J. Webster (ed.), *Wiley Encyclopedia of Electrical and Electronics Engineering*

IMAGE ENHANCEMENT

Thirty years ago, the acquisition and processing of digital imagery belonged almost entirely in the domain of military, academic, and industrial research laboratories. Today, elementary school students download digital pictures from the World Wide Web, proud parents store photographs on a digital CD, business executives cut deals via digital video teleconferencing, and sports fans watch their favorite teams on digital satellite TV. These digital images are properly considered to be sampled versions of continuous real-world pictures. Because they are stored and processed on digital computers, digital images are typically discrete domain, discrete range signals. These signals can be conveniently represented and manipulated as matrices containing the light intensity or color information at each sampled point.

When the acquired digital image is not fit for a prescribed use, image enhancement techniques may be used to modify the image. According to Gonzales and Woods (1), "The principal objective of enhancement techniques is to process an image so that the result is more suitable than the original image for a specific application." So, the definition of image enhancement is a fairly broad concept that can encompass many applications. More usually, however, the image enhancement process seeks to enhance the apparent visual quality of an image or to emphasize certain image features. The benefactor of image enhancement either may be a human observer or a computer vision program performing some kind of higher-level image analysis, such as target detection or scene understanding.

The two simplest methods for image enhancement involve simple histogram-modifying point operations or spatial digital filtering. More complex methods involve modifying the image content in another domain, such as the coefficient domain of a linear transformation of the image. We will consider all of these approaches in this article. A wealth of literature exists on point operations, linear filters, and nonlinear filters. In this discussion, we highlight several of the most important image enhancement standards and a few recent innovations. We will begin with the simplest.

Image Enhancement Techniques

Denote a two-dimensional digital image of gray-level intensities by $\mathbf{I}$. The image $\mathbf{I}$ is ordinarily represented in software accessible form as an $M \times N$ matrix containing indexed elements $I(i, j)$, where $0 \leq i \leq M - 1$, $0 \leq j \leq N - 1$. The elements $I(i, j)$ represent samples of the image intensities, usually called *pixels* (*pict*ure *el*ements). For simplicity, we assume that these come from a finite integer-valued range. This is not unreasonable, since a finite wordlength must be used to represent the intensities. Typically, the pixels represent optical intensity, but they may also represent other attributes of sensed radiation, such as radar, electron micrographs, x rays, or thermal imagery.

Point Operations. Often, images obtained via photography, digital photography, flatbed scanning, or other sensors can be of low quality due to a poor image contrast or, more generally, from a poor usage of the available range of possible gray levels. The images may suffer from overexposure or from underexposure, as in the "mandrill" image in Fig. 1(a). In performing image enhancement, we seek to compute $\mathbf{J}$, an enhanced

version of **I**. The most basic methods of image enhancement involve *point operations,* where each pixel in the enhanced image is computed as a one-to-one function of the corresponding pixel in the original image: $J(i, j) = f[I(i, j)]$. The most common point operation is the linear contrast stretching operation, which seeks to maximally utilize the available gray-scale range. If a is the minimum intensity value in image **I** and b is the maximum, the point operation for linear contrast stretching is defined by

$$J(i, j) = \frac{K - 1}{b - a}[I(i, j) - a] \tag{1}$$

assuming that the pixel intensities are bounded by $0 \leq I(i, j) \leq K - 1$, where K is the number of available pixel intensities. The result image **J** then has maximum gray level $K - 1$ and minimum gray level 0, with the other gray levels being distributed in-between according to Eq. (1). Figure 1(b) shows the result of linear contrast stretching on Fig. 1(a).

Several point operations utilize the image *histogram,* which is a graph of the frequency of occurrence of each gray level in **I**. The histogram value $H_{\mathbf{I}}(k)$ equals n only if the image **I** contains exactly n pixels with gray level k. Qualitatively, an image that has a flat or well-distributed histogram may often strike an excellent balance between contrast and preservation of detail. Histogram flattening, also called *histogram equalization* in Gonzales and Woods (1), may be used to transform an image **I** into an image **J** with approximately flat histogram. This transformation can be achieved by assigning

$$J(i, j) = (K - 1)P(i, j) \tag{2}$$

where $P(i, j)$ is a sample cumulative probability formed by using the histogram of **I**:

$$P(i, j) = \frac{1}{MN} \sum_{k=0}^{I(i,j)} H_1(k) \tag{3}$$

The image in Fig. 1(c) is a histogram-flattened version of Fig. 1(a).

