis – PCA

Reference: Schowengerdt, *Remote Sensing*

The information in the different bands of a multispectral image (e.g., an *RGB* color image) often are highly correlated, meaning that some or many bands may be visually similar. In other words, the information content of a multispectral image often is quite “redundant.” It often is convenient to reduce or even eliminate this redundancy by expressing the image in terms of “axes” other than the original “bands.” The data image is transformed to a new coordinate system by rigidly rotating axes to align with these other “directions” and the image data then projected onto these new axes. In principal components, the rotation produces a new multispectral image with a diagonal covariance matrix, so that there is no covariance between the various bands. One axis of the rotated coordinate system is aligned with the direction of maximum variance of the image, the second axis is perpendicular to the first and aligned with the direction of the second largest variance, etc. The bands in the principal component image are thus arranged in order from largest to smallest variance.

To illustrate, consider the principal components of a 2-band image created from the blue and green bands of the simple color image; the values in the red band were replaced with zeros. The 2-D histogram, blue vs. green, is shown to locate the (approximate) axes of the principal components, that were evaluated using “Hypercube.” Since the third component image was black, only two principal components are needed to fully represent the image. In the outputs, note that the “lightest” objects in the image (the white clouds and door) also are lightest in the first PC. The darkest structure in both channels is the house, which appears “black” in the 1st PC. The gray values are projected onto the orthogonal axis in the second PC image. The gray values of the red house pixels and the white pixels belonging to the clouds and door are projected to the same “mid-gray” value, and thus are indistinguishable in the 2nd PC (the door has “disappeared into the house”).
Principal components of a two-band image, computed in “Hypercube”: the image is created by inserting zeros in the red band. The 2-D histogram blue vs. green is shown. The first principal component projects the gray values onto the “cyan” axis that includes most of the image variance. The second principal component projects the data onto the magenta axis. The images are shown. Note that the first principal component shows the clouds and door as white and the house as black. The door is not visible in the second principal component, as its gray value is the same as that of the house.

The three PCs of the complete RGB image also are shown along with the three eigenvalues. The first PC of the 3-D scene has a very dark “sky” because it exhibits the largest contrast relative to the white objects (clouds and door). The 3rd PC shows the smallest range of variance, which is dominated by image noise.
The three principal components of the RGB house+tree image. Note that the third component image is quite noisy.

15.11 Time-Sequence Images: Video

The other common example a 3-D image plots time on the third image axis. Perhaps the most obvious use is in motion pictures, where the different frames of the movie are the time samples of the 3-D spatial image \( f[x, y, t_n] \). The illusion of continuous motion is created because of the photochemical response of the rods and cones in the retina. The time duration of the process is the source of the phenomenon of “persistence of vision.” The persistence is shorter if image is brighter (movie projectors have rotary shutters that show each frame two or even three times). Movies were originally taken at 16 frames per second, later increased to 24 fps in US. This is the reason why motion in old movies is too fast. The frame rate in Europe is 25 fps, related to the AC frequency of rate is 50 Hz. For this reason, American movies shown in Europe finish more quickly than they do in the US.

The second most familiar example of time-sampled imagery is video, where the 3-D scene \( f[x, y, t] \) is sampled in both time and space to convert the scene to a 1-D function of time \( s[t] \). It took 50 or more years to develop the hardware to scan scenes. The first systems were mechanical, based either on a system that scanned the light reflected from an illuminated scene, or an illumination system that scanned a beam of light over the scene and collected the reflected light. One of the primary developers of mechanically scanned video was the Scotsman John Logie Baird. The hardware of electronic scanning was developed through the efforts of such “illuminaries” as the American Philo T. Farnsworth, who demonstrated a working video system in the 1920s.

Video systems commonly use an “interlaced scan” that alternates scans of the even and odd lines of the full frame, so that the eye is presented with half of the image information every 1/60 s. This is less objectionable to the eye than the original “progressive-scanning” systems that presented a full frame every 1/30 s.

