Humans rely heavily on the sense of sight to collect their own data, which are then transformed into information by the brain; that information forms the basis of human judgements and actions.

Nonetheless, recording and machine processing of images are relatively recent accomplishments. Although photography dates back to the latter half of the nineteenth century, it was not until the early 1920s that one of the first techniques for transmitting a digital image appeared. The Bartlane cable picture transmission system reduced the time it took to send a picture across the Atlantic Ocean from a week to less than three hours. The picture was converted to electronic impulses by special encoding, transmitted across the transatlantic cable, and reconstructed at the receiving end.

Digital image processing as we know it today began in 1964 at the Jet Propulsion Laboratory in Pasadena, California, when, as part of the United States space program, scientists began using digital computers extensively to correct for camera distortions in digital images of the moon. Since then, the scope of applications for digital image processing has broadened enormously. It now includes automatic recovery of properties of objects for the purpose of human interpretation, as well as enhancement of images for visual interpretation by a human viewer. Image processing has played and will continue to play a part in many advanced technological innovations simply because visual information is so rich and multifaceted.

Engineers and computer scientists have developed a wealth of algorithms to restore clean images from noisy ones, to segment images into homogeneous regions, to extract important features from those images, and to reconstruct three-dimensional scenes from two-dimensional slices or projections. The image processing systems that address these tasks have typically used quite simple statistical methods, although there are alternative approaches that employ statistical methods for spatial processes* to obtain optimal solutions.

This entry will be concerned exclusively with digital image processing. An image $X$ is an $M \times N$ rectangular array

$$X \equiv \{X(m,n) : m = 0,\ldots,M - 1; n = 0,\ldots,N - 1\}$$

of picture elements, or pixels. At a pixel location $(m, n)$ one observes a pixel value $X(m, n)$ that can take any one of $K$ (e.g., $K = 256$) possible intensity or gray values. For example, $X$ might be a Thematic Mapper satellite image with a 30 meter $\times$ 30 meter pixel resolution whose gray values are the (discretized) registered intensities in a given band of the electromagnetic spectrum. A grid of such scenes, suitably interpreted, can provide an efficient way to estimate crop yields or to monitor the effects of desertification, forest clearing, erosion, and so forth. Medical imaging offers another important example, where noninvasive diagnostic methods such as magnetic resonance imaging (MRI) provide an image $X$ of metabolic activity in selected parts of the body (e.g., Cho et al. [9]).

The image $X$ could be thought of as an $MN$-dimensional, multivariate statistic. The thousands of remotely-sensed scenes recorded every week lead to massive, multivariate data sets whose statistical analyses can be overwhelming; modern society's ability to collect and store data has outstripped its ability to analyze it. Yet, humans are remarkably adept at image processing within their own vision systems, albeit with considerable person-to-person variability. One might argue that digital image processing is an attempt to remove that subjectivity from what is recognized as a very powerful method of spatial data analysis. Interestingly, this has produced many methods of manipulating image data that do not have a known equivalent in human vision.

As a consequence of the computer's great facility to process data, the processing of image data has become widespread [43]. This is not restricted solely to the visible spectrum of electromagnetic wavelengths. Sensors that detect infrared, ultraviolet, X-rays, and radio waves also produce images, as do signals
other than electromagnetic, such as low frequency acoustic waves (sonar images) and photon emission (single photon emission computed tomography, or SPECT).

This entry is meant to introduce briefly the area of statistical processing of image data within the context of a large and rapidly expanding engineering literature. Most topics receive at least a brief mention.

**HARDWARE OF AN IMAGING SYSTEM**

The hardware for a general imaging system consists of an image acquisition device, an analog-to-digital signal converter, a digital computer upon which the image processing is carried out, a storage device for image data, and an output device such as a printer or camera.

Image data result from spatial sampling of a (conceptually) continuous process. The sampled values represent some physical measurement, such as light intensity (as in photographic images) or the magnitude of signal returned from the interaction between sound and tissue (as in ultrasound images). The sensors that collect the sampled data are often arranged in a rectangular array, whose vertical and horizontal spacing each determines a Nyquist frequency\(^*\) above which aliasing occurs \[26\]. To deal with this, it is assumed that the image is bandlimited, i.e., its Fourier transform is zero outside a bounded region of support.

A common way to acquire an image is through a standard video camera. There are many sensors in a rectangular array (in 1994, as many as 300,000/cm\(^2\)), each one counting the number of photons that impinge upon it. These sensors, or charge coupled devices (CCDs) as they are known, define the pixels of the digital image. The number of photons is then translated into an analog voltage signal (via Ohm’s law: voltage = current × resistance), which is then passed out of the array to a device that converts the analog signal to a digital signal. An analog-to-digital converter (ADC) samples the continuously valued analog intensity signal and produces the one discrete value from a finite set of discrete values that is closest to the analog value. Typical sets for these “intensity” or “gray” values are \{0, \ldots, 2\(b\) – 1\}, where \(b\) is the number of bits; for example, for \(b = 8\) bit data, there are \(K = 256\) possible gray values. The resulting pixel value from a given CCD represents an averaging of the signal across a small finite area of the scene. Spatial resolution is improved by increasing the number of CCDs per unit area but space between CCDs is needed for electrical isolation and to prevent scattering of light across neighboring CCDs \[43\].

