PROCESSING IN ACTION IV

We have encountered a wide variety of digital image processing operations in the preceding chapters of this book. To facilitate the structure of presentation, however, these operations were discussed in the context of their implementation processes. Because of this, easy reference to a given operation is somewhat hindered.

In Part IV the commonly used digital image processing operations have been consolidated by presenting them in a catalog form. This section represents application groups—contrast enhancement, spatial filtering, edge enhancement, geometric manipulations, and other operations.

Each Image Operation Study contains a concise rundown of the functions, implementations, and drawbacks, as well as several before-and-after photographs of a selected operation. Part IV provides quick, concise explanations of these operations without the instructional overhead seen earlier.

Image Operation Studies

Image Operation Study # 1

Operation—Binary Contrast Enhancement

Description—Binary contrast enhancement is an operation used to process an input image into a high-contrast output image consisting of two gray levels, black and white. Pixel gray levels in the input image are individually compared to a selected threshold value. For an input pixel level less than the threshold value, the corresponding output pixel is set to black. Otherwise, the output pixel is set to white. The resulting image is, therefore, composed of black and white pixels, based on the brightness comparisons of the input pixel gray levels and the threshold value.

Application—This enhancement is particularly useful in the extraction of object boundaries from their background when the gray levels between the two are very close. For instance, printed text that has faded or is viewed under low-light-level conditions may be barely discernable from the surface in which it is printed. As long as there exists at least one gray-level difference between the text and its background surface, however, this enhancement will separate the two. A threshold gray level is selected so that the gray level of the text is less and the background more. The result is an output image with the text appearing black and the background white.

Another use for this enhancement is in manufacturing process control applications. Take, for instance, the requirement of automatically measuring the area of a hole in a piece of aluminum. This procedure may be carried out repetitively on specimens traveling down a conveyor under a television camera feeding a simple image processor. The measurement is done by counting the number of pixels that make up the hole. In order to do this, a binary contrast enhancement is performed first. The selection of an appropriate threshold value yields a processed image where the hole appears as black pixels and the rest as white. The hole area may then be calculated by counting all black pixels appearing in the image. This count is readily available from the output image histogram.

Cautions—A difficulty arises with the binary contrast enhancement when the object and its background have common or overlapping gray levels. In this case, it is not possible to choose a threshold value so that all object pixels will go to one level and all background pixels to the other. The resulting output image will not display a good separation between object and background.

Implementation—Single image/pixel point operation. The map will always be a step function, with the transition occurring at the chosen threshold level.

Results

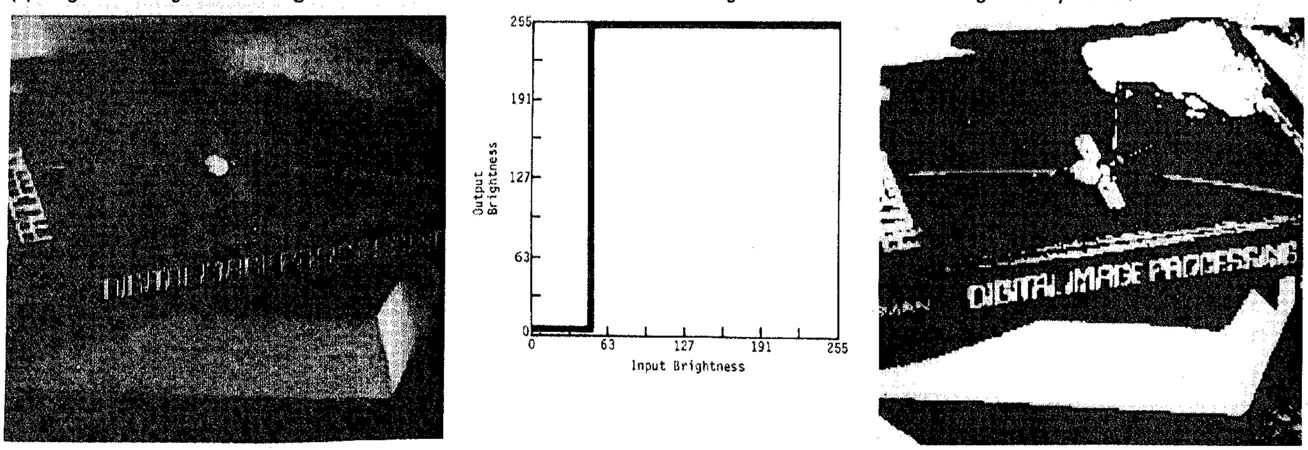


Figure IOS1-1
(a) Original low light level image.
(b) Binary contrast map used to separate book title words from background.
(c) Output image with words "Digital Image Processing" clearly visible.

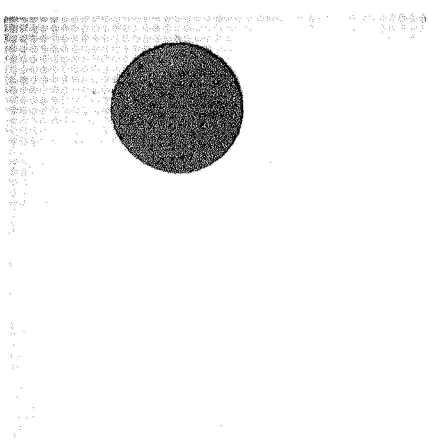
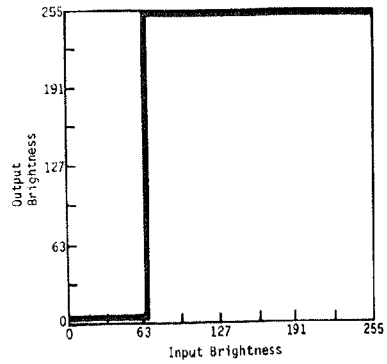
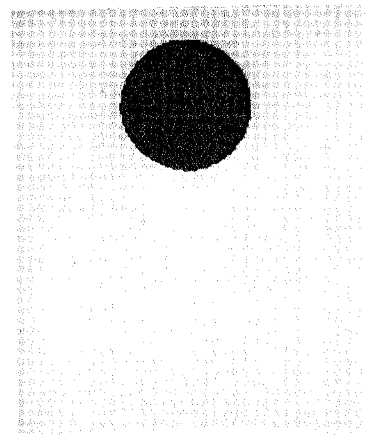


Figure IOS 1-2
(a) Original image of machined part.



(b) Binary contrast map used to highlight machined hole diameter.



(c) Output image.

Image Operation Study # 2

Operation—Histogram Sliding

Description—Histogram sliding is an enhancement used to add or subtract a constant brightness to all pixels within an image. In the context of a histogram, the effect of this operation is to yield an output image with a histogram that is shifted either to the right or left from that of the input image histogram.

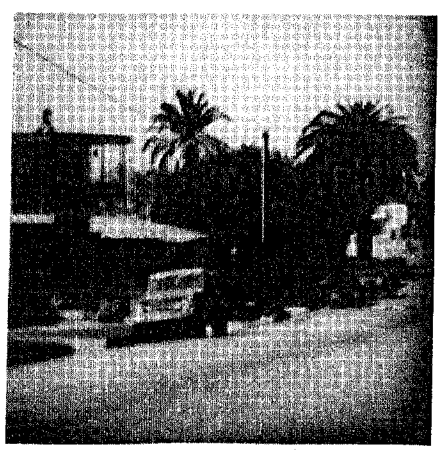
Application—This operation has the visible effect of brightening or darkening an image scene. Since the histogram of the processed image is only shifted and otherwise remains the same, the contrast of the output image will be identical to that of the input image. For use in general contrast enhancement operations, this process may be used to slide the gray level range into a new range facilitating subsequent operations.

Cautions—When doing a histogram slide, be careful of gray level saturation. This occurs when such a constant is added or subtracted that pixels in the input image overflow at the black or white level. For instance, when a value is added to the pixels of an input image the effect is to slide the image histogram to the right. If, however, the slide is too great, pixels will bunch up at the white level. This is because their constant value added to the original gray level totaled to more than the maximum white level. When this happens brightness resolution is lost. Before doing a slide, it is a good idea to preview the input image histogram, taking note of the minimum and maximum gray levels present in the image. To preclude saturation effects, a slide constant must be chosen that will not push the maximum gray level past the full white level or the minimum level past the full black level.

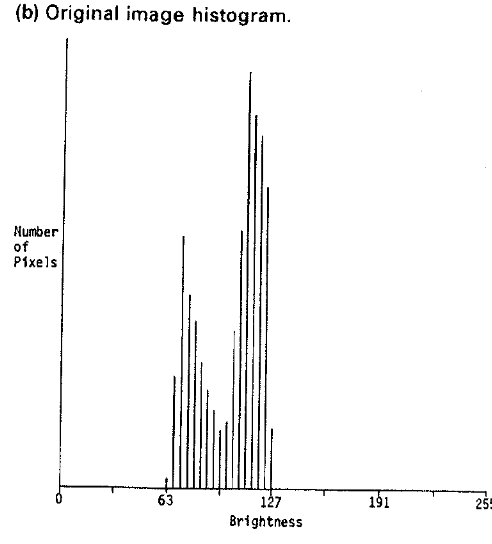
Implementation — Single image/pixel point operation The map will always be a 45° line. In the case of a slide of zero, the line will pass through the origin.

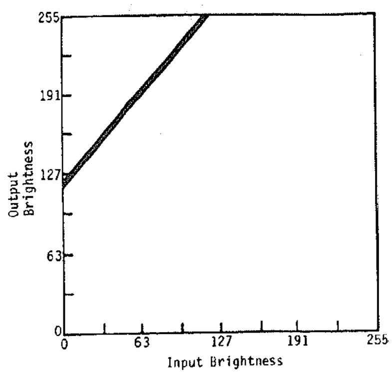
Results

Figure IOS2-1
(a) Original image.



(b) Original image histogram.