Note that lines number 248 to 263 and 511 to 525 are typically blanked to provide time for the beam to return to the upper left hand corner for the next scan; other signals (such as closed captioning or the second audio program) are transmitted during this “flyback” time.
Figure 15.1: Interlaced format of NTSC video raster: one frame in 1/30 second is composed of two fields, each taking 1/60 second. The first “field” includes 262.5 odd lines and the second “field” has 262.5 even lines. Lines 248-263 in the first field and lines 511-525 in the second field are “blanked” – this is the “retrace” time for the CRT beam and is the time when additional information (e.g., closed captioning) is transmitted.

15.11.1 Color-Space Transformations for Video Compression

References:

Falk, Brill, Stork, Seeing the Light, §10
Glassner, Principles of Digital Image Synthesis, §1

Particular color transformations have been developed for use in many different applications, including various schemes for image transmission. Consider the transformation used in the color video standard in North America, that was developed by the National Television Standards Committee (NTSC), which converts RGB values to three other channels by a linear transformation: the “luminance” $Y$ (used by “black-and-white” receivers), and two “chrominance” channels $I$ and $Q$ via:

$$\begin{bmatrix}
0.299 & 0.587 & 0.114 \\
0.596 & -0.274 & -0.322 \\
0.211 & -0.523 & 0.312
\end{bmatrix}
\begin{bmatrix}
R \\
G \\
B
\end{bmatrix}
= \begin{bmatrix}
Y \\
I \\
Q
\end{bmatrix}$$

Note that the sum of the weights applied to the luminance channel $Y$ is $0.299 + 0.587 + 0.114 = 1.0$, while the sums of the weights of the two chrominance channels are both 0. In other words, the luminance channel is a weighted sum of RGB (analogous to an integral), while the chrominance channels are weighted differences (similar to derivatives). In the context of linear systems, $Y$ is the result of “spectral lowpass filtering,” while $I$ and $Q$ are the outputs of what may be loosely described as “spectral highpass filters.”

If the input $R$, $G$, and $B$ are in the range $[0, 255]$, so will be the range of $Y$. Both chrominance channels of any gray input pixel (where $R = G = B$) is zero, and the range of allowed chrominances is bipolar and fills the available dynamic range,
e.g., \(-103 \leq I \leq 152, +133 \leq Q \leq -122\), totaling 256 levels for 8-bit RGB inputs. The positive polarity of \(I\) is reddish (often described as “orange”), and its negative polarity is “green+blue” or cyan; hence the \(I\) information is sometimes called the “orange-cyan” axis. The positive polarity of \(Q\) is “red+blue” or purple, and the negative polarity is green, so the \(Q\) information is the “purple-green” axis.

The transformation of a “mid-gray” pixel where the red, green, and blue images are identically \(\alpha\) is:

\[
\begin{bmatrix}
0.299 & 0.587 & 0.114 \\
0.596 & -0.274 & -0.322 \\
0.211 & -0.523 & 0.312
\end{bmatrix}
\begin{bmatrix}
\alpha \\
\alpha \\
\alpha
\end{bmatrix}
= 
\begin{bmatrix}
\alpha \\
0 \\
0
\end{bmatrix}
\]

so that the luminance is the gray value of the three colors and the two chrominance channels are both zero.

The transformation from \(YIQ\) back to \(RGB\) is the inverse of the forward matrix operator:

\[
\begin{bmatrix}
0.299 & 0.587 & 0.114 \\
0.596 & -0.274 & -0.322 \\
0.211 & -0.523 & 0.312
\end{bmatrix}^{-1}
\begin{bmatrix}
1.0 & 0.956 & 0.621 \\
1.0 & -0.273 & -0.647 \\
1.0 & -1.104 & 1.701
\end{bmatrix}
\]

\(\text{(rounded)}\)

\[
\begin{bmatrix}
1.0 & 0.956 & 0.621 \\
1.0 & -0.273 & -0.647 \\
1.0 & -1.104 & 1.701
\end{bmatrix}
\begin{bmatrix}
Y \\
I \\
Q
\end{bmatrix}
= 
\begin{bmatrix}
R \\
G \\
B
\end{bmatrix}
\]

Note that the \(R\), \(G\), and \(B\) channels all include 100% of the luminance channel \(Y\).