A third point operation, frame averaging, is useful when it is possible to obtain multiple images $\mathbf{G}_i$, $i = 1, \ldots, n$, of the same scene, each a version of the ideal image **I** to which deleterious noise has been unintentionally added:

$$\mathbf{G}_i = \mathbf{I} + \mathbf{N}_i \tag{4}$$

where each noise "image" $\mathbf{N}_i$ is an $M \times N$ matrix of discrete random variables with zero mean and variance σ^2. The noise may arise as electrical noise, noise in a communications channel, thermal noise, or noise in the sensed radiation. If the noise images are not mutually correlated, then averaging the n frames together will form an effective estimate $\hat{\mathbf{I}}$ of the uncorrupted image **I**, which will have a variance of only σ^2/n:

$$\hat{I}(i, j) = \frac{1}{n} \sum_{i=1}^{n} G_i(i, j) \tag{5}$$

This technique is only useful, of course, when multiple frames are available of the same scene, when the *information* content between frames remains unchanged (disallowing, for example, motion between frames), and when the *noise* content does change between frames. Examples arise quite often, however. For example, frame averaging is often used to enhance synthetic aperture radar images, confocal microscope images, and electron micrographs.

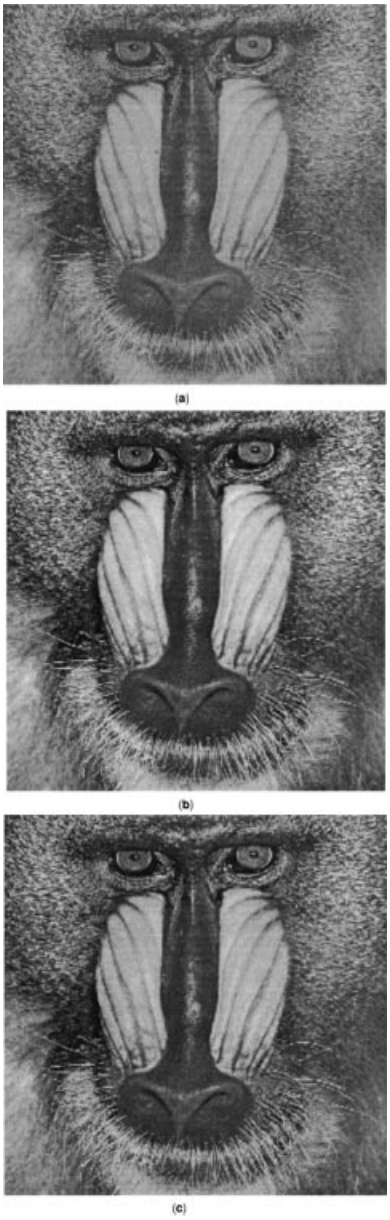

Fig. 1. (a) Original "Mandrill" image (low contrast). (b) "Mandrill" enhanced by linear contrast stretching. (c) "Mandrill" after histogram equalization.

Linear Filters. Linear filters obey the classical linear superposition property as with other linear systems found in the controls, optics, and electronics areas of electrical engineering (2). Linear filters can be realized by linear convolution in the spatial domain or by pointwise multiplication of discrete Fourier transforms in the frequency domain. Thus, linear filters can be characterized by their frequency selectivity and spectrum shaping. As with 1-D signals, 2-D digital linear filters may be of the low-pass, high-pass or band-pass variety.

4 IMAGE ENHANCEMENT

Much of the current interest in digital image processing can be traced to the rediscovery of the *fast Fourier transform* (*FFT*) some 30 years ago (it was known by Gauss). The FFT computes the discrete Fourier transform (*DFT*) of an $N \times N$ image with a computational cost of $O(N^2 \log_2 N)$, whereas naive DFT computation requires N^4 operations. The speedup afforded by the FFT is tremendous. This is significant in linear filtering-based image enhancement, since linear filters are implemented via convolution:

$$\mathbf{J} = \mathbf{F} * \mathbf{G} \tag{6}$$

where $\mathbf{F}$ is the impulse response of the linear filter, $\mathbf{G}$ is the original image, and $\mathbf{J}$ is the filtered, enhanced result. The convolution in Eq. (6) may be implemented in the frequency domain by the following pointwise multiplication ($\cdot$) and inverse Fourier transform (*IFFT*):

$$\mathbf{J} = \mathrm{IFFT}[\mathrm{FFT}(\mathbf{F}_0) \cdot \mathrm{FFT}(\mathbf{G}_0)] \tag{7}$$

where $\mathbf{F}_0$ and $\mathbf{G}_0$ are $2N \times 2N$ *zero-padded* versions of $\mathbf{F}$ and $\mathbf{G}$. By this we mean that $F_0(i,j) = F(i,j)$ for $0 \leq i, j \leq N-1$ and $F_0(i,j) = 0$ otherwise; similarly for $\mathbf{G}_0$. The zero padding is necessary to eliminate wraparound effects in the FFTs which occur because of the natural periodicities that occur in sampled data.