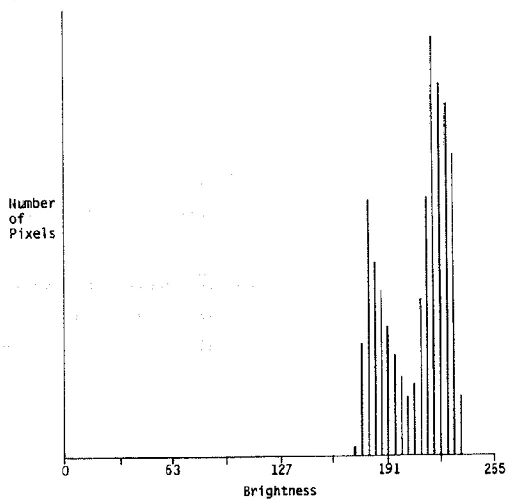




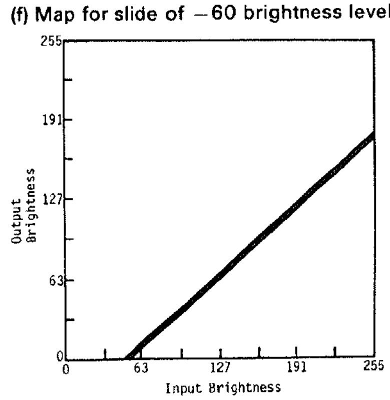
(c) Map for slide of +120 brightness levels.

(d) Result of +120 slide.

(e) Histogram after +120 slide.



(f) Map for slide of −60 brightness levels.



(g) Result of −60 slide.

(h) Histogram after −60 slide.

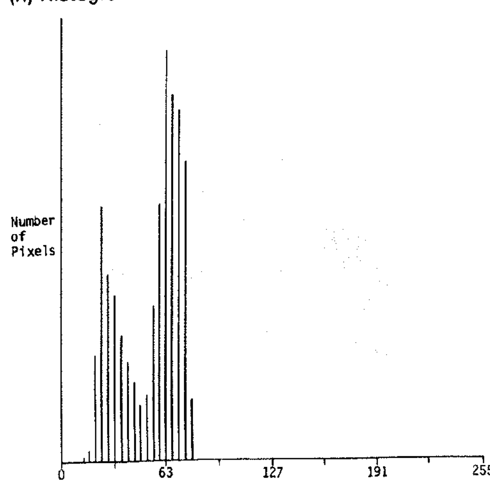


Image Operation Study # 3

Operation—Histogram Stretching

Description—Histogram stretching is a contrast enhancement that allows multiplication and division of all pixel brightnesses within an image by a constant value. This has the effect of stretching or shrinking the input image histogram, thereby controlling the gray level range of the output image histogram.

Application—The apparent effect of this process upon an image is to increase or decrease its contrast. The stretching operation expands or reduces the contrast and dynamic range of the image. General contrast enhancements may be carried out with this operation in conjunction with others such as histogram sliding.

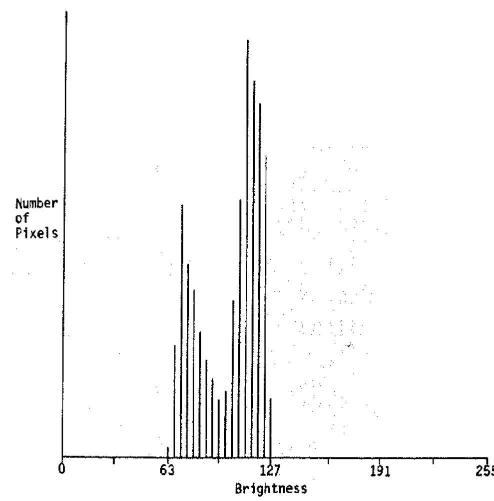
Cautions—Because of the multiplication of pixel brightnesses by constants, it is very easy to overflow the allowed white level in a given gray scale. Care in the selection of multiplicative constants must be exercised so that overflow saturation is prevented. Additionally, an important artifact of the division of pixel brightnesses by a contrast is brightness resolution loss. For instance, dividing pixel brightnesses in an input image by two shrinks the histogram of the image by a factor of two, making it readily apparent that gray scale occupancy has also been halved.

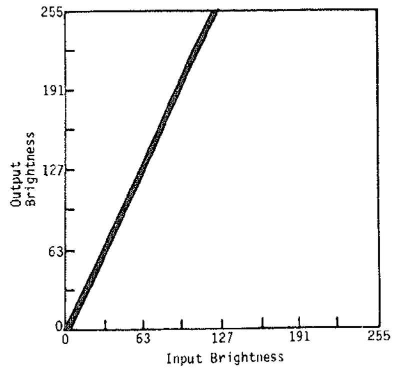
Implementation—Single image/pixel point process This map will always be a straight line passing through the origin. In the case of a stretch by a factor of 1, the line will be 45°.

Results

Figure IOS3-1
(a) Original image.

(b) Original image histogram.

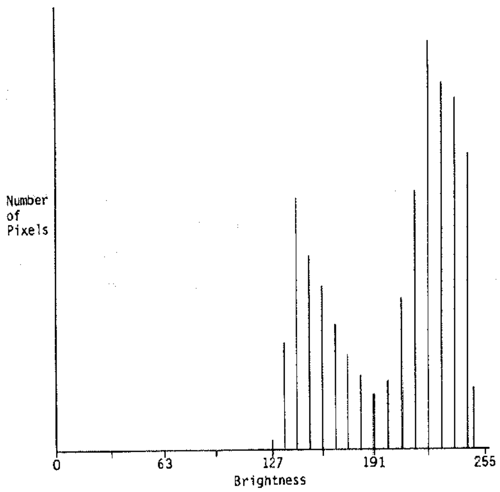




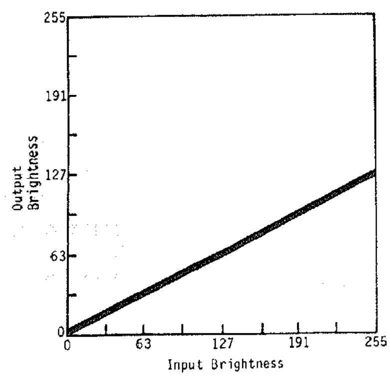
(c) Map for brightness stretch of 2.

(d) Result of times 2 stretch.

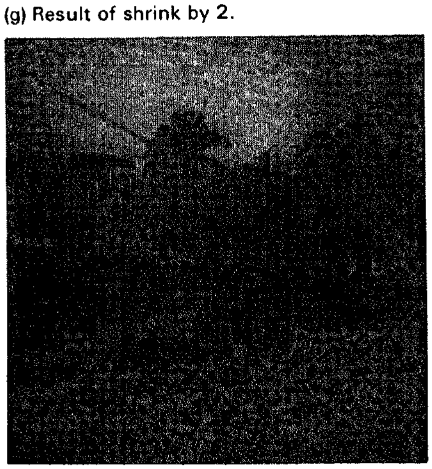
(e) Histogram after times 2 stretch.



(f) Map for brightness shrink by factor of 2.



(g) Result of shrink by 2.



(h) Histogram after shrink by 2.

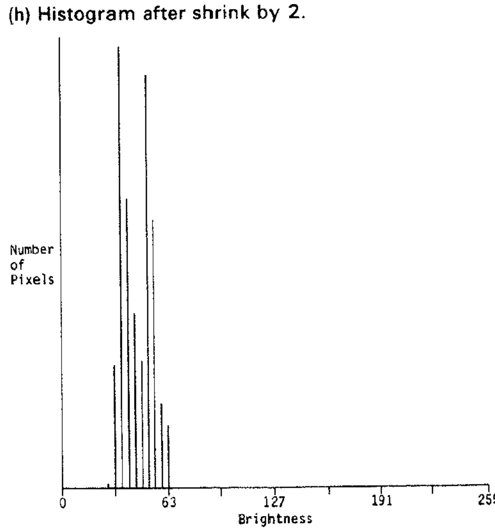


Image Operation Study #4

Operation—Contrast Enhancement

Description—The common case of contrast enhancement is made up of a combination of both the histogram slide and stretch operations. By implementing both processes in one operation, we may produce an output image having a histogram dictating an image with the subjective attributes of good contrast.

Application—General contrast enhancement serves to improve an image based on its contrast and dynamic range characteristics. This type of processing is often used to correct an image for such things as poor exposure and improper scene illumination. An important need for contrast enhancement is in the clean-up of an image preceding or following other processing. Often the results of other operations will leave artifacts based on the algorithms employed. These artifacts, when undesired, may often be corrected for using basic contrast enhancement techniques.

Cautions—As seen in the histogram slide and stretch examples, careful selection of constants is required to preclude overflow and gross resolution loss conditions.

Implementation—Single image/pixel point operation The map is composed by the combination of two LUTs as derived from appropriate slide and stretch operations; see histogram slide and histogram stretch examples.

Results

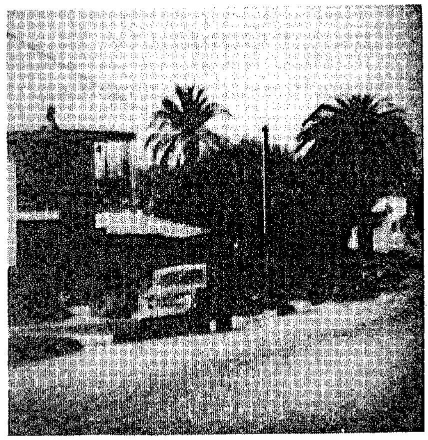
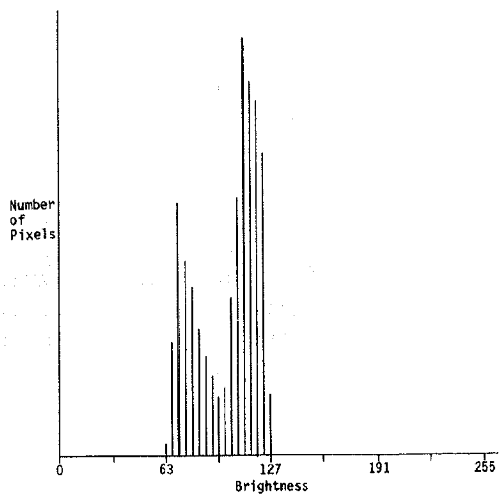
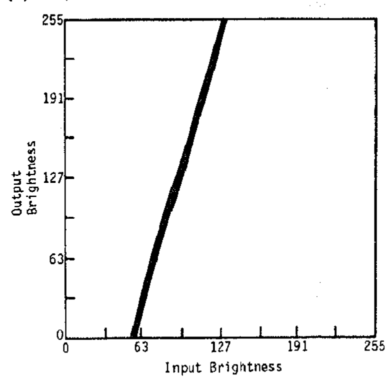


Figure IOS4-1
(a) Original image.