Other color transformations are used in other video systems. The two common systems used in Europe and Asia are \(PAL\) (“Phase Alternation by Line”) and \(SECAM\) (“Systeme Electronique Couleur Avec Memoire”). Each broadcasts 625 lines at 25 frames per second and uses the \(YUV\) triad of luminance and chrominance, where the luminance is the same combination of \(RGB\) but the chrominance channels are slightly different. The \(RGB\) transformation to \(YUV\) is:

\[
\begin{bmatrix}
0.299 & 0.587 & 0.114 \\
-0.148 & -0.289 & 0.437 \\
0.615 & -0.515 & -0.100
\end{bmatrix}
\begin{bmatrix}
R \\
G \\
B
\end{bmatrix}
= 
\begin{bmatrix}
Y \\
U \\
V
\end{bmatrix}
\]

The luminance calculations for \(YIQ\) and \(YUV\) are identical. The ranges of allowed chrominances are bipolar: for 8-bit \(RGB\) inputs \(-144 \leq U \leq 111\) and \(-98 \leq V \leq 157\) (each with 256 levels). The NTSC selected the \(YIQ\) standard over \(YUV\) because tests indicated that more \(I-Q\) data could be discarded without affecting the subjective
image quality.

Color image decomposition, all images displayed with full dynamic range (i.e., maximum possible gray value is mapped to white and minimum to black): first row RGB; second row YIQ; third row YUV.

Obviously there exists an infinite number of invertible $3 \times 3$ matrix transformations, and thus of invertible color transformations. The 3-D histograms of a particular input image before and after transformation generally will differ. Therefore segmentation of objects with particular similar colors likely will be easier or more successful in a particular color space.
15.11.2 Multiple-Frame Averaging

Consider a series of video or movie images of an invariant (i.e., stationary) object \(f[x,y]\) corrupted by additive noise which changes from pixel to pixel and frame to frame:

\[ g[x,y,t_i] = f[x,y] + n[x,y,t_i] \]

where \(n[x,y,t_i]\) is a random number selected from a Gaussian distribution with \(\mu = 0\). The additive noise will tend to obscure the “true” image structure \(f[x,y]\). One common problem in image processing is to enhance the visibility of the invariant objects in the image by attenuating the noise. If the gray values \(n\) are truly random, i.e., all values of \(n\) in the range \((-\infty, \infty)\) can exist with equal likelihood, then little can be done to improve the image. Fortunately, in realistic imaging problems the probability of each value of \(n\) is determined by some probability density function (histogram) \(p[n]\) and we say that the noise is stochastic. The most common probability distributions in physical or imaging problems are the uniform, Poisson, and normal. Less common, but still physically realizable, distributions are the Boltzmann (negative exponential) and Lorentzian. A general discussion of stochastic functions is beyond the scope of the immediate discussion, though we will go into more detail later while reviewing statistical filters. Interested readers should consult Frieden’s book *Probability, Statistical Optics, and Data Testing* (Springer-Verlag, 1991) for detailed discussions of physically important stochastic processes. At this point, we will state without proof that the central limit theorem determines the most common probability density function in physical problems is the normal distribution \(N[\mu, \sigma^2]\). The histogram of noise gray values in the normal distribution is:

\[ p[n] = \frac{1}{\sqrt{2\pi}\sigma^2} \exp \left[ -\frac{(n - \mu)^2}{2\sigma^2} \right] \equiv N(\mu, \sigma^2) \]

The normal distribution is completely characterized by two parameters: the mean value \(\mu\) and the variance \(\sigma^2\) (or its equivalent, the standard deviation \(\sigma\)). This is the equation for the familiar bell curve with maximum value located at \(n = \mu\) and with full width of \(2\sigma\) measured at approximately 20% of the maximum amplitude. In the special case where the mean value \(\mu = 0\), the normal distribution commonly is called a Gaussian distribution. The remainder of the discussion in this section will assume that the additive noise has been selected at random from a normal distribution.