Eight bits, or one byte, of data per pixel correspond nicely to the usual ways in which data are stored in computers. However, nonlinearities in the ADC, electronic noise from the camera itself, and degradation over time of the electronic circuitry, can all combine to make the two least significant bits of most standard video images useless. This results in 64 useful gray levels from the original 256, although all the 256 gray levels are retained for further processing and display. When the image data (stored digitally in a computer or a computer storage device) is recalled, a digital-to-analog converter produces voltages that are most commonly displayed using a cathode ray tube (CRT). This reconstruction process is relatively noise-free. The CRT displays a monochrome image consisting of black (0) to white (256) with varying shades of gray in between. Because the human eye can visually distinguish only 30 or so gray levels \[20\], 256 levels is more than adequate for those applications where the human eye will judge the quality of an image.

Limitations in the way CCDs can count photons or electronic noise in circuitry, cabling, and the ADC can result in noisy image data that differ from the true pixel values of the original scene (the signal) that would have been produced had there been no distortion. The ratio of the variability in the signal to that in the noise is the signal-to-noise ratio (SNR). When SNR is high, the signal is typically easily discernible to the eye; when SNR is low, small features that have few pixels are often invisible to the eye. Removing noise is an important step in image processing. Unfortunately, it is often impractical to collect many images of the same scene and use averaging to remove the noise. Thus noise removal is usually done on an image-by-image basis.
In the sequel we focus on the mathematical and statistical methods used in image processing. Nevertheless, an understanding of the process of image acquisition, as described briefly above, is important for designing efficient image processing algorithms.

PROCESSING OF IMAGE DATA

While the distinction is not universal, the terms image processing and image analysis often have a different meaning in the engineering literature. Image processing usually refers to operations on an image whose result is another image (e.g., noise removal); such operations are sometimes called filters. Image analysis usually refers to the extraction of summary statistics from an image (e.g., area proportion of a given phase in a multiphase image).

This section outlines the steps that typically make up the processing of digital image data; see Fig. 1. For any particular image, the actual steps followed depend on the type of data and what goal the user has in mind.

In Fig. 1, the processing algorithms performed within the dotted box are sometimes referred to as computer vision algorithms rather than image processing algorithms. Here, the information extracted is a description, interpretation, or representation of a scene. This leads to the more general notion of a computer vision system (e.g., Ballard and Brown [3]), a part of which is the image processing system. Below the top four boxes in Fig. 1 are two boxes whose end result is a verbal or symbolic representation of scene content. For example, upon analysis of a scene, a computer vision system could output the description: “The field of view contains crop land of corn, soybeans, and wheat in the following percentages . . .”

The role of statistics, particularly spatial statistics, in image processing and image analysis has been small but its importance is beginning to be realized [5,6,22]. Statistics brings to the image-processing literature the notion of optimality of the algorithms that restore, segment, reconstruct, or extract features of images in the presence of uncertainty. That uncertainty, or variability, can arise in a number of ways, such as noise generation of the type previously described.

More controversially, even a “noise-free” image may be regarded as a single realization from an ensemble of possible images. In this case, the variability is derived from decisions as diverse as choice of scene, physical conditions of the medium in which the imaging takes place, and the digitization. These influences have led to an image processing approach based on Bayesian inference and statistical modeling. That is, a prior distribution \( \pi(\theta) \) is specified for the “true” pixel intensity values \( \theta \). This, along with \( f(x|\theta) \), the probability distribution of the noisy image \( X \) given \( \theta \), allows the posterior distribution \( p(\theta|x) \) of \( \theta \) given \( X=x \) to be calculated (in principle) via Bayes’ Theorem:

\[
p(\theta|x) = \frac{f(x|\theta)p(\theta)}{\sum_{\eta}f(x|\eta)p(\eta)}.
\] (1)

The posterior \( p(\theta|x) \) represents current uncertainty about the true image in light of having observed \( X=x \), a noisy version of it. Should a second, independent observation \( Y=y \) be taken of the same image, \( p(\theta|x) \) is updated to reflect the now current uncertainty through

\[
p(\theta|x,y) = \frac{f(y|\theta)p(\theta|x)}{\sum_{\eta}f(y|\eta)p(\eta|x)}.
\]

As more and more observations are taken, the posterior probability distribution reflects
Image restoration corresponds to prediction of the true image \( \theta \). Suppose one incurs a “loss” \( L(\theta, \delta(x)) \) when the true value is \( \theta \) but the predictor \( \delta(x) \) is used. Then the optimal image \( \delta^*(x) \) is defined to be the predictor that minimizes the Bayes risk, or \( E[L(\theta, \delta(X))] \), where the expectation is taken over both \( \theta \) and \( X \) (see Decision Theory). In fact \( \delta^*(x) \) minimizes

\[
\sum_{\theta} L(\theta, \delta(X)) p(\theta|x)
\]

and hence \( \delta^* \) depends on \( p(\theta|x) \), the posterior distribution of \( \theta \) given \( X = x \). Intuitively speaking, the loss function reflects the relative penalties the image analyst wishes to place on various reconstructions \( \delta(x) \) in relation to the true image \( \theta \). The choice of loss function may well depend on the goal of the reconstruction (e.g., low misclassification rate, high contrast, differential importance given to subregions of the image).