(b) Original image histogram.



(c) Map for slide of −60 and stretch of 4.



(d) Result of slide and stretch.

(e) Histogram after slide and stretch.

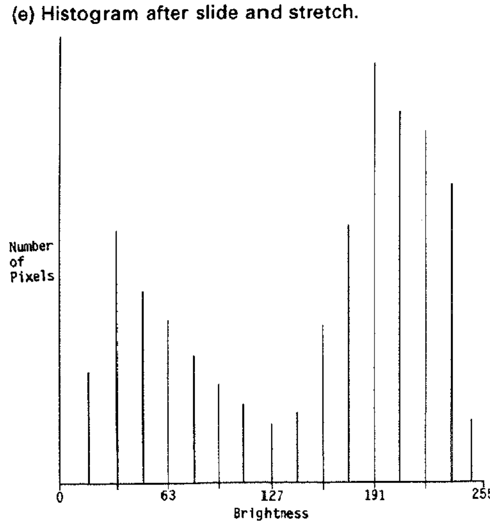


Image Operation Study # 5

Operation—Complement Image

Description—A complement image is an image where all pixel brightnesses are logically complemented, or reversed. Black pixels become white, white pixels become black, and the gray levels in between take on their respective reverse brightnesses. The end result is a negative image having the same appearance as a film negative of the original.

Application—This process is often useful in the analysis of an image. Because the eye responds logarithmically to brightness changes, details characterized by small brightness deviations in the white regions may be undetectable. Complementing the image converts these small changes in the white regions to the black regions, where they are more visible. Although the image takes on a rather unnatural appearance, analysis of fine details hidden in bright regions is often aided.

Partial complementing of an image also produces results helpful to the analyst. An example would be to leave the lower half of the gray scale untouched while complementing the upper half. Dark regions in the original image are unaffected while bright regions are complemented down into the dark regions where details are often more visible.

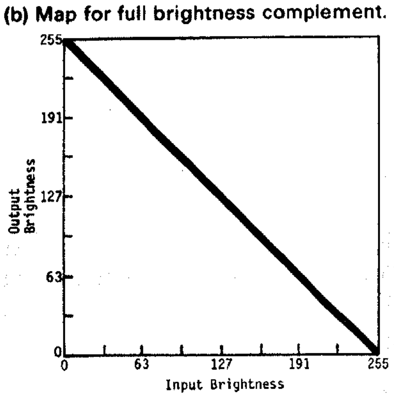
Cautions—The process of partial image complementing may be tailored to give the best results for a given application. The area of the gray scale to be complemented may usually be chosen by trial techniques. The final selection is left up to visual comparison.

Implementation—Single image/pixel point process The portion of the gray scale to be complemented uses a map appearing as a 45° line traversing downward, left to right.

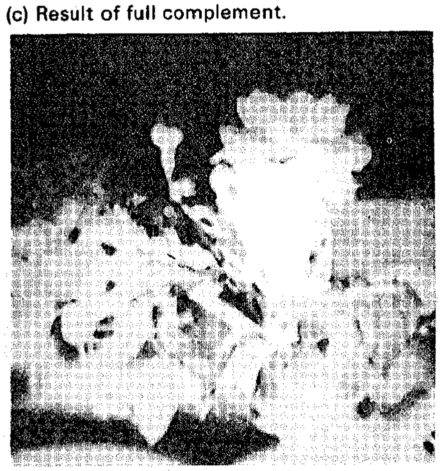
Results

Figure IOS5-1
(a) Original image of flowers.

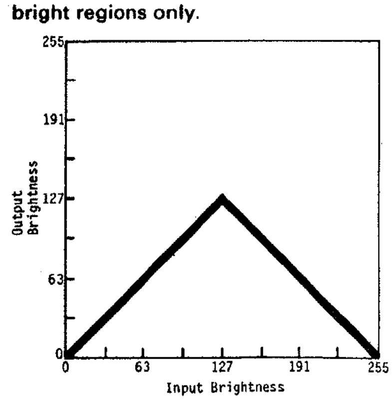
(b) Map for full brightness complement.



(c) Result of full complement.



(d) Map for complement of bright regions only.



(e) Result of bright region complement.

Image Operation Study # 6

Operation—Low Pass Filter

Description—Low pass filtering of an image produces an output image in which high spatial frequency components have been attenuated. The cutoff point at which higher frequencies become attenuated is varied by the selection of mask weighting coefficients.

Application—Low pass filtering is employed as a smoothing operation to reduce high spatial frequency components that may be present in an image. In particular, this operation is useful in removing visual noise, which generally appears as sharp bright points. Because of the very high spatial frequency in these "spikes," the low pass filter acts well in their attenuation.

Low pass filters also are used to simply reduce high-frequency components in order to more closely examine the low-frequency content of an image. For instance, say the object of interest in an image is a cloud formation and a distracting telephone cable appears silhouetted in front. We may use a low pass filter to attenuate the cable while leaving the cloud relatively untouched. This is because the cloud is basically composed of low-frequency components and the cable contains high-frequency components. This filtering operation will attenuate the cable with little effect to the cloud.

Cautions—Because of the limitations of a 3 × 3 kernel size in carrying out group processes, the spatial frequency at which the low pass filter begins attenuation is not highly selectable. As a result, the output image will often be either over- or under-filtered for a given application. For this reason, it is easy to lose high-frequency information that was intended to be retained. The optimum mask to be used is a tradeoff issue, usually determined by visual inspection of the results.

Implementation—Group process The mask coefficients will always add to 1. All coefficients are positive and, therefore, fractional. Three typical masks are given:

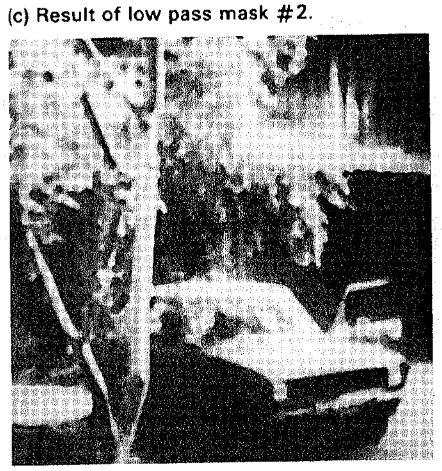
1/9	1/9	1/9	1/10	1/10	1/10	1/16	1/8	1/16
1/9	1/9	1/9	1/10	1/5	1/10	1/8	1/4	1/8
1/9	1/9	1/9	1/10	1/10	1/10	1/16	1/8	1/16
	Mask 1			Mask 2			Mask 3	

Results

Figure IOS6-1
(a) Original image.

(b) Result of low pass mask #1.

(c) Result of low pass mask #2.



(d) Result of low pass mask #3.

Image Operation Study # 7

Operation—High Pass Filter

Description—High pass filtering of an image produces an output image in which high spatial frequency components are accentuated. The cutoff point at which higher frequencies become accentuated is varied by the selection of mask weighting coefficients.

Application—High pass filtering is used in the enhancement of edges and other high-frequency components within an image. Images that do not appear clear may be sharpened by high pass filtering. The sharpness of an image is related to the content of high spatial frequency components within the image. Therefore, the high pass filter serves well in the enhancement.

Additionally, this filter is useful any time it is desired to enhance high spatial frequencies for viewability of certain features. An example of this might be the enhancement of a spatially detailed object—say, a tree appearing in front of flat surface such as dirt-covered ground. The effect of the filter will be to enhance the tree while leaving the ground relatively untouched.

Cautions—As in the case of the low pass filter, cutoff frequency is not highly selectable when using the standard 3 × 3 kernel size. The final mask selection most appropriate to a given application is generally determined by comparative visual inspection.

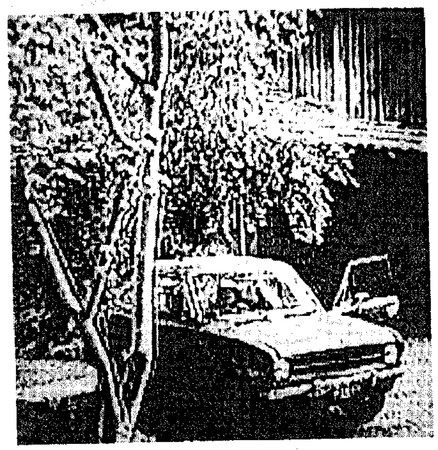
Implementation—Group process The mask coefficients always add to 1. A large coefficient will generally appear in the center of the mask surrounded by smaller positive and negative coefficients. Three typical masks are given.

$$\begin{matrix} -1 & -1 & -1 \\ -1 & 9 & -1 \\ -1 & -1 & -1 \end{matrix} \qquad \begin{matrix} 0 & -1 & 0 \\ -1 & 5 & -1 \\ 0 & -1 & 0 \end{matrix} \qquad \begin{matrix} 1 & -2 & 1 \\ -2 & 5 & -2 \\ 1 & -2 & 1 \end{matrix}$$

Mask 1 Mask 2 Mask 3

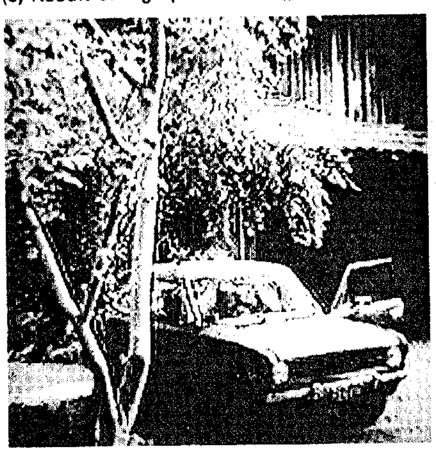
Results

Figure IOS7-1
(a) Original image.