If $\mathbf{G}$ is corrupted as in Eq. (4) and $\mathbf{N}$ contains white noise with zero mean, then enhancement means noise-smoothing, which is usually accomplished by applying a low-pass filter of a fairly wide bandwidth. Typical low-pass filters include the average filter, the Gaussian filter and the ideal low-pass filter. The average filter can be supplied by averaging a neighborhood (an $m \times m$ neighborhood, for example) of pixels around $G(i, j)$ to compute $J(i, j)$. Likewise, average filtering can be viewed as convolving $\mathbf{G}$ with a box-shaped kernel $\mathbf{F}$ in Eq. (7). An example of average filtering is shown in Fig. 2(a)–(c). Similarly, a Gaussian-shaped kernel $\mathbf{F}$ may be convolved with $\mathbf{G}$ to form a smoothed, less noisy image, as in Fig. 2(d). The Gaussian-shaped kernel has the advantage of giving more weight to closer neighbors and is well-localized in the frequency domain, since the Fourier transform of the Gaussian is also Gaussian-shaped. This is important because it reduces noise "leakage" at higher frequencies. In order to provide an "ideal" cutoff in the frequency domain, the FFT of $\mathbf{G}_0$ can be zeroed beyond a cutoff frequency (this is equivalent to multiplying by a binary DFT $\mathbf{F}_0$ in Eq. (7)). This result, shown in Fig. 2(e), reveals the *ringing* artifacts associated with an ideal low-pass filter.

Linear filters are also useful when the goal of image enhancement is sharpening. Often, an image is blurred by a novice photographer who moves the camera or improperly sets the focus. Images are also blurred by motion in the scene or by inherent optical problems, such as with the famous Hubble telescope. Indeed, any optical system supplies contributes some blur to the image. Motion blur and defocus can also be modeled as a linear convolution of $\mathbf{B}*\mathbf{I}$, where $\mathbf{B}$, in this case, represents linear distortion. This distortion is essentially a low-pass process; therefore, a high-pass filter can be used to sharpen or deblur the distorted image. The most obvious solution is create an *inverse filter* $\mathbf{B}^{-1}$ that exactly reverses the low-pass blurring of $\mathbf{B}$. The inverse filter is typically defined in the frequency domain by mathematically inverting each frequency component of the Fourier transform of $\mathbf{B}$, creating a high-pass filter $\mathbf{B}^{-1}$. Let the complex-valued components of the DFT of $\mathbf{B}$ be denoted by $\tilde{B}(u,v)$. Then, the components of $\mathbf{B}^{-1}$ are given by

$$\tilde{B}^{-1}(u,v) = \frac{1}{\tilde{B}(u,v)} \tag{8}$$

The image can be sharpened by pointwise multiplying the (zero-padded) FFT of the blurred image by the (zero-padded) FFT of $\mathbf{B}^{-1}$, then performing the inverse FFT operation, which is why this enhancement is often called *deconvolution*. It must be noted that difficulty arises when the Fourier transform of $\mathbf{B}$ contains zero-valued

Fig. 2. (a) Original "Winston" image. (b) Corrupted "Winston" image with additive Gaussian-distributed noise ($\sigma = 10$). (c) Average filter result (5×5 window). (d) Gaussian filter result (standard deviation $\sigma = 2$). (e) Ideal low-pass filter result (cutoff $= N/4$). (f) Wavelet shrinkage result.

elements. In this case, a simple solution is the *pseudo-inverse filter,* which leaves the zeroed frequencies as zeros in the construction of $\mathbf{B}^{-1}$.

A challenging problem is encountered when *both* linear distortion ($\mathbf{B}$) and additive noise ($\mathbf{N}$) degrade the image $\mathbf{I}$:

$$\mathbf{G} = \mathbf{B} * \mathbf{I} + \mathbf{N} \tag{9}$$

If we apply a low-pass filter **F** to ameliorate the effects of noise, then we only further blur the image. In contrast, if we apply an inverse (high-pass) filter $\mathbf{B}^{-1}$ to deblur the image, the high-frequency components of the broadband noise **N** are amplified, resulting in severe corruption. This ill-posed problem of conflicting goals can be attacked by a compromise between low-pass and high-pass filtering. The famous Wiener filter provides such a compromise [see Russ (3)]. If η represents the noise power and **N** is white noise, then the frequency response of the Wiener filter is defined by

$$\tilde{F}(u,v) = \frac{\tilde{B}^*(u,v)}{|\tilde{B}(u,v)|^2 + \eta} \tag{10}$$

where $\tilde{B}(u,v)$ is the complex conjugate of $\tilde{B}(u,v)$. The Wiener filter attempts to balance the operations of denoising and deblurring optimally (according to the mean-square criterion). As the noise power is decreased, the Wiener filter becomes the inverse filter, favoring deblurring. However, the Wiener filter usually produces only moderately improved results, since the tasks of deconvolution (high-pass filtering) and noise-smoothing (low-pass filtering) are antagonistic to one another. The compromise is nearly impossible to balance using purely linear filters.

Nonlinear Filters. Nonlinear filters are often designed to remedy deficiencies of linear filtering approaches. Nonlinear filters cannot be implemented by convolution and do not provide a predictable modification of image frequencies. However, for this very reason, powerful nonlinear filters can provide performance attributes not attainable by linear filters, since frequency separation (between image and noise) is often not possible.