It is probably clear intuitively that an image created by averaging a collection of noise images \(n[x,y,t_i]\) over time will tend toward a uniform image whose gray level is the mean \(\mu\) of the noise, i.e., the variations in gray level about the mean will “cancel out”:

\[ \frac{1}{N} \sum_{i=1}^{N-1} n[x,y,t_i] \cong \mu \cdot 1[x,y] \]

If the sequence of input images includes an invariant object on a background of additive normal noise, the visibility of the object will be enhanced in the average
image:
\[
\frac{1}{N} \sum_{i=1}^{N-1} (f[x, y] + n[x, y, t_i]) = \frac{1}{N} \sum_{i=1}^{N-1} f[x, y] + \frac{1}{N} \sum_{i=1}^{N-1} n[x, y, t_i]
\]
\[
\approx f[x, y] + \mu \cdot 1[x, y]
\]

This result is proven directly below, but may be seen more easily if the reader is familiar with the statistical concept that the probability density function of the sum of \( N \) random variables is the \( N \)-fold convolution of the individual probability density functions.

To quantify the visibility of the object in a noisy image, it is necessary to quantify the visibility of the noise, i.e., the variability of gray level due to the stochastic signal. The average gray value due to noise is:
\[
\langle n[x, y] \rangle = \int_{-\infty}^{+\infty} n[x, y] \, p[n] \, dn = \mu
\]

where \( \langle \cdot \rangle \) denotes the averaged value of the quantity and \( p[n] \) is the probability density function of the noise. The mean value obviously is not an appropriate measure because it does not describe the variability of gray value. A useful quantity is the variance of the noise which describes the average of the difference between the square of the gray value due to the noise and the mean:
\[
\sigma^2[n] = \langle (n - \mu)^2 \rangle = \int_{-\infty}^{+\infty} (n - \mu)^2 \, p[n] \, dn
\]
\[
= \int_{-\infty}^{+\infty} (n^2 - 2n\mu + \mu^2) \, p[n] \, dn
\]
\[
= \int_{-\infty}^{+\infty} n^2 \, p[n] \, dn - 2\mu \int_{-\infty}^{+\infty} n \, p[n] \, dn + \mu^2 \int_{-\infty}^{+\infty} p[n] \, dn
\]
\[
= \int_{-\infty}^{+\infty} n^2 \, p[n] \, dn - 2\mu \cdot \mu + \mu^2 \cdot 1
\]
\[
= \langle n^2 \rangle - 2\mu^2 + \mu^2
\]
\[
\sigma^2[n] = \langle n^2 \rangle - \mu^2
\]

A measure of the relative visibility of the signal and the noise is the ratio of the signal to the noise variance, and is often called the signal-to-noise ratio (S/N or SNR). It may be expressed as power or amplitude:

\[
\text{Power SNR} \equiv \frac{f^2[x, y]}{\sigma^2[n]}
\]
\[
\text{Amplitude SNR} \equiv \sqrt{\frac{f^2[x, y]}{\sigma^2[n]}} = \frac{f[x, y]}{\sigma[n]}
\]
After averaging $P$ frames of signal plus noise, the gray level at the pixel $[x, y]$ will approach:

$$g[x, y] = \frac{1}{P} \sum_{i=1}^{P} \left( f[x, y] + n[x, y, t_i] \right)$$

$$= \sum_{i=1}^{P} \frac{f[x, y]}{P} + \sum_{i=1}^{N} \frac{n[x, y, t_i]}{P}$$

$$\approx f[x, y] + \mu$$

The variance of the noise at pixel $[x, y]$ after averaging $P$ frames is:

$$\sigma^2[n] = \langle n^2 [x, y] \rangle - \mu^2 = \sum_{i=1}^{P} \left( \frac{n[x, y, t_i]}{P} \right)^2 - \mu^2$$

$$= \frac{1}{P^2} \sum_{i=1}^{P} n[x, y, t_i] \sum_{j=1}^{P} n[x, y, t_j] - \mu^2$$

$$= \frac{1}{P^2} \sum_{i=1}^{P} (n[x, y, t_i])^2 + \frac{2}{P^2} \sum_{i>j}^{P} n[x, y, t_i] \cdot n[x, y, t_j] - \mu^2$$