(b) Result of high pass mask #1.

(c) Result of high pass mask #2.



(d) Result of high pass mask #3.

Image Operation Study # 8

Operation—Median Filter

Description—The median filter is a group-like process operating on a 3 × 3 pixel neighborhood. It does not fall within the category of a spatial convolution, however, because its result is not based on a weighted average of the nine pixels in the neighborhood. Instead, the output of the median filter is the median value of the nine pixel brightness values. The median value of a neighborhood is determined by placing the nine brightnesses into ascending numerical order and selecting the center value such that four values are less than or equal to and four greater than or equal to the center value. This center value becomes the output pixel in the calculation.

Application—The primary use for the median filter is in pictorial noise removal. Noise spikes appear as bright pixels randomly distributed across an image. Since these spikes are bright in comparison with their neighbors, they generally end up near the top when the nine pixels are placed in ascending order. The median value tends to reduce any influence from the spike itself.

An interesting use for this filter is in art applications. When applied several times, an image begins to look as if it were hand painted. Details tend to look brushed on. Using the median filter, real-life images may be processed into synthetic paintings. Adding the effects of other operations may generate additional interesting effects.

Cautions—The median filter is somewhat unpredictable, often leaving an image worse off than it was initially. The greatest loss is suffered in details composed of high spatial frequencies. These regions become somewhat distorted. The tradeoff is worth it, though, if an image is badly speckled with white noise spikes, because the resulting output image will be virtually spike-free.

Implementation—Group process (frame process) The median filter operates similar to a group process, deriving its results from 3 × 3 neighborhoods of pixels. However, it is not a convolution operation and is therefore not handled by hardware used in general group processing. For this reason, the median filter is most often implemented in software through a host computer. Since the group process hardware is not used, it is probably more suitable to call the operation a frame process. Spike suppression is illustrated on a typical 3 × 3 neighborhood. All of the brightnesses in the group are between 5 and 20 except for the center pixel, which is 210. This "white spike" is ultimately replaced with the value of 15, eliminating its distracting effect on the image.

Pixel brightness in a 3×3 group			Brightness values placed in ascending order
10	20	5	
15	210	20	5 10 10 15 15 20 20 20 210
15	10	20	↑ Median value = Result to replace group center pixel

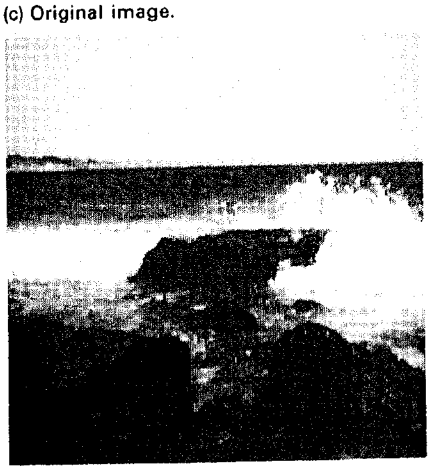
Results

Figure IOS8-1

(a) Original image with noise spikes.

(b) Result after median filtered.

(c) Original image.



(d) Result after application of the median filter twice — the painted appearance is evident.

Image Operation Study # 9

Operation—Unsharp Masking Enhancement

Description—The unsharp masking enhancement produces an output image in which high-frequency details are improved. Lower-frequency regions of the image are left basically untouched. This process generally yields a more subtle and visually appealing enhancement than that of the high pass filter. It is interesting to note that this operation is borrowed from the photographic darkroom, where it is commonly used.

The enhancement is formed by the subtraction of a low pass filtered image from its original. The idea being that edges in the low passed image are changing more slowly than their counterparts in the original. When a slower-changing transition is subtracted from a faster one, the result is a transition with overshoot and undershoot. This translates to edge accentuation. This phenomenon is illustrated:

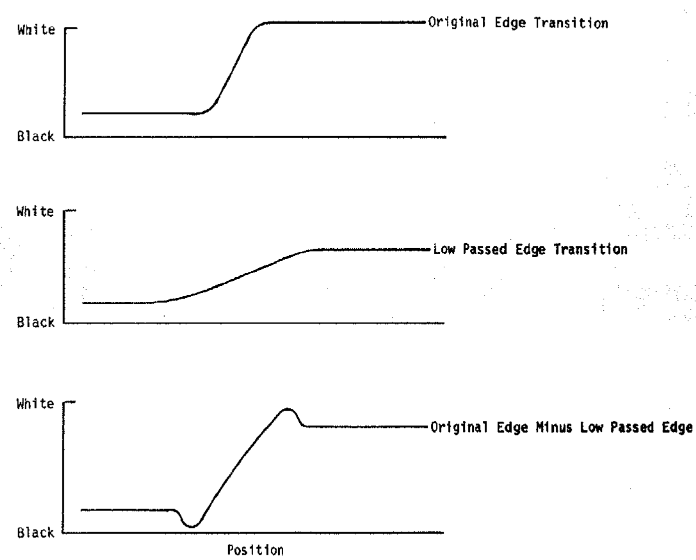


Figure IOS9-1
Unsharp masking edge accentuation.

Prior to subtraction, the low passed image may be brightness scaled by a histogram shrink (stretch) operation. This has the effect of controlling the amount of accentuation to occur in the enhancement.

Application—The unsharp masking operation produces an output image that appears sharpened, remaining visually appealing. Because of this, it is often employed to simply sharpen images suffering from some sort of blurring. Examples of blurred cases may include images exhibiting very poor contrast, haze obstruction, or even minor defocus.

Cautions—The variables in this process are (1) the extent of low pass filtering to be evoked upon the original image before subtraction and (2) the amount of brightness scaling to be used on the low passed image before subtracting it from the original. The amount of low pass filtering affects the amount of the over- and undershoot components in the final image transitions. Brightness scaling has an impact on how accentuated the edges will appear and how attenuated the low-frequency regions will be. For instance, large brightness scaling will leave the original minimally changed in low-frequency areas but also will not accentuate the edges to a high degree. On the other hand, no scaling allows the low-frequency regions to be almost entirely subtracted out to black while the edges will be highly accentuated. A good tradeoff brightness scaling factor must be chosen with care to effect the desired results.

Implementation—Frame process, Single image/pixel point process, Dual image/pixel point process The unsharp masking enhancement uses a group process to carry out the low pass filter operation. An appropriate low pass mask may be chosen. The low pass image is brightness scaled by a histogram shrink operation using a single image/pixel point process. The low pass image is then subtracted from the original, using a dual image/pixel point process. The resulting image may then be contrast enhanced by once again using a histogram stretch operation. This process is outlined:

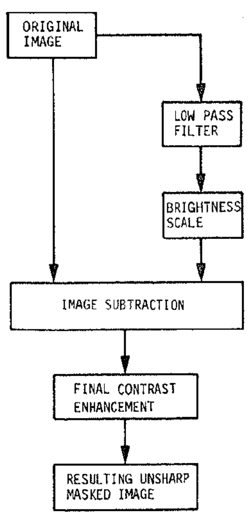
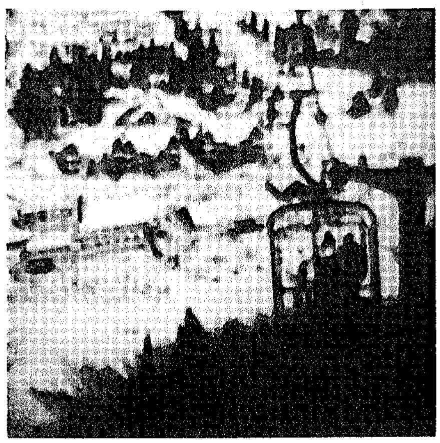


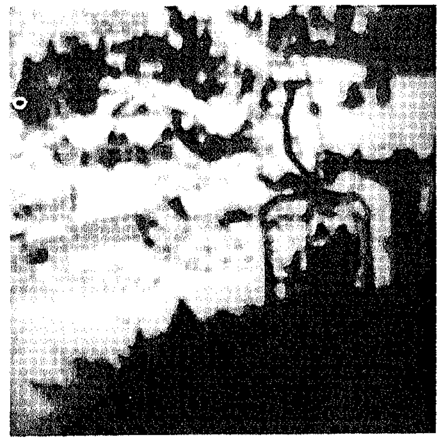
Figure IOS9-2 Unsharp masking operation flow diagram.

Results

Figure IOS9-3



(a) Original image displaying slight haze.



(b) Low passed image.

(c) Result of subtracting scaled low passed image from original, creating the final unsharp masked image.

Image Operation Study # 10

Operation—Shift and Difference Edge Enhancement

Description—The shift and difference operation provides the simplest means for generating edge enhancement in an image. Three edge enhancement options are available—vertical, horizontal, and the combination of the two. Vertical edge enhancement is accomplished by shifting the original image, pixel by pixel, to the left by one pixel. The shifted image is then subtracted from the original. This produces the brightness difference, or slope, of the two pixels. Whenever horizontally adjacent pixels were nearly the same brightness value in the original image, the output image will be dark. At a vertical edge, however, a large difference exists and will appear bright in the output image. The result is an image where vertical edges composed of black-to-white transitions in the left-to-right direction will appear bright. Constant or slowly changing brightness regions will appear dark.

Horizontal edge enhancement is formed by shifting the original image up by one pixel and carrying out the subtraction. Simultaneous vertical and horizontal enhancement requires a one-pixel shift left and up prior to the subtraction.