Nonlinear filters are usually defined by local operations on *windows* of pixels. The window, or structuring element, defines a local neighborhood of pixels such as the window of pixels at location (i, j):

$$\mathbf{G}_{\mathbf{K}}(i,j) = \{G(i-m, j-n) : (m,n) \in \mathbf{K}\} \tag{11}$$

where **K** is the window defining pixel coordinate offsets belonging to the local neighborhood of $I(i, j)$. The output pixels in the filtered image **J** can be expressed as nonlinear many-to-one functions of the corresponding *windowed sets* of pixels in the image **G**:

$$J(i,j) = f[\mathbf{G}_{\mathbf{K}}(i,j)] \tag{12}$$

So, the nonlinear filtering operation may be expressed as a function of the image and the defined moving window: $\mathbf{J} = f(\mathbf{G}, \mathbf{K})$. The windows come in a variety of shapes, mostly symmetric and centered. The size of the window determines the scale of the filtering operation. Larger windows will tend to produce more coarse scale representations, eliminating fine detail.

Order Statistic Filters and Image Morphology. Within the class of nonlinear filters, order statistic (*OS*) filters encompass a large group of effective image enhancers. A complete taxonomy is given in Bovik and Acton (4). The OS filters are based on an arithmetic ordering of the pixels in each window (local neighborhood). At pixel location (i, j) in the image **G**, given a window **K** of $2m + 1$ pixels, the set of order statistics is denoted by

$$\mathbf{G}_{\mathbf{K}}^{\mathrm{OS}}(i,j) = \{G_{(1)}^{\mathrm{OS}}(i,j), G_{(2)}^{\mathrm{OS}}(i,j), \ldots, G_{(2m+1)}^{\mathrm{OS}}(i,j)\} \tag{13}$$

where $G^{\mathrm{OS}}{}_{(1)}(i,j) \leq G^{\mathrm{OS}}{}_{(2)}(i,j) \leq \ldots \leq G^{\mathrm{OS}}{}_{(2m+1)}(i,j)$. These are just the original pixel values covered by the window and reordered from smallest to largest.

Perhaps the most popular nonlinear filter is the median filter (5). It is an OS filter and is implemented by

$$J(i,j) = G^{\mathrm{OS}}_{(m+1)}(i,j) \tag{14}$$

assuming a window size of $2m + 1$ pixels. The median smoothes additive white noise while maintaining edge information—a property that differentiates it from all linear filters. Particularly effective at eliminating impulse noise, the median filter has strong optimality properties when the noise is Laplacian-distributed (6).

An example of the smoothing ability of the median filter is shown in Fig. 3(a)–(c). Here, a square 5×5 window was used, preserving edges and removing the impulse noise. Care must be taken when determining the window size used with the median filter, or streaking and blotching artifacts may result (7).

More general OS filters can be described by a weighted sums of the order statistics as follows:

$$\begin{aligned} J(i,j) = {} & A(1)G^{\mathrm{OS}}_{(1)}(i,j) + A(2)G^{\mathrm{OS}}_{(2)}(i,j) + \cdots \\ & + A(2m+1)G^{\mathrm{OS}}_{(2m+1)}(i,j) \end{aligned} \tag{15}$$

where **A** is the vector that determines the weight of each OS. Several important enhancement filters evolve from this framework. The *L-inner mean filter* or *trimmed mean* filter may be defined by setting $A(k) = 1/(2L + 1)$ for $(m + 1) \leq (m + 1 + L)$ and $A(k) = 0$ otherwise. This filter has proven to be robust (giving close to optimal performance) in the presence of many types of noise. Thus, it is often efficacious when the noise is unknown. *Weighted median filters* also make up a class of effective, robust OS filters (8,9).

Other nonlinear filters strongly related to OS filters include *morphological filters,* which manipulate image intensity profiles and thus are shape-changing filters in these regard. Image morphology is a rapidly exploding area of research in image processing (10). Through the concatenation of a series of simple OS filters, a wide scope of processing techniques emerge. Specifically, the basic filters used are the erosion of **G** by structuring element **K** defined by

$$J(i,j) = G^{\mathrm{OS}}_{(1)}(i,j) \tag{16}$$

and the dilation of **G** by **K** defined by

$$J(i,j) = G^{\mathrm{OS}}_{(2m+1)}(i,j) \tag{17}$$

The erosion of **G** by **K** is often represented by $\mathbf{G} \ominus \mathbf{K}$, while the dilation is represented by $\mathbf{G} \oplus \mathbf{K}$.

By themselves, the erode and dilate operators are not useful for image enhancement, because they are biased and do not preserve shape. However, the alternating sequence of erosions and dilations is indeed useful. The *close filter* is constructed by performing dilation and then erosion:

$$\mathbf{G} \cdot \mathbf{K} = (\mathbf{G} \oplus \mathbf{K}) \ominus \mathbf{K} \tag{18}$$

while the *open filter* is erosion followed by dilation:

$$\mathbf{G} \circ \mathbf{K} = (\mathbf{G} \ominus \mathbf{K}) \oplus \mathbf{K} \tag{19}$$

Open and close filters are *idempotent,* so further closings or openings yield the same result, much like band-pass filters in linear image processing. Although bias-reduced, the open filter will tend to remove small bright