If the noise values are selected from the same distribution, all terms in the first sum on the right are identical:

$$\frac{1}{P^2} \sum_{i=1}^{P} (n[x, y, t_i])^2 = \frac{1}{P^2} \sum_{i>j}^{P} (\sigma_i^2 + \mu^2)$$

$$= \frac{1}{P^2} \left( P \cdot (\sigma_i^2 + \mu^2) \right)$$

$$= \frac{(\sigma_i^2 + \mu^2)}{P}$$

Because the noise values are uncorrelated by assumption, the second term on the right is just the square of the mean:

$$\frac{1}{P^2} \sum_{i=1}^{P} (n[x, y, t_i])^2 + \frac{2}{P^2} \sum_{i>j}^{P} n[x, y, t_i] \cdot n[x, y, t_j] - \mu^2$$

$$= 2 \cdot \left( \frac{\mu \cdot \mu}{2} \right) = \mu^2$$
The variance of the average image is:

\[ \sigma^2[n] = \frac{(\sigma_i^2 + \mu^2)}{P} + \mu^2 - \mu^2 \]

\[ = \frac{\sigma_i^2 + \mu^2}{P} \]

If the mean value of the noise is \( \mu = 0 \) (Gaussian noise), then the variance of the sum is reduced by a factor of \( P \) and standard deviation is reduced by \( \sqrt{P} \):

\[ \sigma^2[n] = \frac{\sigma_i^2}{P} \]

\[ \sigma[n] = \sqrt{\sigma^2[n]} = \frac{1}{\sqrt{P}} \sigma_i[n] \]

The amplitude SNR of the averaged image is:

\[ SNR_{out} = \frac{f[x,y]}{\sqrt{\sigma^2[n]}} = \frac{f[x,y]}{\left(\frac{\sigma_i[n]}{\sqrt{P}}\right)} = \sqrt{P} \cdot \frac{f[x,y]}{\sigma_i} = \sqrt{P} \cdot SNR_{in} \]

\[ SNR_{out} = \sqrt{P} \cdot SNR_{in} \]

Thus the effect of averaging multiple frames which include uncorrelated additive noise from a Gaussian distribution is to decrease the width of the histogram of the noise by a factor of \( \left(\sqrt{P}\right)^{-1} \), which increases the signal-to-noise ratio of the image by \( \sqrt{P} \). For example, a video image from a distant TV station often is contaminated by random noise (“snow”). If the image is invariant over time, its signal-to-noise ratio can be increased by averaging; if 90 frames of video (\( \approx 3 \) sec) are averaged, the SNR of the output image will increase by a factor of \( \sqrt{90} \approx 9.5 \).

If the noise is correlated to some degree from frame to frame (i.e., its spatial structure is partially correlated from image to image), then averaging will not improve the SNR so rapidly. For example, consider imaging of a submerged object through a water surface. Wave motion will distort the images but there will be some correlation between frames. The SNR might improve only as, say, \( P^{\frac{1}{4}} \approx 3 \) for \( P = 90 \) frames.
CHAPTER 15 POINT OPERATIONS

Averaging of independent noise samples: (a) signal \( f[x] \); (b) one realization of Gaussian noise \( n_1[x] \) with \( \mu = 0 \) and \( \sigma = 1 \); (c) \( f[x] + n_1[x] \); (d) \( \frac{1}{9} \sum_{i=1}^{9} (f[x] + n_i[x]) \), showing the improved signal-to-noise ratio of the averaged image.

15.11.3 Required Number of Bits for Image Sums, Averages, and Differences

If two 8-bit images are added, then the gray value of the output image lies in the interval \([0, 510]\), with 511 gray values requiring 9 bits of data to represent fully. If constrained to 8 useful bits of data in the output image, half of the gray-scale variations data in the summation must be discarded. In short, it is necessary to requantize the summation image. If two 8-bit images are averaged, then the gray values of the resulting image are in the interval \([0, 255]\), but half-integer values are virtually guaranteed, thus ensuring that the average image also has 9 bits of data unless requantized. The central limit theorem indicates that the histogram of a summation or average image should approach a Gaussian form.
15.12 Image Subtraction for Change Detection

Subtracting images of the same scene recorded at different times will highlight pixels whose gray value has changed in the interval:

\[ g[x, y] = f[x, y, t_1] - f[x, y, t_0]. \]