Application—The shift and difference enhancement is useful in the general extraction of vertical or horizontal edges. This enhancement can be of help in the analysis of an image having poor edge detail.

The output image of this enhancement takes on the appearance of a relief map of the original image. By adding a brightness scaled amount of the result back to the original, a uniquely highlighted image is produced.

Cautions—Because this operation subtracts one image from another, there is always the possibility of subtracting a bright level from a dark level. The result is an arithmetic underflow condition, which is usually handled by forcing the result to black. The net effect in a horizontal edge enhancement is the highlighting of edges composed of black-to-white transitions in the left-to-right direction. A white-to-black transition will not be enhanced because the arithmetic underflow of subtracting white from black will produce black as a result. Likewise, vertical edge enhancement will highlight black-to-white transitions in the top to bottom direction.

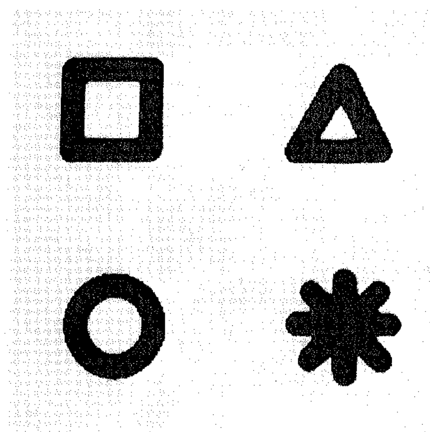
Implementation—Group process The most straightforward method in which to carry out this process is by actually shifting the original image and subtracting it from the original. However, the shifting process is often more difficult for an image processing system to handle than it may seem. The alternative method is done by way of a group process. The center coefficient is set to 1 and another coefficient to -1. The rest are left at zero. The overall effect is the subtraction of a shifted image from its original. The mask coefficients always add to 0.

The three masks are given:

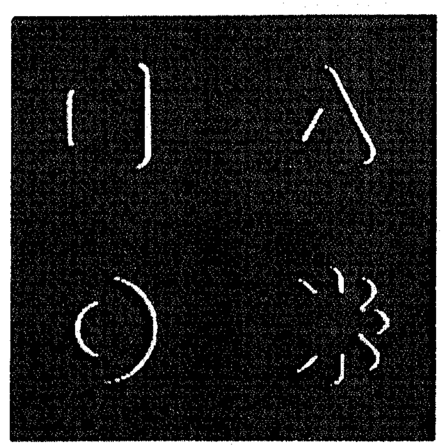
$\begin{matrix} 0 & 0 & 0 \\ -1 & 1 & 0 \\ 0 & 0 & 0 \end{matrix}$	$\begin{matrix} 0 & -1 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{matrix}$	$\begin{matrix} -1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{matrix}$
Vertical edge enhancement	Horizontal edge enhancement	Vertical and horizontal edge enhancement

Results

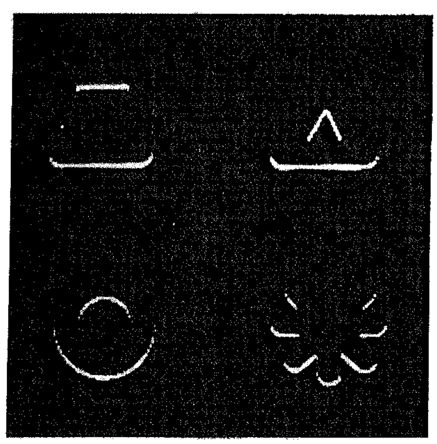
Figure IOS10-1
(a) Original pattern image.



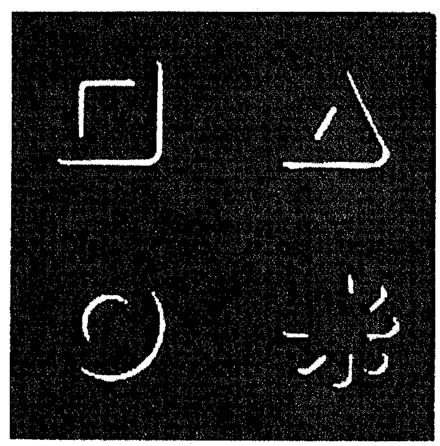
(b) Vertical shift and difference edge enhancement.



(c) Horizontal.



(d) Vertical and horizontal.



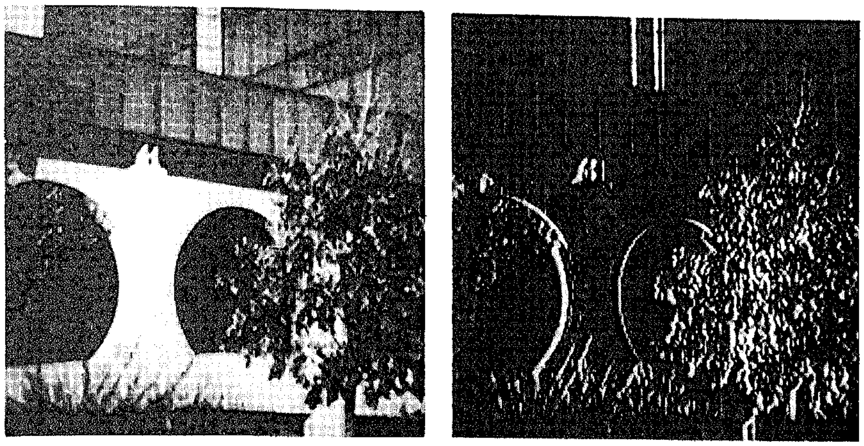


Figure IOS 10-2
(a) Original image of building.
(b) Vertical shift and difference edge enhancement.

(c) Horizontal.
(d) Vertical and horizontal.

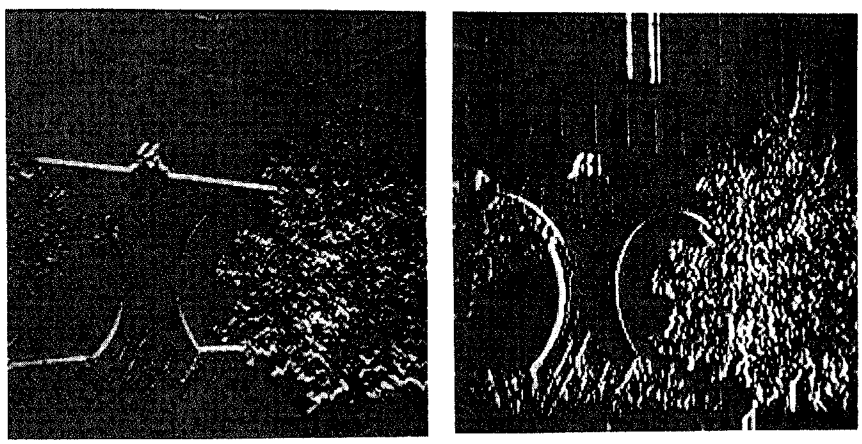


Image Operation Study # 11

Operation—Gradient-Directional Edge Enhancement

Description—The gradient filter produces an output image where high spatial frequency components, such as edges, are highly accentuated. Low-frequency spatial components are attenuated sharply. The gradient enhancement is directional. It may be selected to occur in any one of the eight compass directions—N, NE, E, SE, S, SW, W, or NW. Edges composed of black-to-white transitions, in the defined direction, are enhanced to white. The level of white is dependent upon the amount of brightness change encountered in the original image. Constant brightness regions become black in the output image. The end result is a black background with white outlines of the objects in the original image.

Application—The gradient enhancement is useful in the extraction of object edges when the direction of the edges is of importance. This is often helpful in the analysis of images. By independently generating eight gradient images, an original image is broken into spatial directional parts. A host computer may then make judgements about object edge boundary directions within the image. Additionally, complex images may conceal object directional information that becomes evident once processed by the gradient operation. Such an application is in aerial geological surveys.

Cautions—The gradient operation produces a visual abstract of the original image. The process enhances edge information in a defined direction and, therefore, does not always bear a great resemblance to the original. The extracted information is used either in the enhancement of the original or in some image-analysis operation. As in other group processes, the 3 × 3 kernel size poses some restrictions. For the gradient operation, one limitation is that only eight directions of enhancement are available. Also, the angular width of enhancement for any selected direction is rather wide. Most applications have no trouble working within these limitations. If needed, though, a larger kernel size will provide additional flexibility.

Implementation—Group process The mask coefficients always add to 0. The eight directional masks are given.

$$\begin{matrix}1 & 1 & 1\\ 1 & -2 & 1\\ -1 & -1 & -1\end{matrix}\quad\begin{matrix}1 & 1 & 1\\ -1 & -2 & 1\\ -1 & -1 & 1\end{matrix}\quad\begin{matrix}-1 & 1 & 1\\ -1 & -2 & 1\\ -1 & 1 & 1\end{matrix}\quad\begin{matrix}-1 & -1 & 1\\ -1 & -2 & 1\\ 1 & 1 & 1\end{matrix}$$

N NE E SE

$$\begin{matrix}-1 & -1 & -1\\ 1 & -2 & 1\\ 1 & 1 & 1\end{matrix}\quad\begin{matrix}1 & -1 & -1\\ 1 & -2 & -1\\ 1 & 1 & 1\end{matrix}\quad\begin{matrix}1 & 1 & -1\\ 1 & -2 & -1\\ 1 & 1 & -1\end{matrix}\quad\begin{matrix}1 & 1 & 1\\ 1 & -2 & -1\\ 1 & -1 & -1\end{matrix}$$