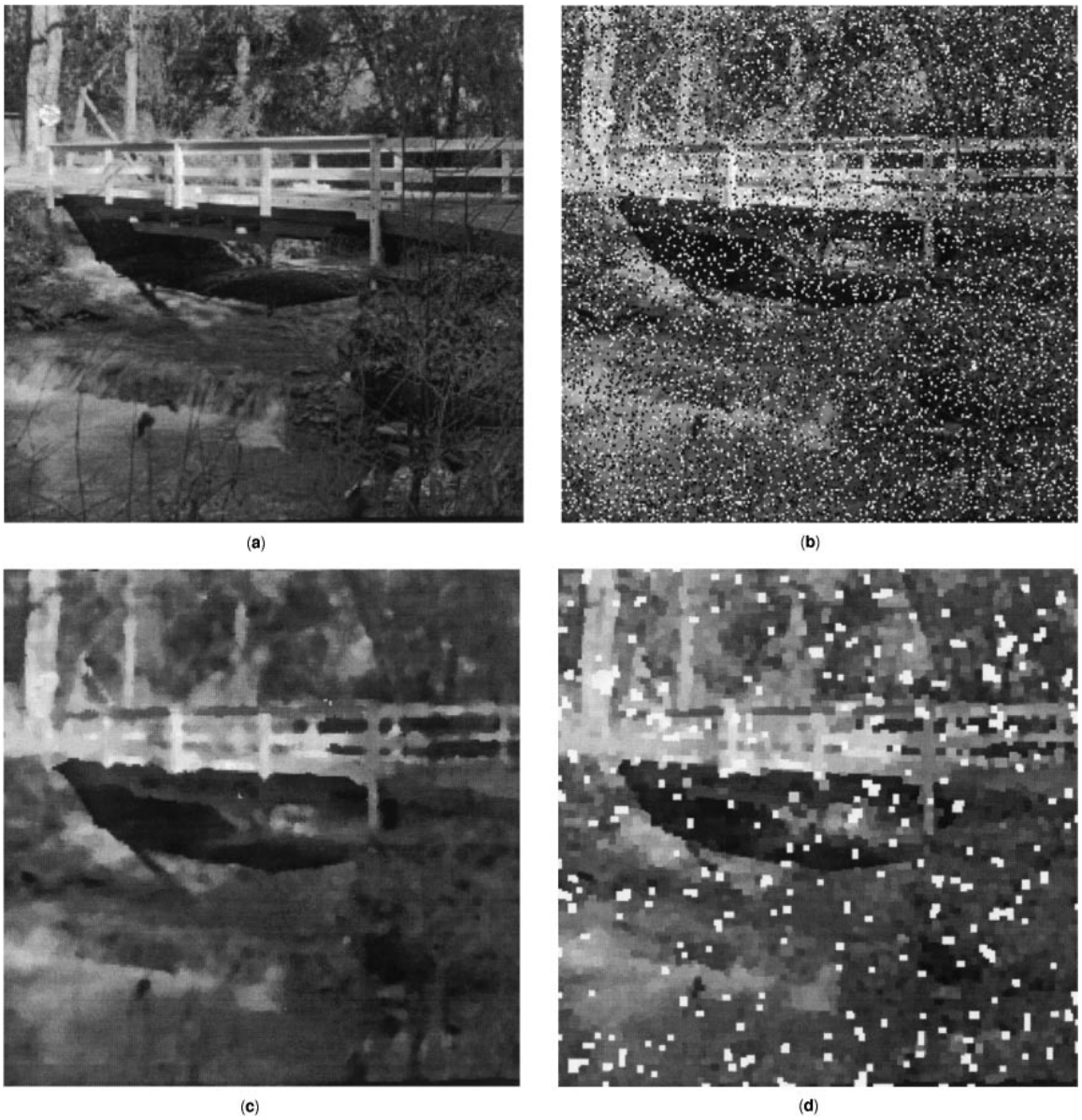

Fig. 3. (a) Original "Bridge" image. (b) "Bridge" image corrupted with 20% salt and pepper noise. (c) Median filter result ($\mathbf{B} = 5 \times 5$ square). (d) Open–close filter result ($\mathbf{B} = 3 \times 3$ square).

image regions and will separate loosely connected regions, while the close filter will remove dark spots and will link loosely connected regions.

To emulate the smoothing properties of the median filter with morphology, the open and close filters can be applied successively. The *open–close filter* is given by $(\mathbf{G} \circ \mathbf{K}) \cdot \mathbf{K}$, and the *close–open filter* is given by $(\mathbf{G} \cdot \mathbf{K}) \circ \mathbf{K}$. Since open–close and close–open filters involve only minimum (erode) and maximum (dilate) calculations, they offer an affordable alternative to the median OS filter, which requires a more expensive ordering of each

windowed set of pixels. However, in the presence of extreme impulse noise, such as the salt and pepper noise shown in Fig. 3(b), the open–close (or close–open) cannot reproduce the results of the median filter [see Fig. 3(d)].