Invariant pixels will subtract to 0, pixels that have \{ brightened \} will have \{ positive \} gray level. This technique is applied to motion/change detection and may be interpreted as a discrete version of the time derivative.

\[ \frac{\partial f[x, y, t]}{\partial t} = \lim_{\Delta t \to 0} \frac{f[x, y, t + \Delta t] - f[x, y, t]}{\Delta t} \]

For multitemporal digital images, the smallest nonvanishing time interval \( \Delta t \) is the interval between frames (\( \Delta t = t_1 - t_0 \)). The time derivative is the difference image of adjacent frames:

\[ \frac{\partial f[x, y, t]}{\partial t} \to \partial_t \{ f[x, y, t] \} \equiv f[x, y, t_0 + (t_1 - t_0)] - f[x, y, t_0] \]

\[ = f[x, y, t_1] - f[x, y, t_0] \]

Note that the difference image is bipolar, white > 0, gray = 0, black < 0. The difference image \( g[x, y] \) is bipolar and must be scaled to fit the available discrete dynamic range \([0, 255]\) for display. Often \( g_0 = 0 \) is mapped to the mid-level gray (e.g., 127), the maximum negative level is mapped to 0, the brightest level to 255, and the intervening grays are linearly compressed to fit.

\[ f[x, y, t_j] \quad f[x, y, t_2] \quad f[x, y, t_2] - f[x, y, t_1] \]

\[ t_2 \approx t_1 + 3 \text{ sec.} \]

**Difference of two images of the same scene taken approximately 3 seconds apart. The images are from a movie taken by James Noel during the Mallory Everest Expedition of 1924. The one image has been translated by a small distance, thus**
creating “edge” regions. Differences of zero are mapped to “midgray”, while regions where the second image is brighter or darker map to white and black, respectively.

In this example, the difference image may be used to help determine how well registered the two images are; if perfectly registered, only pixels that have really changed between images will have gray values other than 0.

### 15.13 Difference Images as Features

In feature extraction and recognition applications (e.g. remote sensing), linear combinations (i.e. sums and differences) of multispectral imagery may supply additional useful information for classification. For example, the difference image of spectral band 3 (red) and 4 (infrared) (out of a total of seven bands) imaged by LANDSAT helps classify urban vs. rural features in the image. In the simple house-tree image, the visibility of the house is enhanced in Red-Green and Blue-Red images.

![Color difference images: the “house” is more obvious in the green-red image, and the tree in the red-blue image.](image)

Note that the difference images are noticeably noisier, especially Blue-Green; difference images enhance all variations in gray value, whether desired or not. The concerns about displaying bipolar gray values of time derivatives exist here as well.

#### 15.13.1 Number of Bits in Difference Image

If two 8-bit images (with gray values $f_n$ satisfying the constraint $0 \leq f_n \leq 255$) are subtracted, then the gray values of the resulting image $g[x, y]$ lie in the range $-255 \leq g \leq +255$, for a total of 511 possible values requiring 9 bits of data. To obtain 8 useful bits of data in the output image, half of the data in the difference image must be discarded (usually the least-significant bit, though if the pixels are well correlated, then the most-significant bit may be zero).

The principles of statistical distributions tell us that the difference of images with data generated from two statistical distributions is the convolution of the two distributions centered about the difference in the mean values.
15.14 “Mask” or “Template” Multiplication

Pixel-by-pixel multiplication is an essential part of local neighborhood and global image operations, e.g., the Fourier transform. It also is useful to mask out sections of an image, perhaps to be replaced by objects from other images. This is occasionally useful in segmentation and pattern recognition, but is essential in image synthesis, such as for special effects in movies.

15.15 Image Division

Images recorded by a system with spatially nonuniform response are functions of both the input distribution \( f[x, y] \) and the spatial sensitivity curve \( s[x, y] \), \( 0 \leq s[x, y] \leq 1 \):

\[
g[x, y] = f[x, y] \cdot s[x, y]
\]

This is a deterministic multiplicative degradation of the image; the image may be restored by dividing out the noise. An estimate of the true image brightness can be computed at each pixel by division:

\[
\hat{f}[x, y] = \frac{g[x, y]}{s[x, y]} \approx f[x, y]
\]

(n.b., no image information is recorded at pixels where \( s[x, y] = 0 \) and thus the true value cannot be recovered at those pixels). This technique has been applied to remotely sensed imagery where information about image brightness is critical. Note that errors in the sensitivity function greatly distorts the recovered signal. Similarly, additive noise creates big problems in image division.