S SW W NW

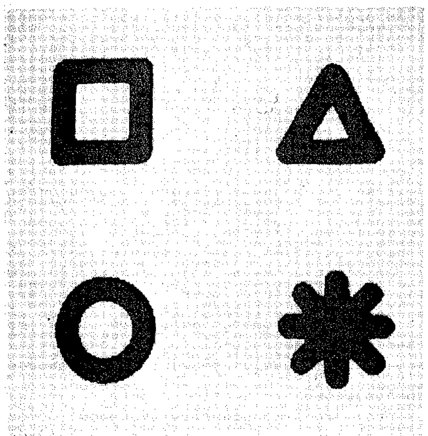
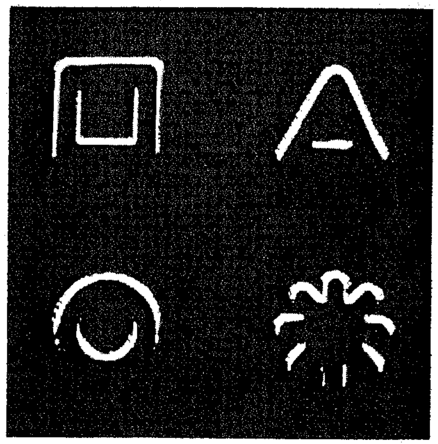
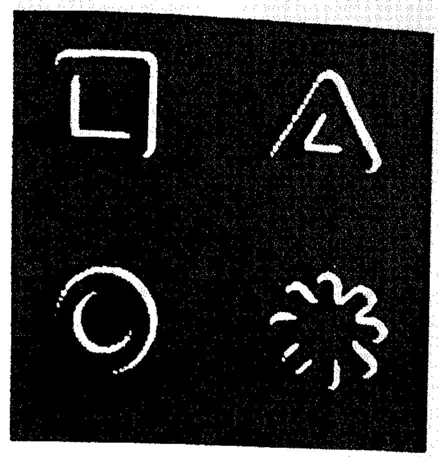


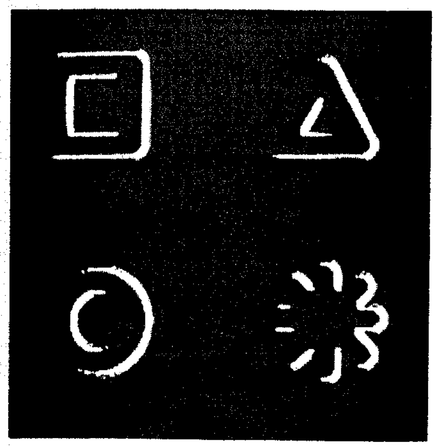
Figure IOS11-1
(a) Original pattern image.



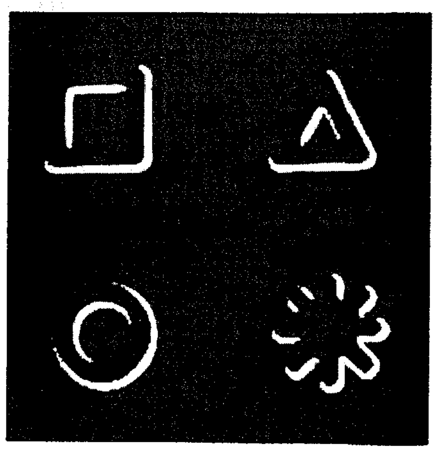
(b) North direction gradient edge enhancement.



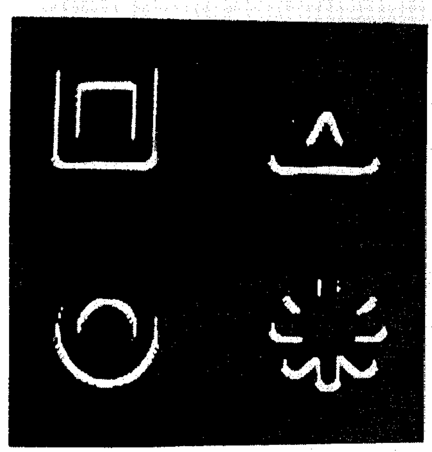
(c) Northeast.



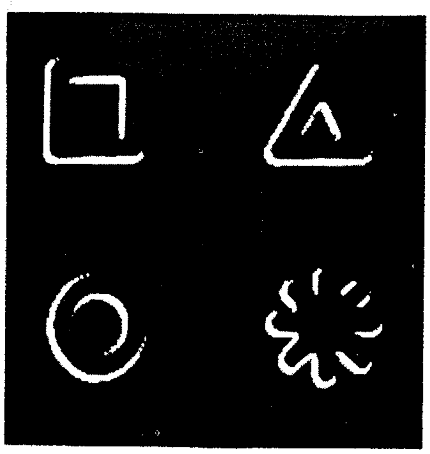
(d) East.



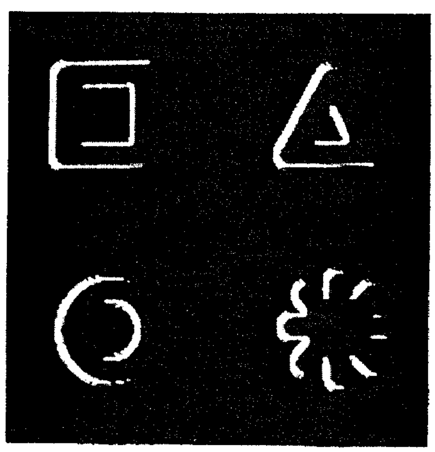
(e) Southeast.



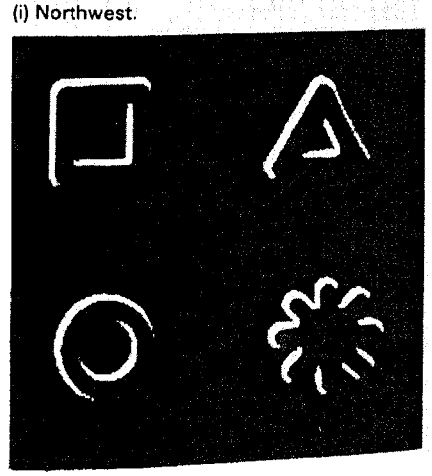
(f) South.



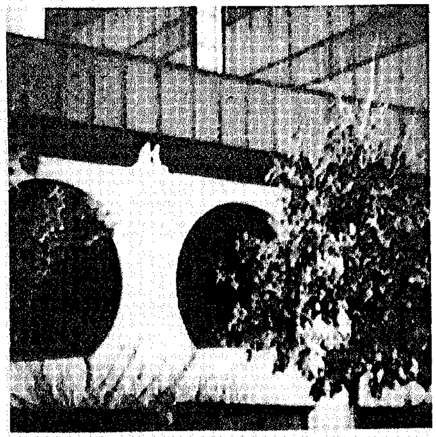
(g) Southwest.



(h) West.



(i) Northwest.



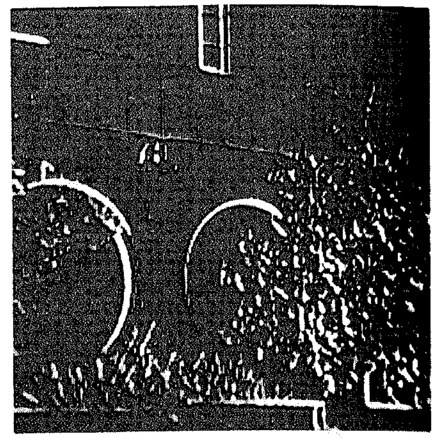


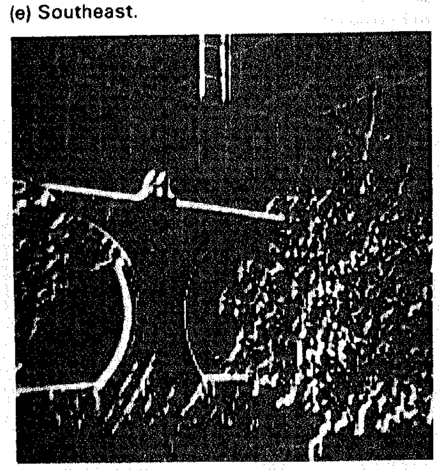
Figure IOS11-2
(a) Original image of building.

(b) North direction gradient edge enhancement.

(c) Northeast.

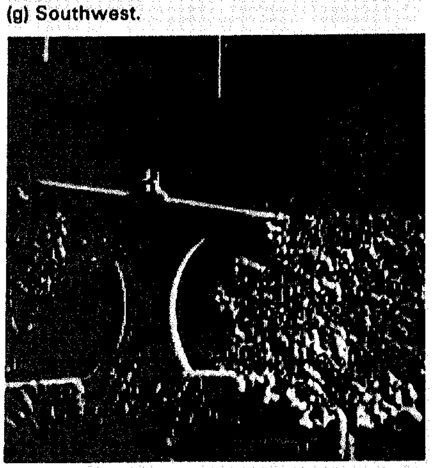
(d) East.

(e) Southeast.



(f) South.

(g) Southwest.



(h) West.

(i) Northwest.

Image Operation Study #12

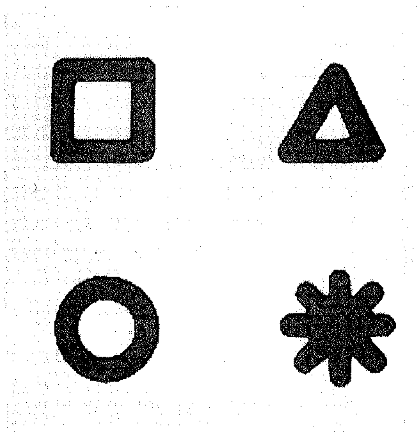
Operation—Laplacian Edge Enhancement

Description—The Laplacian edge enhancement produces an output image where high spatial frequency components such as edges are highly accentuated. Low spatial frequency components are attenuated sharply. The enhancement is omnidirectional. Edges composed of black-to-white transitions, in any direction, are enhanced to white. The level of white is dependent on the amount of brightness change encountered in the original image. Constant and linearly increasing or decreasing brightness regions become black in the output image. The end result is a black background with white outlines of the objects in the original image.