In general, changing the loss function will lead to a different optimal restoration. If the 0–1 loss function

\[
L(\theta, \delta(X)) = \begin{cases} 0 & \text{if } \theta = \delta(x), \\ 1 & \text{if } \theta \neq \delta(x) \end{cases}
\]

is used, then \( \delta^*(x) = \arg \max_{\theta} p(\theta|x) \), which is the maximum a posteriori (or MAP) estimator. The MAP estimator is simply the mode of the posterior probability density (or mass) function of \( \theta \). If the images \( \theta \) and \( \delta(x) \) are regarded as vectors of pixels and the squared-error loss function

\[
L(\theta, \delta(X)) = (\theta - \delta(x))^T(\theta - \delta(x))
\]

is used, then \( \delta^*(x) = E[\theta|X = x] \), the posterior mean of \( \theta \).

Probably the most difficult task in Bayesian image processing is the choice of prior \( \pi(\theta) \). Because the pixel gray values \( \{\theta(m, n) : m = 0, \ldots, M-1; n = 0, \ldots, N-1\} \) are in a spatial array, a prior that quantifies the idea that nearby pixel values tend to be more alike than those far apart is desired. Markov random fields, expressed in their Gibbsian form, offer a very attractive class of prior models [4,16]. Briefly, two pixel locations \( u \neq v \) are said to be neighbors if \( \Pr[X(u) = x(u)|X(s) = x(s) : s \neq u] \) depends functionally on \( X(v) \). A clique is a set of pixel locations, all of whose elements are neighbors of each other. Then a Markov random field expressed in its Gibbsian form is

\[
\pi(\theta) \propto \exp \left( - \sum_{\kappa \in C} V_\kappa(\theta_\kappa) \right),
\]

where \( \kappa \) is a clique, \( \theta_\kappa \equiv \{\theta(s) : s \in \kappa\} \) and the summation is over the set of all cliques \( C \). The potential energy functions \( \{V_\kappa : \kappa \in C\} \) are chosen by the user to generate prior probabilities of various (local) configurations of gray values. For example, unlikely configurations can be downweighted by setting \( V_\kappa(\theta_\kappa) \) to be large for values of \( \theta_\kappa \) that give rise to those configurations.

Bayes’ Theorem using a Markov random field prior \( \pi(\theta) \) and a local noise distribution
RESTORATION AND NOISE REMOVAL

When an image is acquired, it often comes with defects due to the physical limitations of the process used to acquire the data, such as sensor noise, poor illumination, blur due to incorrect camera focus, atmospheric turbulence (for remotely-sensed data), and so on. Image restoration refers to the processing of the image data so as to "restore" the observed image back to its true, noise-free state.

Many linear operations have been developed expressly for noise removal in image processing; refs. [20,26,36] describe the better known ones. On occasions, the system noise itself is known or can be approximated. For example, in photoelectronic systems, the noise* \( \delta \) in the electron beam current is often modeled [26] as

\[
\delta(m,n) = g(m,n)^{1/2} \epsilon_1(m,n) + \epsilon_2(m,n),
\]

where \((m, n)\) is the pixel location, \(g\) is the signal of the scanning beam current, and \(\epsilon_1\) and \(\epsilon_2\) are zero-mean, mutually independent, Gaussian white-noise processes. The dependence of the noise \(\delta\) on the signal \(g\) is because the detection and recording processes involve random electron emissions assumed to have a Poisson distribution* with mean \(g\). Based on this model, filters can be designed to remove the noise.

Assumptions about the signal usually involve some form of piecewise smoothness over the image domain. For example, discontinuities or change-points in intensity is an appropriate assumption when the image is one of distinct objects, within which a smooth function can be used for modeling the objects’ texture. This is illustrated by Korostelev and Tsybakov [28], who apply multidimensional nonparametric estimation techniques to obtain minimax estimators of a true, piecewise-smooth image intensity. Grenander and Miller [22] take a Bayesian viewpoint, but their models have this same idea of a true image made up of textured objects with sharp boundaries.

On occasion, such as for real-time video sequencing, multiple images of the same scene are available. Averaging of these images can help reduce noise that is random in nature, thus increasing the signal-to-noise ratio.