Many variants of the OS and morphological filters have been applied successfully to certain image enhancement applications. The *weighted majority with minimum range* (*WMMR*) filter is a special version of the OS filter (17) where only a subset of the order statistics are utilized (11). The subset used to calculate the result at each pixel site is the group of pixel values with minimum range. The WMMR filters have been shown to have special edge enhancing abilities and, under special conditions, can provide near piecewise constant enhanced images.

To improve the efficiency of OS filters, *stack filters* were introduced by Wendt et al. (12). Stack filters exploit a *stacking property* and a superposition property called the *threshold decomposition.* The filter is initialized by decomposing the K-valued signal (by thresholding) into binary signals, which can be processed by using simple Boolean operators. "Stacking" the binary signals enables the formation of the filter output. The filters can be used for real-time processing.

One limitation of the OS filters is that spatial information inside the filter window is discarded when rank ordering is performed. A recent group of filters, including the C, Ll, and *permutation* filters, combine the spatial information with the rank ordering of the OS structure (13). The combination filters can be shown to have an improved ability to remove outliers, as compared to the standard OS design, but have the obvious drawback of increased computational complexity.

Diffusion Processes. A newly developed class of nonlinear image enhancement methods uses the analogy of heat diffusion to adaptively smooth the image. *Anisotropic diffusion,* introduced by Perona and Malik (14), encourages intraregion smoothing and discourages interregion smoothing at the image edges. The decision on local smoothing is based on a *diffusion coefficient* which is generally a function of the local image gradient. Where the gradient magnitude is relatively low, smoothing ensues. Where the gradient is high and an edge may exist, smoothing is inhibited. A discrete version of the diffusion equation is

$$\begin{aligned}[I(i,j)]_{t+1} = \{I(i,j) + \lambda[&c_N(i,j)\nabla \boldsymbol{I}_N(i,j) + c_S(i,j)\nabla \boldsymbol{I}_S(i,j) \\ &+ c_E(i,j)\nabla \boldsymbol{I}_E(i,j) + c_W(i,j)\nabla \boldsymbol{I}_W(i,j)]\}_t\end{aligned} \tag{20}$$

where t is the iteration number, λ is a rate parameter ($\lambda \leq 1/4$), and the subscripts N, S, E, W represent the direction of diffusion. So, $\nabla I_N(i,j)$ is the simple difference (directional derivative) in the northern direction [i.e., $\nabla I_N(i,j) = I(i-1,j) - I(i,j)$], and $c_N(i,j)$ is the corresponding diffusion coefficient when diffusing image $\mathbf{I}$ and location (i,j). One possible formation of the diffusion coefficient (for a particular direction d) is given by

$$c_d(i,j) = \exp\left\{-\left[\frac{\nabla I_d(i,j)}{k}\right]^2\right\} \tag{21}$$

where k is an edge threshold. Unfortunately, Eq. (21) will preserve outliers from noise as well as edges. To circumvent this problem, a new diffusion coefficient has been introduced that uses a filtered image to compute the gradient terms (15). For example, we can use a Gaussian-filtered image $\mathbf{S} = \mathbf{I} * \mathbf{G}(\sigma)$ to compute the gradient terms, given a Gaussian-shaped kernel with standard deviation σ. Then the diffusion coefficient becomes

$$c_d(i,j) = \exp\left\{-\left[\frac{\nabla S_d(i,j)}{k}\right]^2\right\} \tag{22}$$

Fig. 4. (a) Original "Cameraman" image. (b) "Cameraman" image corrupted with additive Laplacian-distributed noise. (c) After eight iterations of anisotropic diffusion with diffusion coefficient in Eq. (21). (d) After eight iterations of anisotropic diffusion with diffusion coefficient in Eq. (22).

and can be used in Eq. (20). A comparison between the two diffusion coefficients is shown in Fig. 4 for an image corrupted with Laplacian-distributed noise. After eight iterations of anisotropic diffusion using the diffusion coefficient of Eq. (21), sharp details are preserved, but several outliers remain [see Fig. 4(c)]. Using the diffusion coefficient of Eq. (22), the noise is eradicated but several fine features are blurred [see Fig. 4(d)].

Anisotropic diffusion is a powerful enhancement tool, but is often limited by the number of iterations needed to achieve an acceptable result. Furthermore, the diffusion equation is inherently ill-posed, leading

to divergent solutions and introducing artifacts such as "staircasing." Research continues on improving the computational efficiency and on developing robust well-posed diffusion algorithms.

Wavelet Shrinkage. Recently, *wavelet shrinkage* has been recognized as a powerful tool for signal estimation and noise reduction or simply *de-noising* (16). The wavelet transform utilizes scaled and translated versions of a fixed function, which is called a "wavelet," and is localized in both the spatial and frequency domains (17). Such a joint spatial-frequency representation can be naturally adapted to both the global and local features in images. The wavelet shrinkage estimate is computed via thresholding wavelet transform coefficients:

$$\mathbf{J} = \mathrm{IDWT}[f[\mathrm{DWT}[\mathbf{G}]]] \tag{23}$$

where DWT and IDWT stand for *discrete wavelet transform* and *inverse discrete wavelet transform,* respectively (17), and $f[\,]$ is a transform-domain point operator defined by either the *hard-thresholding* rule

$$f[x] = \begin{cases} x & \text{if } |x| > t \\ 0 & \text{otherwise} \end{cases} \tag{24}$$

or the *soft-thresholding* rule

$$f[x] = \begin{cases} x - t & \text{if } x > t \\ 0 & \text{if } |x| \le t \\ x + t & \text{if } x < -t \end{cases} \tag{25}$$

where the value of the threshold t is usually determined by the variance of the noise and the size of the image.