15.15.1 Image Division to Correct Spatial Sensitivity

Imaging systems often suffer from a consistent multiplicative error. One very common example is the variation in sensitivity of the pixels in a CCD sensor due to the intrinsic variability in the substrate properties or manufacturing. These multiplicative errors are measured and corrected via a subsequent division.

Consider a biased sine wave \( f[x] \) recorded by an imaging system whose sensitivity falls off away from the origin. The image may be recovered completely if the sensitivity curve is nonzero everywhere and there is no noise in either the recorded signal or the estimate of the sensitivity.
1-D simulation of spatial compensation for sensitivity correction: (a) original signal \( f[x] \) is a biased sinusoid; (b) sensitivity function \( s[x] \); (c) \( g[x] = f[x] \cdot s[x] \); (d) correction \( \hat{f}[x] = \frac{g[x]}{s[x]} \).

Noise in the estimate of the sensitivity results in distortion of the recovered signal; the effect is more severe where the SNR is low. The deviation of the added noise is 0.005.
Spatial correction in presence of noise: (a) $g[x] + n[x]$, where $n[x]$ is zero-mean Gaussian noise with $\sigma = 0.005$; (b) $\hat{f}[x] = \frac{g[x] + n[x]}{s[x]}$, showing the large errors due to incorrect division of two small values.

15.15.2 Image Division to Enhance Low-Contrast Imagery

Astronomers would often need to examine fine structure (i.e. significant brightness differences) that are hidden by a larger-scale brightness gradient of the object. Included among the significant low-contrast structural features are streamers in the solar corona that radiate outward from the solar surface; the brightness details provide clues about the physical nature of the solar atmosphere and magnetic fields. However, the radial brightness gradient of the corona makes it very difficult to image the full length of the coronal streamers. The overall dynamic range of ground-based imagery of the solar corona is approximately three orders of magnitude (limited by atmospheric sky brightness). Imagery from air- or spacecraft may add an order of magnitude for a total dynamic range of $10^4$. This may be recorded on a wide-range photographic negative ($\Delta D > 4$), but it is not possible to print that dynamic range by normal photographic techniques.

The problem of enhancing the visibility of small changes in coronal image brightness is similar to correction of detector sensitivity just considered and may be attacked by digital methods. The brightness gradient of the corona is analogous to the 1-D sensitivity function $s[x]$; division of the original image by the brightness gradient “equalizes” or “flattens” the gray-level variations in the background, thus allowing the smaller-scale variations across the image to be displayed on a device with limited dynamic range. An estimate of the coronal brightness gradient may be determined by averaging fitted curves of the radial brightness or by making a low-resolution (blurred) image. The latter is more commonly used, as it may be derived via simple local neighborhood digital operations to be considered next. The recovered image is the ratio of the measured high-resolution image and the image of the brightness gradient. This technique may be applied to archival photographic negatives of historical eclipses to provide additional information about the history of solar conditions and thus their effects on the earth’s climate.
Examples of Image Division for Local Contrast Enhancement

Examples of recovered information from old imagery of a solar eclipse and a comet are shown. The pictures are from *Enhancement of Solar Corona and Comet Details* by Matuska, *et al.*, *Proc. SPIE*, 119, pp. 28-35, 1977 and in *Optical Engineering*, 17(6), 661-665, 1978. The images are scans of xerographic prints from microfilm, which is the reason for the poor quality (courtesy of Wallace Memorial Library!)

*Average of four “raw” images of the 1973 solar eclipse, showing the corona with little discernable detail.*

*Image of solar corona obtained by “pseudounsharp masking,” this is the ratio of the original average of four images image and a lowpass-filtered replica.*

*Image of solar corona obtained by dividing the original image by a lowpass-filtered replica and following with highpass filtering to “sharpen” the fine detail.*
Original image of comet 1957V (Mrkos)

Enhanced image of Comet Mrkos via “pseudounsharp” masking.