While linear operations work fairly well for many types of white noise, they do not work well for additive noise that is "spiky" or impulsive in nature (i.e., having large positive or negative values in a spatially compact area). However, the median filter, which is a special case of a rank or order statistic filter [31], does work well on this type of noise. To apply the median filter, choose a local pixel neighborhood, such as a \(3 \times 3\) or a \(5 \times 5\) neighborhood, to act as a window or mask. Then place the mask over the image at each pixel location and compute the median value for the 9 or 25 pixels, respectively, inside the mask. The output image has, at the central location, the median value. In Fig. 2(b), we show the result of applying the \(3 \times 3\) median filter to the grey-level input image as shown in Fig. 2(a). The median filter is related to a broad class of nonlinear imaging operations that come from an area known as mathematical morphology.

Mathematical morphology unifies the mathematical concepts in image-processing and image-analysis algorithms that were being applied in the 1970s to such diverse areas as microbiology, petrography, and metallography [33,45,46]. More recently, a comprehensive image algebra for digital images has been developed; it includes discrete mathematical morphology as a subset [40,41].
Mathematical morphology offers a powerful and coherent way of analyzing objects in two or more dimensions [45,47]. Its highly nonlinear operations were first developed for shape analysis of binary images, but the approach extends to noise removal, connectivity analysis, skeletonizing, and edge detection, to name a few. The mathematical morphological operation of dilation and its dual operation of erosion transform an image (the data) through the use of a structuring element or template.

The template provides a reference shape by which objects in the image can be probed in an organized manner. For the usual input image $X$ let the template be defined by $\{B(i,j): (i,j) \in T\}$, where $(i,j) \in T$ implies $(-i,-j) \in T$ and $T$ typically has many fewer elements in it than the $MN$ pixels of $X$. Then the dilation and erosion of $X$ by $B$ are given, respectively, by

$$C(m,n) = \max \{X(m-i,n-j) + B(i,j): (i,j) \in T\},$$
$$D(m,n) = \min \{X(m-i,n-j) - B(-i,-j): (i,j) \in T\}.$$ 

While these two operations are themselves simple, cascades or combinations of such operations, including set-theoretic ones like set complementation and union (for the binary case), can lead to complex and flexible algorithms for image processing.

The discrete Fourier transform (DFT) and its numerous fast versions are widely used in image processing, particularly for noise reduction [26]. The two-dimensional DFT of an $M \times N$ image $X$ is another $M \times N$ image, defined by

$$\hat{X}(i,j) = \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} X(m,n) \times \exp \left[ - \left( \frac{2\pi im}{M} \cdot \frac{\sqrt{-1}}{M} \right) + \frac{2\pi jn}{N} \cdot \frac{\sqrt{-1}}{N} \right],$$

where $i = 0, \ldots, M-1$ and $j = 0, \ldots, N-1$. A commonly-used image processing filter is to calculate the DFT of an image and then to select a range of $i$ and $j$ for which $\hat{X}(i,j)$ is modified to be zero. Filters that pass only the low frequency values, in which $\hat{X}(i,j)$ is modified to be zero for large values of $i$ and $j$, are low-pass filters, while those that pass only high frequency values are high-pass filters. The DFT is a member of a larger class of operations called unitary transforms [36]. Here, the image is represented in terms of basis vectors for the $MN$-dimensional space; such a representation can be useful in compression and filtering of image data, feature extraction, and other methods of image processing. As the DFT itself is computationally demanding, much effort has been expended to develop fast Fourier transforms for these problems [6].

Reconstruction of image data from projections is required when energy is emitted from a source as a beam and collected at the end of the path by a sensor. The only information given directly by this imaging-acquisition system is the amount of energy being emitted from the source and the sum of the energy as it hits the sensor, after it has gone through the object under study. By rotating the source-sensor assembly around the object, projection views for many different angles can be obtained. Systems that...
acquire data in this manner are computerized tomography, or CT, systems. The goal of these image reconstruction problems is to reconstruct the interior of the object from the projection information collected; the principal mathematical tool used is the Radon transformation\(^*\) [17, 39, 49].

**IMAGE ENHANCEMENT**

Image enhancement generally refers to the accentuation of edges or gray-level contrast in order to make the data more useful for analysis or display. The distinction between image enhancement and noise reduction is not always clear. For example, the removal of unwanted artifacts with less than a pre-specified maximal diameter could be viewed as either.

Consider the local averaging operator, where pixel values in a window or mask are averaged, and the new averaged value replaces the old pixel value at the center of the mask. Figure 3 shows an example of a 3×3 averaging mask, where the weights used in the local averaging are superimposed on the pixel locations in the mask. When the new averaged values instead of the original values are used to calculate the new output values, this gives a moving average. Local averaging is equivalent to a low-pass filter and it tends to blur edges in the image. High-pass filtering is more appropriate for image enhancement; because edges contain mostly high frequency components, it results in a sharpening of edges in the spatial domain. Gonzalez and Woods [20] provide a good description of these techniques.