The key idea of wavelet shrinkage derives from the approximation property of wavelet bases. The DWT compresses the image **I** into a small number of DWT coefficients of large magnitude, and it packs most of the image energy into these coefficients. On the other hand, the DWT coefficients of the noise **N** have small magnitudes; that is, the noise energy is spread over a large number of coefficients. Therefore, among the DWT coefficients of **G**, those having large magnitudes correspond to **I** and those having small magnitudes correspond to **N**. Apparently, thresholding the DWT coefficients with an appropriate threshold removes a large amount of noise and maintains most image energy.

Though the wavelet shrinkage techniques were originally proposed for the attenuation of image-independent white Gaussian noise, they work as well for the suppression of other types of distortion such as the blocking artifacts in JPEG-compressed images (18,19). In this case, the problem of enhancing a compressed image may be viewed as a de-noising problem where we regard the compression error as additive noise.

We applied the wavelet shrinkage to enhancing the noisy image shown in Fig. 2(b) and show the de-noised image in Fig. 2(f), from which one can clearly see that a large amount of noise has been removed, and most of the sharp image features were preserved without blurring or ringing effects. This example indicates that wavelet shrinkage can significantly outperform the linear filtering approaches.

Figure 5 illustrates an example of the enhancement of JPEG-compressed images (20). Figure 5(a) shows a part of the original image. Fig. 5(b) shows the same part in the JPEG-compressed image with a compression ratio 32:1, where blocking artifacts are quite severe due to the loss of information in the process of compression. Figure 5(c) reveals the corresponding part in the enhanced version of Fig. 5(b), to which we have applied wavelet shrinkage. One can find that the blocking artifacts are greatly suppressed and the image quality is dramatically improved.

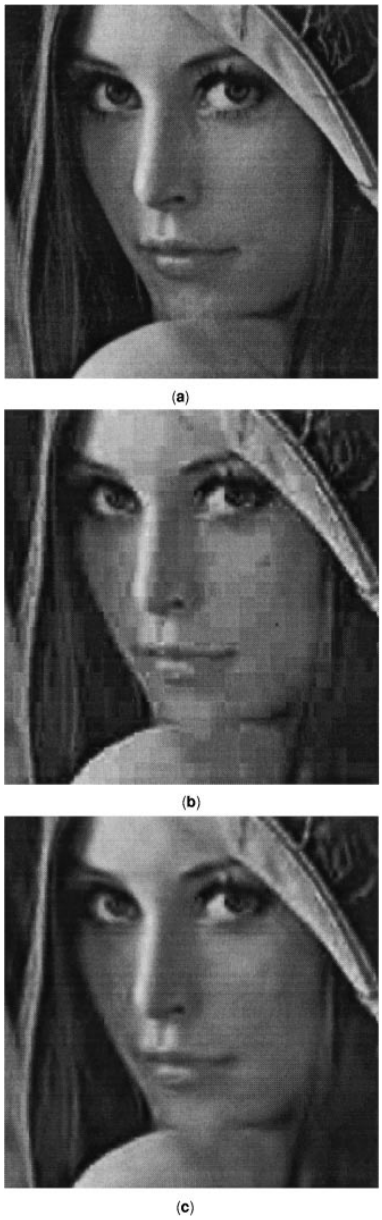

Fig. 5. (a) Original "Lena" image. (b) "Lena" JPEG-compressed at 32:1. (c) Wavelet shrinkage applied to Fig. 5b.

Homomorphic Filtering. To this point, we have described methods that only deal with additive noise. In several imaging scenarios, such as radar and laser-based imaging, signal-dependent noise is encountered. The signal-dependent noise can be modeled as a multiplicative process:

$$\mathbf{G} = \mathbf{N} \cdot \mathbf{I} \tag{26}$$

for noise values $N(i, j) \geq 0$ ("$\cdot$" is again pointwise multiplication). Applying the traditional low-pass filters or nonlinear filters is fruitless, since the noise is signal dependent. But we can decouple the noise from the signal using a homomorphic approach.

The first step of the homomorphic approach is the application of a logarithmic point operation:

$$G'(i, j) = \log[G(i, j)] \tag{27}$$

Since $\log[G(i, j)] = \log[N(i, j)] + \log[I(i, j)]$, we now have the familiar additive noise problem of Eq. (4). Then we can apply one of the filters discussed above, such as the median filter, and then transform the image back to its original range with an exponential point operation.