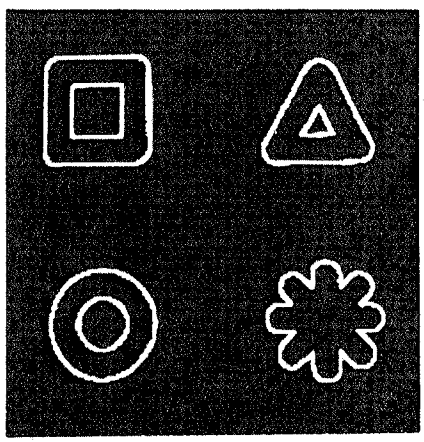
Application—The Laplacian filter is used in the extraction of object edges, or boundaries. Pattern recognition algorithms for use in robotics control often begin processing with a Laplacian operation.

For general sharpening of a poor image, good results are often obtained by adding a portion of the Laplacian of an image back to its original. The result is a highlighting of object boundaries within the image.

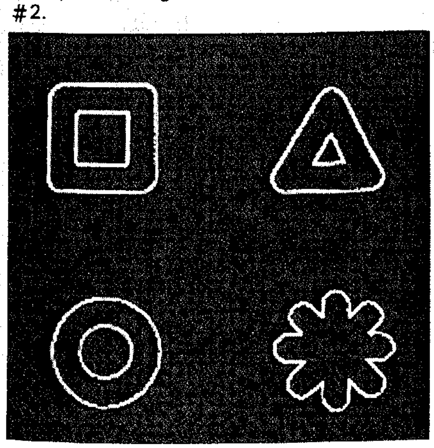
Figure IOS 12-1
(a) Original pattern image.



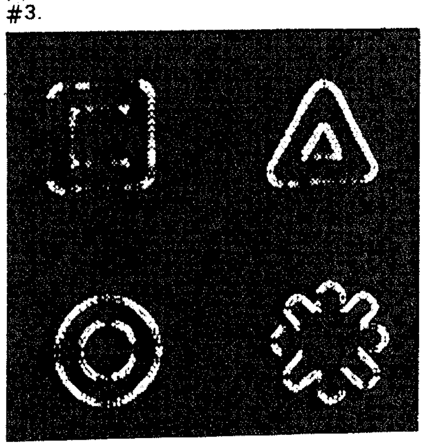
(b) Laplacian edge enhancement—mask #1.



(c) Laplacian edge enhancement—mask #2.



(d) Laplacian edge enhancement—mask #3.

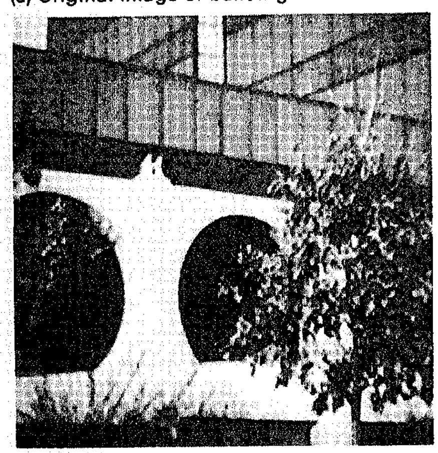


Cautions—The Laplacian operation produces a visual abstract of the original image. The process enhances edge information and, therefore, does not always bear a great resemblance to the original. The extracted edge information may then be used in either the enhancement of the original or in some image analysis operation. As in other group processes, the 3 × 3 kernel size poses some restrictions on the frequency selectivity of the filter. By tailoring the mask coefficients and viewing the results, a suitable filter may be obtained.

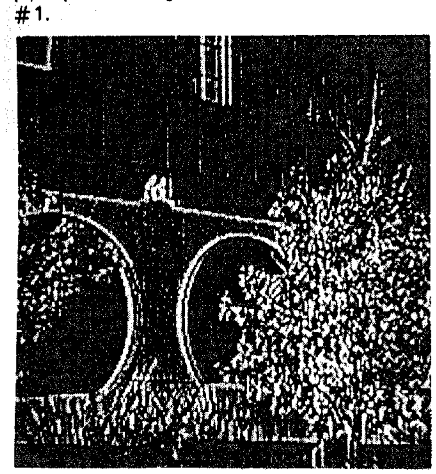
Implementation—Group process The mask coefficients always add to 0. A large coefficient will generally appear in the center of the mask surrounded by smaller positive and negative coefficients. Three typical masks are given.

−1	−1	−1		0	−1	0		1	−2	1
−1	8	−1		−1	4	−1		−2	4	−2
−1	−1	−1		0	−1	0		1	−2	1
Mask 1				Mask 2				Mask 3		

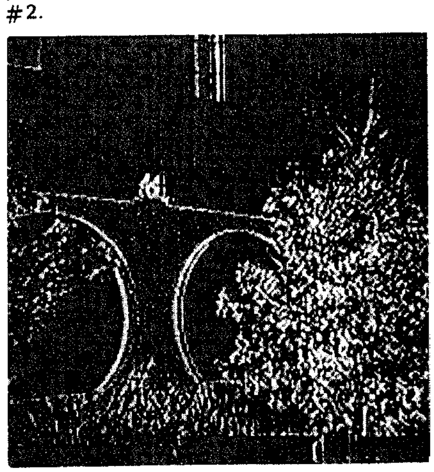
Figure IOS12-2
(a) Original image of building.



(b) Laplacian edge enhancement—mask #1.



(c) Laplacian edge enhancement—mask #2.



(d) Laplacian edge enhancement—mask #3.

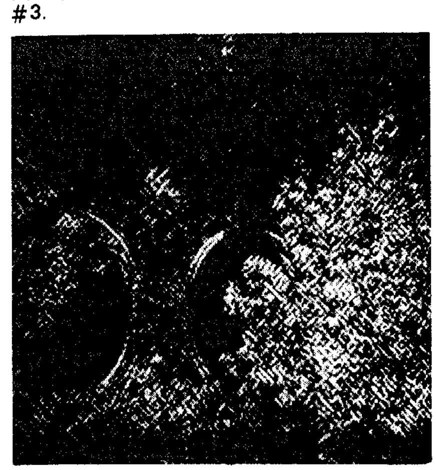


Image Operation Study # 13

Operation—Line Segment Enhancement

Description—Line segment enhancement produces an output image similar to edge enhancement operations. The change is that in addition to directional edge enhancement, the highlighted edge segments are produced in a more connected manner. The output image is a line segment enhancement in one of four directions—vertical, horizontal, left-to-right diagonal, or right-to-left diagonal.

In vertical line segment enhancement, two slopes are calculated for each row in the 3×3 kernel. The left neighboring pixel is subtracted from the center pixel, as is the right neighboring pixel. These two slopes are then subtracted from one another forming a slope-difference. Applied to the three rows of pixels in the 3×3 kernel, the sum of these operations yield the final result of the output pixel.

In the vertical enhancement, the net result is an image where all vertical edges appear bright and connected. Constant or slowly changing brightness regions appear dark. Horizontal and diagonal line segment enhancements are formed by changing the orientation of the group process mask.

Application—Line segment enhancements are useful in the extraction of edge information when it is desired to have edge segments as connected as possible. Where other edge enhancements tend to leave highlighted edges in a broken form, this enhancement attempts to connect severed segments. Additional passes of this enhancement will further connect broken segments. Ultimately, this type of operation acts as a good prelude to automatic machine inspection of parts where dimensions are to be measured from edge to edge.

Cautions—This enhancement process may generally be implemented without regard to downfalls. The final applicability to a particular application may be made by visual comparison with other processes.

Implementation—Group process The mask coefficients always add to 0. The four directional masks are given.

$$\begin{matrix} -1 & 2 & -1 \\ -1 & 2 & -1 \\ -1 & 2 & -1 \end{matrix}$$

Vertical line segment enhancement

$$\begin{matrix} -1 & -1 & -1 \\ 2 & 2 & 2 \\ -1 & -1 & -1 \end{matrix}$$

Horizontal line segment enhancement

$$\begin{matrix} -1 & -1 & 2 \\ -1 & 2 & -1 \\ 2 & -1 & -1 \end{matrix}$$

Left-to-right diagonal line segment enhancement

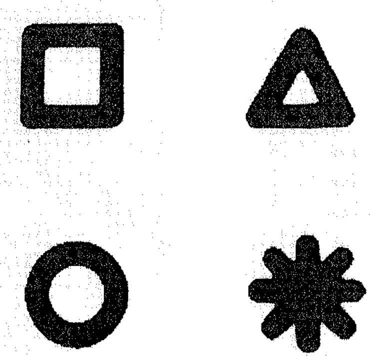
$$\begin{matrix} 2 & -1 & -1 \\ -1 & 2 & -1 \\ -1 & -1 & 2 \end{matrix}$$

Right-to-left diagonal line segment enhancement

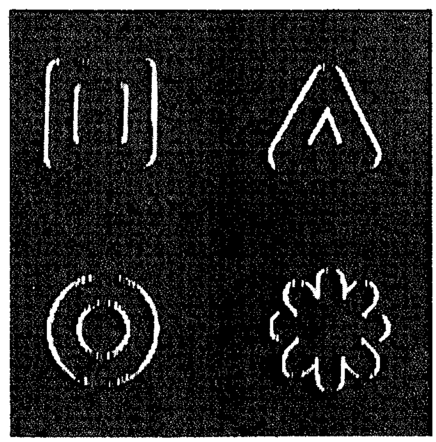
Results

Figure IOS13-1

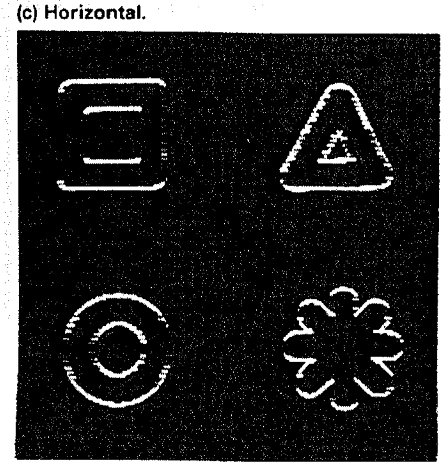
(a) Original pattern image.