Manipulation of the contrast in an image is another commonly used enhancement scheme. Here, the gray levels in an image are changed so that contrasts between regions stand out to the eye. Histogram equalization (see, e.g., Jain [26]) provides a way to do this. The histogram\(^*\) of an image is a graph of the relative frequency of occurrence of the gray levels in the image. In histogram equalization, the goal is to transform the original histogram into a uniform one. The original gray values are then mapped to the new gray values according to the transformation. If the gray value \(X(m, n)\) is considered to be a non-negative random variable with probability mass function \(f(x)\) and cumulative distribution function \(F(x)\), then

\[
U(m, n) \equiv F(X(m, n)) = \sum_{x=0}^{X(m,n)} f(x)
\]

has, by the probability integral transformation\(^*\), a discrete uniform distribution over (0, 1). In practice, the histogram of relative frequencies replaces \(f(x)\), and some discretization of \(U(m, n)\) is needed to convert it to the new gray values.

When applying histogram equalization to an image, an output image is produced that stretches the low-contrast gray levels in an image. This results in an image with a flat histogram and increased contrast. If this does not produce desired results, then a histogram transform other than the uniform can be tailored by the user. Histogram equalization can be applied at any time in the processing sequence to provide increased contrast to the image data.

**SEGMENTATION**

Segmentation is usually performed after restoration and enhancement. Segmentation is a broad term describing procedures that break the image up into regions having different properties, as desired by the user. Here we use the terms segmentation and classification synonymously although, on occasion,
classification is used to refer to the regional description following segmentation.

Segmentation of pixels in an image can be viewed from two perspectives: the decomposing of an image into regions based on similarities within each region, such as through topological or gray-value properties; or the separating of an image into regions based on differences between them, such as through edge or boundary detection. Using the first approach, an image can be separated into regions where, for example, the gray values in each region are in a narrow range of values. Approaches such as $k$-nearest neighbor classification [14], decision-tree classification [1], or artificial neural networks* [23] will achieve this goal.

For the first approach, a Bayesian discriminant analysis* is possible. In the simplest form, it classifies each pixel of the image without regard to the intensities of surrounding pixels. Suppose there are $H$ possible classes to which a pixel might belong and let $(\pi(k) : k = 1, \ldots, H)$ denote the prior probability of any pixel belonging to that class. At pixel $(m, n)$, the true class is $\theta(m, n)$ and the observed intensity is $X(m, n)$. From the noise distribution, given by density $f(x(m, n)|\theta(m, n))$, and the prior, $\pi$, a Bayes classification* rule declares $\hat{\theta}(m, n) = k^*(x(m, n))$, where $k^*(x(m, n))$ maximizes the posterior distribution

$$p(k|x(m, n)) = \frac{f(x(m, n)|k)\pi(k)}{\sum_{\ell=1}^{H} f(x(m, n)|\ell)\pi(\ell)}$$

with respect to $k$. Then $\{\hat{\theta}(m, n)\}$ is a segmentation of the image. The segmentation tends to be very patchy, which has led to the development of contextual (spatial) classification techniques that account for neighboring pixel values in the classification criteria ([34], Chapter 13).

With binary images (i.e., pixel values taking one of two possible values, namely, zero and one), the topological property of connectivity is a useful tool for segmentation. Two common connectivity notions used in image processing are four-connectivity and eight-connectivity [20]. Let $X(m, n)$ and $X(k, l)$ represent two pixels in a binary image on an $M \times N$ rectangular array. The two pixels are four-neighbors if (1) $X(m, n) = X(k, l) = 1$, and (2) $0 < [(m-k)^2 + (n-l)^2] \leq 1$. A four-path from $X(m_1, n_1)$ to $X(m_k, n_k)$ is a sequence of pixels $X(m_1, n_1), X(m_2, n_2), \ldots, X(m_k, n_k)$ such that $X(m_i, n_i) = 1$ for $i = 1, \ldots, k$ and $X(m_i, n_i)$ is a four-neighbor of $X(m_{i+1}, n_{i+1})$ for $i = 1, \ldots, k - 1$. A region is four-connected if each pair of pixels in that region has a four-path between them. Similarly, two pixels $X(m, n)$ and $X(k, l)$ are eight-neighbors if they satisfy (1) and (2): $0 < [(m-k)^2 + (n-l)^2] \leq 2$. This latter condition simply allows pixels on the diagonals to be considered possible eight-neighbors. Eight-paths and eight-connectivity within a region are defined analogously.

Fast algorithms are available to determine regions of the image that are connected in various ways [2]. An important connectivity scheme used in image processing is eight-connectivity for the white or background pixels (those with gray values of 0), and four-connectivity for the black or object pixels (those with gray values of 1). This avoids paradoxical connectivities that can occur in image processing, such as in a region of the image where sharp corners of two objects almost touch; it should not be possible that both background and object be connected in that region. For a more detailed discussion on digital topology, see Kong and Rosenfeld [27].