Applications and Extensions

The applications of image enhancement are as numerous as are the sources of images. Different applications, of course, benefit from enhancement methods that are tuned to the statistics of physical processes underlying the image acquisition stage and the noise that is encountered. For example, a good encapsulation of image processing for biological applications is found in Häder (21).

With the availability of affordable computing engines that can handle video processing on-line, the enhancement of time sequences of images is of growing interest. A video data set may be written as $I(i, j, k)$, where k represents samples of time. Many of the techniques discussed earlier can be straightforwardly extended to video processing, using three-dimensional FFTs, 3-D convolution, 3-D windows, and 3-D wavelet transforms. However, a special property of video sequences is that they usually contain image motion, which is projected from the motion of objects in the scene. The motion often may be rapid, leading to time-aliasing of the moving parts of the scene. In such cases, direct 3-D extensions of many of the methods discussed above (those that are not point operations) will often fail, since the processed video will often exhibit ghosting artifacts arising from poor handling of the aliasing data. This can be ameliorated via *motion-compensated* enhancement techniques. This generally involves two steps: motion estimation, whereby the local image motion is estimated across the image (by a matching technique), and compensation, where a correction is made to compensate for the shift of an object, before subsequent processing is accomplished. The topic of motion compensation is ably discussed in Tekalp (22).

BIBLIOGRAPHY

1. R. C. Gonzales R. E. Woods *Digital Image Processing*, Reading, MA: Addison-Wesley, 1995.
2. A. Antoniou *Digital Filters: Analysis and Design*, New York: McGraw-Hill, 1979.
3. J. C. Russ *The Image Processing Handbook*, Boca Raton, FL: CRC Press, 1995.
4. A. C. Bovik S. T. Acton The impact of order statistics on signal processing, In H. N. Nagaraja, P. K. Sen, and D. F. Morrison (eds.), *Statistical Theory and Applications*, New York: Springer-Verlag, 1996, pp. 153–176.
5. J. W. Tukey *Exploratory Data Analysis*, Reading, MA: Addison-Wesley, 1971.
6. A. C. Bovik T. S. Huang D. C. Munson, Jr. A generalization of median filtering using linear combinations of order statistics, *IEEE Trans. Acoust. Speech Signal Process.*, **ASSP-31**: 1342–1350, 1983.
7. A. C. Bovik Streaking in Median Filtered Images, *IEEE Trans. Acoust. Speech Signal Process.*, **ASSP-35**: 493–503, 1987.
8. O. Yli-Harja J. Astola Y. Neuvo Analysis of the properties of median and weighted median filters using threshold logic and stack filter representation. *IEEE Trans. Acoust. Speech Signal Process.*, **ASSP-39**: 395–410, 1991.
9. P.-T. Yu W.-H. Liao Weighted order statistic filters—their classification, some properties and conversion algorithm. *IEEE Trans. Signal Process.*, **42**: 2678–2691, 1994.

10. J. Serra Image Analysis and Morphology: Vol. 2: *The Theoretical Advances.* London: Academic Press, 1988.
11. H. G. Longbotham D. Eberly The WMMR filters: A class of robust edge enhancers, *IEEE Trans. Signal Process.*, **41**: 1680–1684, 1993.
12. P. D. Wendt E. J. Coyle N. C. Gallagher, Jr. Stack filters, *IEEE Trans. Acoust. Speech Signal Process.*, **ASSP-34**: 898–911, 1986.
13. K. E. Barner G. R. Arce Permutation filters: A class of nonlinear filters based on set permutations. *IEEE Trans. Signal Process.*, **42**: 782–798, 1994.
14. P. Perona J. Malik Scale-space and edge detection using anisotropic diffusion, *IEEE Trans. Pattern Anal. Mach. Intell.*, **PAMI-12**: 629–639, 1990.
15. F. Catte *et al.* Image selective smoothing and edge detection by nonlinear diffusion, *SIAM J. Numer. Anal.*, **29**: 182–193, 1992.
16. D. L. Donoho De-noising by soft-thresholding. *IEEE Trans. Inf. Theory*, **41**: 613–627, 1995.
17. G. Strang T. Nguyen *Wavelets and Filter Banks*, Wellesley, MA: Wellesley-Cambridge Press, 1996.
18. D. Wei C. S. Burrus Optimal wavelet thresholding for various coding schemes, *Proc. IEEE Int. Conf. Image Process.*, **I**: 610–613, 1995.
19. D. Wei A. C. Bovik Enhancement of compressed images by optimal shift-invariant wavelet packet basis. *J. Visual Commun. Image Represent.*, in press.
20. W. B. Pennebaker J. L. Mitchell *JPEG—Still Image Data Compression Standard*, New York: Van Nostrand Reinhold, 1993.
21. D. P. Häder (ed.) *Image Analysis in Biology*, Boca Raton, FL: CRC Press, pp. 29–53, 1991.
22. A. M. Tekalp *Digital Video Processing*, Upper Saddle River, NJ: Prentice-Hall, 1995.

SCOTT T. ACTON
Oklahoma State University
DONG WEI
The University of Texas at Austin
ALAN C. BOVIK
The University of Texas at Austin