The second approach to segmentation is through boundary detection. Here the image is segmented by specifying the boundaries that separate them; for example, the goal may be to segment so that gray values in each region are approximately constant. More generally, regions that have certain texture, shape, or feature properties can also be used to segment an image. One can distinguish between regions in an image through such summary statistics as first- or second-order moments from the normalized histogram, spatial moments of regions, ranges of gray values, frequency content, and connectivity measures for each region [26].

The problem of boundary detection, or edge detection, can be solved using one of two general approaches: difference operators or parametric models. An edge pixel is one whose neighboring pixels have a wide variation in gray value from the given pixel’s gray value. A difference operator uses local information about the gradient at a pixel
location to determine whether that pixel is an edge pixel or not. If the estimated gradient is beyond a specified level, the pixel is declared an edge pixel. Edge detectors include the Sobel, the Robert, the Prewitt, the Kirsch [20], the Marr-Hildreth [32], and the Canny [7] difference operators.

The Marr-Hildreth and Canny operators are examples of multi-scale edge detectors. These smooth the image data with a convolution, typically Gaussian, followed by a detection of the zero-crossings of the second derivative of the smoothed image data. Performing this at different values for the variance parameter in the Gaussian convolution, that is, at different “scales," and then combining the results, can result in the detection of boundaries. Although many difference operators are isotropic, with known limitations due to both the isotropy and sensitivity to noise, they nonetheless yield a popular class of edge detectors. See Torre and Poggio [48] for a unified framework for edge detectors based on difference operators.

Another way to detect edges is to use an edge model and compute its degree of match to the image data. While such detectors are usually more computational, they can give better results and more information for further processing. The Hueckel [25], Hough [13], and Nalwa-Binford [35] edge detectors are algorithms based on fitting the image data to a model of the edges, which are fit by a criterion like least-squares*. The Hueckel edge detector models an edge as a step or ramp function and the Nalwa-Binford edge detector uses a surface described by a piecewise polynomial. The Hough edge detector is actually a more general curve-finding transform that can be used for the analysis of shapes in images. However, there are computational sensitivities associated with the Hough transform that complicate practical implementations [37].

We next present an example of boundary segmentation using a Bayesian statistical model. Non-Bayesian methods of the type described above require additional processing to thin edges and link them, and, as such, they are difficult to implement without much trial and error.

Recall the Bayesian statistical model used in (1) with $X$ denoting the observed (noisy) image and $\theta$ the true image. Assume that $\theta$ is a Markov random field with realizations that are constant within connected regions of the $M \times N$ rectangular array (regions with constant gray values correspond to objects in the scene). Let $\omega$ denote the true boundary image such that $\omega(m,n) = 1$ if a boundary pixel is present at $(m,n)$, and $\omega(m,n) = 0$ otherwise. The goal is to predict $\omega$ (optimally); from our loss function formulation, the optimal predictor will be based on the posterior distribution

$$p(\omega|x) \propto f(x|\omega)\pi(\omega),$$

where $\pi(\omega)$ is the prior on the boundary image $\omega$. Boundaries of objects should be closed and thin; this will be guaranteed [24] with a choice of prior taking the form of a Markov random field, where

$$\pi(\omega) \propto \exp \left[ -\sum_{\kappa \in \mathcal{C}} V_{\kappa}(\omega_{\kappa}) \right],$$

$\mathcal{C}$ is the set of all cliques, and $V_{\kappa}(\omega_{\kappa}) = \infty$ if

1. boundary pixels are isolated or terminate in any eight-neighborhood contained in $\kappa$ (this guarantees closed boundaries); or
2. neighborhood boundary pixels in an eight-neighborhood of a central boundary pixel are vertically, horizontally, or diagonally adjacent to each other, where the eight-neighborhood is contained in $\kappa$ (this guarantees one-pixel-wide boundaries).

This specification of the prior $\pi(\omega)$ guarantees that its support, and that of the posterior $\pi(\omega|x)$, is contained in $\Omega_{p}$, the set of all closed and one-pixel-wide boundaries (that is, the set of permissible boundaries).

A Gaussian model is assumed for the observed intensities: $X(m,n) = \theta(m,n) + \epsilon(m,n)$, where $\epsilon$ is a Gaussian white noise process with zero mean and variance $\sigma^{2}$, independent of $\theta$. Each $\omega \in \Omega_{p}$ partitions the $M \times N$ array into disjoint connected regions $\{d_{i}(\omega) : i = 1, \ldots, K(\omega)\}$, where it is assumed that $\theta$ is constant on connected regions.
Define \( \mu_i(\omega) \equiv \theta(m, n) \) for \( (m, n) \in \mathcal{D}_i(\omega) \), and let \( l_i(\omega) \) denote the number of distinct sites \( (m_{ij}, n_{ij}) \in \mathcal{D}_i(\omega) \). Thus

\[
f(x|\omega) = \prod_{i=1}^{K_{(\omega)}} \prod_{j=1}^{l_i(\omega)} \left[ \frac{(2\pi\sigma^2)^{-1/2} \exp \left[ -\frac{(x(m_{ij}, n_{ij}) - \mu_i(\omega))^2}{2\sigma^2} \right] } \right],
\]

which models the probability density function of the observed intensities, assumed to be conditionally independent.

As a consequence of Bayes’ Theorem and the Markov-random-field form of the prior \( \pi(\omega) \), the posterior \( p(\omega|x) \) is also a Markov random field with the same neighborhoods and cliques as the prior, but with potential functions modified by the noise distribution \( f(x|\omega) \) given above. Recall that the optimal estimator under 0–1 loss is the MAP estimator \( \delta^*(x) \) that maximizes \( p(\omega|x) \) with respect \( \omega \). It is inefficient to search through all the \( 2^{MN} \) possible boundary images to find the one in \( \Omega_1 \) that achieves the maximum. A stochastic relaxation method called simulated annealing\(^*\) is guaranteed, when implemented properly, to converge to the global maximum [16]. An approximation that is computationally more feasible, called iterated conditional modes (ICM), was introduced by Besag [5]. When the ICM algorithm converges, it does so to a local maximum.

Figure 4(a) shows a \( 64 \times 64 \) input image \( X \) taken from a larger, synthetic aperture radar (or SAR) image of sea ice; the intensity values range from 0 to 255 [24]. Figure 4(b) is the boundary image used to start the ICM algorithm; notice that it is closed and one pixel wide. Figure 4(c) shows the result after one iteration of ICM; Fig. 4(d) shows the (locally) optimal boundary image to which ICM converged (after four iterations).

**FEATURE EXTRACTION**

A feature of an image is a descriptor that gives quantitative information about some region within the image. For example, in remote sensing, the gray value \( X(m, n) \) might contain information about the reflectivity or transmissivity at pixel location \( (m, n) \); in infrared imagery, the amplitude of the signal represents temperatures of the objects in the scene. Other features include those based on a histogram of gray values, on edges, on shapes, on textures, and on transforms.

![Figure 4. (a) Input image to boundary detection algorithm. (b) Boundary image used to start the ICM algorithm. (c) Boundary image after one iteration of the ICM algorithm. (d) Boundary image output from ICM algorithm [18].](image-url)
Once a region in an image has been identified using segmentation techniques, a histogram of the gray values within that region can be calculated. By identifying a random variable $G$ with the normalized histogram, so that

$$Pr[G = g] \equiv f(g)$$

$$= \frac{\text{number of pixels with gray value } g}{\text{total number of pixels in region}}$$

one can define histogram features of that region. They include histogram features of the boundary as a one-dimensional, complex-valued sequence of points and use coefficients of the Fourier series representation to give a unique boundary representation. However, this latter method is sensitive to the starting point, rotation, and scaling of the boundary.

If the interior of a closed-loop boundary is given gray value 1 and the exterior gray value 0, the two-dimensional moments provide shape information [44]. The $(p, q)$th moment is

$$m_{pq} = \text{ave}(x^p y^q : X(i, j) = 1); \quad p, q = 0, 1, \ldots,$$

where $\text{ave}()$ denotes the average of the argument set. The central moments are then

$$\mu_{pq} = \text{ave}((i - m_{10})^p (j - m_{01})^q : X(i, j) = 1); \quad p, q = 0, 1, \ldots$$

For example, $m_{10}$ and $m_{01}$ represent the center of gravity; $\mu_{20}, \mu_{02}$, and $\mu_{11}$ represent moments of inertia; and $\mu_{30}, \mu_{21}, \ldots$ represent asymmetry characteristics of an object.

Related to shape features are topological features, such as the Euler number and the number of holes for binary images [36]. The number of holes in the binary image is counted as the number of regions of white (the background) that are wholly contained within the regions of black (the objects). The Euler number is defined to be the number of connected regions minus the number of holes within the regions.

Texture features include a variety of different measures. The size of texture building blocks (or texels) can be represented by the spatial range of the autocorrelation function (ACF) (see SERIAL CORRELATION). However, several different textures can have the same ACF, so this is not always a reliable measure [26]. Parameters estimated from spatial statistical models can also be thought of as texture features [11].

Histogram features can give texture information; the gray level co-occurrence matrix is a popular, although computationally-intensive method [20]. This matrix gives information regarding the relative positions of pixel values that are spatially close. First, a vector $v$ is chosen that describes the relative displacement of pairs of pixel locations in the image. Then, the frequency of occurrence of pairs of pixel gray values are counted.
for all those pairs of pixel locations that are displaced by vector $v$ and are within a given subregion of the image. For a region with $K$ gray values, the co-occurrence matrix will be a $K \times K$ matrix, where the $(i, j)$th entry in the matrix counts the number of times gray level $i$ occurs at the tail of vector $v$ and gray level $j$ occurs at the head of $v$. The matrix is usually normalized; in this case, the $(i, j)$th entry gives an estimate of the joint probability that a pair of points at a relative displacement $v$ will have gray values $i$ and $j$. The co-occurrence matrix is essentially a local two-dimensional histogram which, if the number of gray values is more than just a few, can require large amounts of computation time.

Transform features involve representing the region via a standard series expansion that guarantees uniqueness in the coefficients of the transformed signal, and then using information in the transformed image as a feature of the region. The discrete Fourier transform $\tilde{X}$ of an image $X$ is an example of such a feature. After band-passing certain frequencies and setting the rest of the frequency coefficients equal to zero (e.g., bandpass frequencies within an annular region around the origin), the result can be inverse-transformed back to the spatial domain. Other transformations, such as the Haar wavelet*, discrete cosine, and Hotelling (or Karhunen-Loeve*) transforms, also provide features [26].

OTHER APPLICATIONS

Inverse problems such as pattern recognition* and function approximation occur regularly in the context of image processing. Standard statistical pattern recognition techniques for images are reviewed by Fu [15] and Gonzalez and Thomason [19]; newer developments that use probability and statistics in one form or another include artificial neural networks [8,23,30], fuzzy logic systems [29], and genetic algorithms [18]. Digital morphing or warping [50], which is a geometric transformation of a digital image used primarily in television advertisements and music videos, is inherently a sampling process. The transformations involve generating a sequence of images, whereby certain strategically sampled gray values remain constant, but their spatial location changes. Interpolation of the data must be performed from frame to frame in the remaining locations, which can involve statistical concepts.

Remote sensing of the earth's surface is now achieved through platforms that allow several different sensors to sample simultaneously the same scene; for example, electromagnetic radiation can be sensed over a number of different frequency bands. Colored images can be considered to be multispectral, corresponding to the intensities of red, green, and blue frequencies. Occasionally, one sees gray-value images converted to color images by assigning colors to various ranges of gray values; these are called pseudo-color images. As a consequence of having several pieces of information at one pixel location, the “fusion” (i.e., multivariate analysis) of the vector of image values has developed into a recent research area. A multivariate statistical approach can be found, for example, in McLachlan [34], and Cressie and Helterbrand [10] summarize the multivariate spatial models that could be used for statistical modeling.

Because image data require huge amounts of memory for processing as well as massive databases for storage, much effort has been expended on efficient compression of data with little loss of information [38]. In addition, efficient ways to encode digital images for transmission over communications channels are being sought. Image compression techniques fall into two main areas: predictive encoding [21] and transform coding [42]. Predictive-encoding techniques take advantage of redundancy in the image data. Transform-coding techniques transform the image data so that a large amount of information is packed into a small number of samples or coefficients. Typically there is loss of information when compressing data but, up to a point, this is outweighed by vast improvements in their storage and transmission.

Finally, an area of important activity is computer-vision algorithms. Once the image has been processed at the pixel level, extracting features and other information, regional descriptions can be generated. Higher-level
conceptual processing is necessary to identify information in images and these techniques are referred to collectively as computer vision techniques [3]. Geometrical properties of shapes, such as maximal diameter, can be derived from shape features, texture properties can help define boundaries between regions, and moment values can give properties about regions. Using a priori information whenever possible, one classifies the regions or objects. After classification, relational information between identified objects in the image is output or is used for further processing.

For example, suppose that a region in an image has been identified as containing malignant cells and the region is contained within a certain organ that has also been identified. Then that relational information can be output to the human user, or further processing can be performed to identify the type of cells more precisely. More abstract constructs such as syntactic grammars [15] are also used for extracting relational information from processed images.

REFERENCES


**ANNOTATED BIBLIOGRAPHY**

(Some of these also appear in the preceding references.)

Ballard, D. H. and Brown, C. M. (1982). *Computer Vision*. Englewood Cliffs, NJ. An extensive collection of high-level image understanding techniques with a strong artificial-intelligence flavor. Objects are described at four levels of abstraction: images; segmented images; geometric structures; and relational structures.


Grenander, U. (1994). *General Pattern Theory*. Oxford University Press, Oxford. Creates mathematical knowledge representations of complex patterns, which are expressed through their typical behavior as well as through their variability around their typical form. Algorithms are derived for the understanding, recognition, and restoration of observed patterns.


Mardia, K. V., ed. (1994). *Statistics and Images: 2. Advances in Applied Statistics*, a supplement to the *Journal of Applied Statistics*. Carfax Publishing, Oxfordshire. A follow-on from Volume 1, where new topics such as neural networks, pattern recognition, object recognition, wavelets, statistical morphology, and motion deblurring are covered. The chapters are written by a number of authors and are solely devoted to new results in statistical image analysis.


Serra, J. (1982). *Image Analysis and Mathematical Morphology*. Academic Press, London. A comprehensive monograph on basic and advanced topics in mathematical morphology, where the structure of objects modeled by sets in Euclidean space is formalized. Considerations for digital image analysis are also given.


See also Gibbs Sampling; Image Restoration and Reconstruction; Markov Random Fields; Simulated Annealing; Spatial Data Analysis; and Spatial Processes.

Noel Cressie
Jennifer L. Davidson