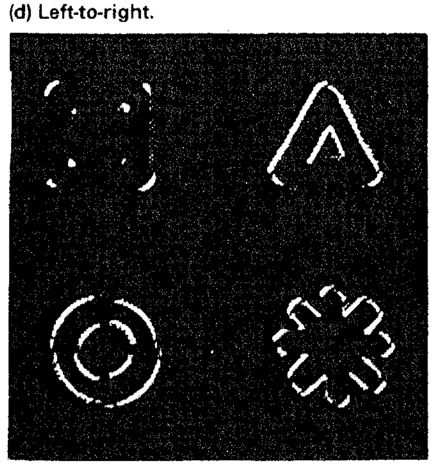
(b) Vertical line segment enhancement.



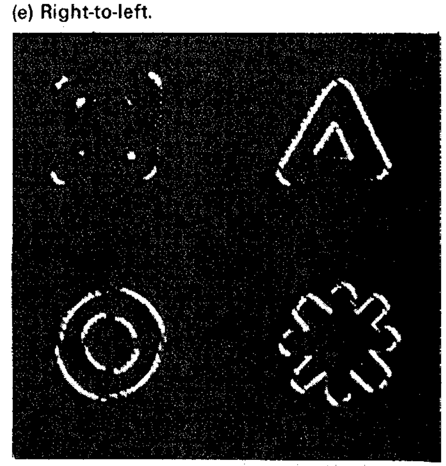
(c) Horizontal.



(d) Left-to-right.



(e) Right-to-left.



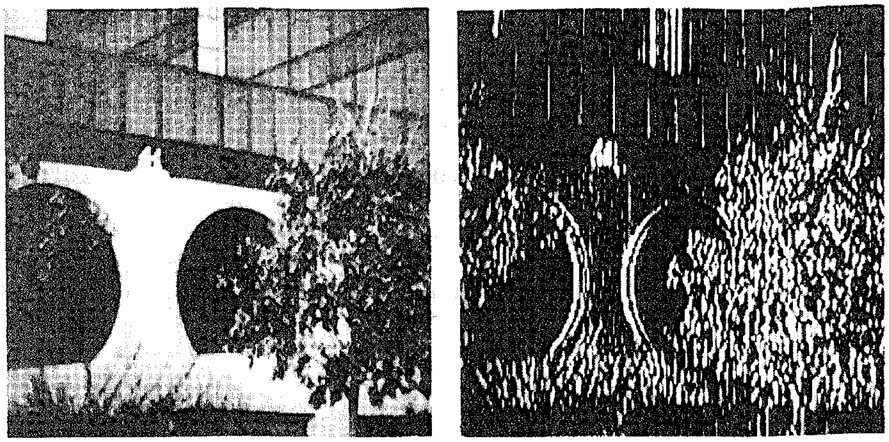


Figure IOS13-2
(a) Original image of building. (b) Vertical line segment enhancement.

(c) Horizontal. (d) Left-to-right. (e) Right-to-left.

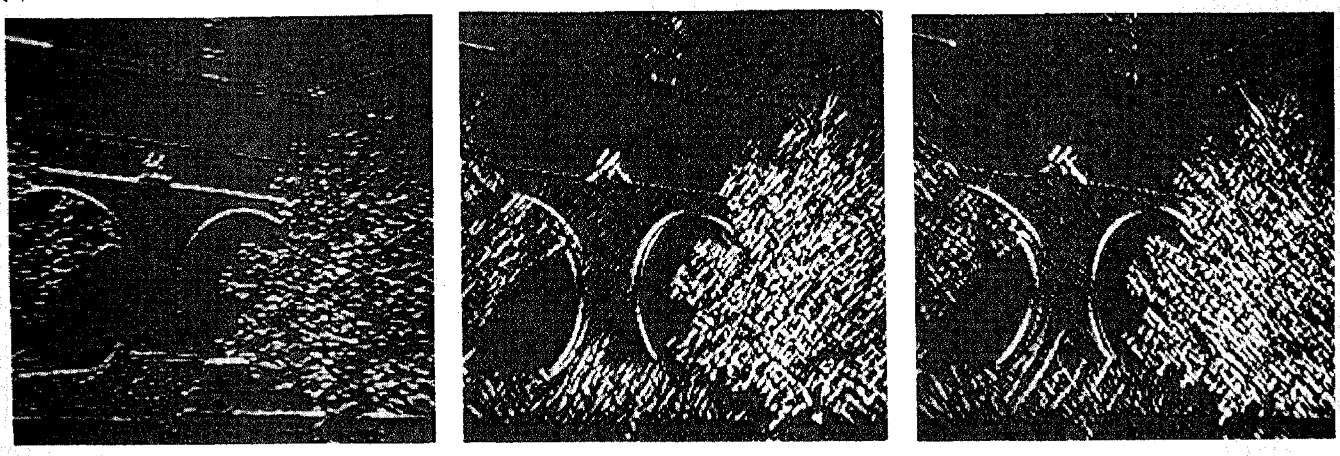


Image Operation Study # 14

Operation—Image Scaling

Description—Image scaling allows the spatial enlargement and reduction of image size. A scaling factor, S, is selected. For an $S = 1$ the output image is identical to the original. An S less than 1 represents an image size reduction, whereas an S greater than 1 calls for an enlargement. In all cases, S is the scaling factor to occur in both horizontal and vertical directions.

To produce an image reduced in size by a factor of 3, $S = 1/3$ is selected. The output image will be 1/3 the height and 1/3 the width of the original. The reduction is accomplished by eliminating two out of every three pixels in both the horizontal and vertical directions.

In producing an image enlarged by a factor of 3, $S = 3$ is selected. Since the output image must be contained by the same frame size as the original, the final enlargement will be a section of the original, blown up to occupy the entire frame. The enlargement is done by replicating every pixel 3 times in both the horizontal and vertical directions.

Application—Image scaling is used in the geometric manipulation of images prior to or following various processing operations. In particular, dual image/pixel point processes often require that the two input images be in tight geometric alignment prior to their combination. Improper alignment can produce undesired fringe effects around object boundaries.

Enlarging a portion of an image can help in the visual analysis of detailed objects. Furthermore, an enlargement may allow visible recognition of an image's degradation, helping in the selection of corrective processing.

In graphic arts, image reduction facilitates the combination of various input images into a composite image.

Cautions—Image reduction is carried out by eliminating pixels, which compresses the spatial dimensions of the frame. As a result, spatial resolution is reduced. Once an image has been reduced, the stripped pixels are lost and may not be recovered except by retrieving the original image from some master storage source. In enlargement, pixels of a selected area are replicated to stretch the spatial dimensions of a portion of the image to the full frame size. Hence, no image data is lost from the enlarged region. The remainder of the original image, though, is entirely lost. Again, recovering the lost image data requires retrieving the original from a master source.

Implementation—Frame process This operation is generally carried out in software by a host computer. The basic flow chart is given, where

S = scaling factor
x = input image pixel address
y = input image line address
x' = output image pixel address
y' = output image line address
PRC = pixel replication counter
LRC = line replication counter

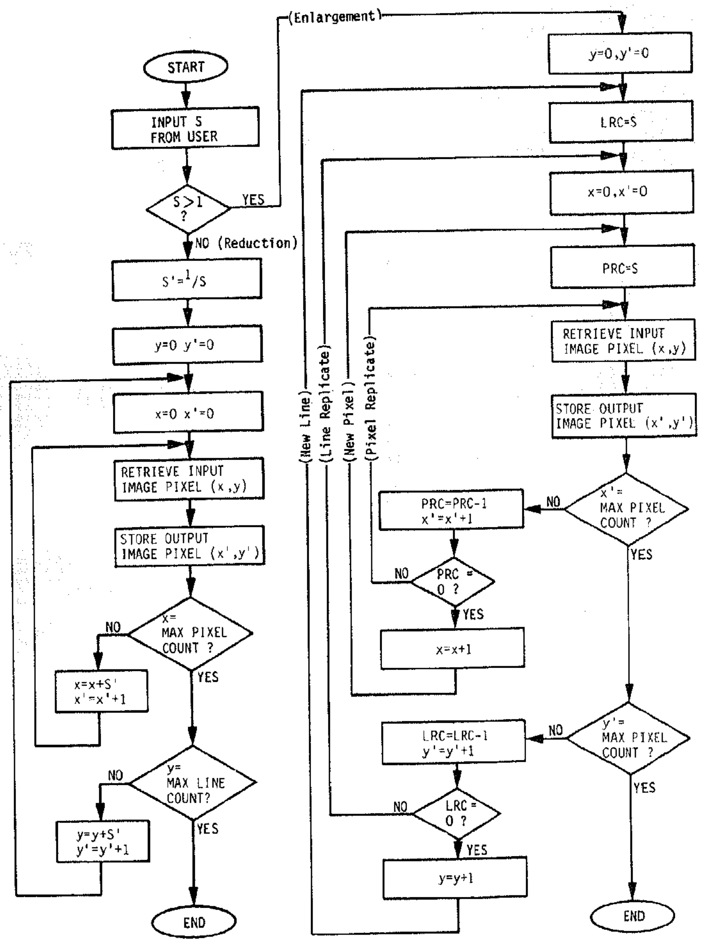
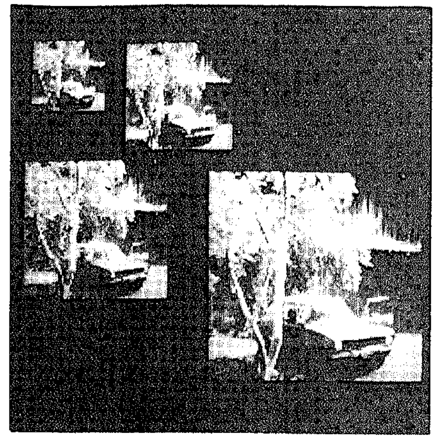


Figure IOS14-1 Image scaling flow diagram.

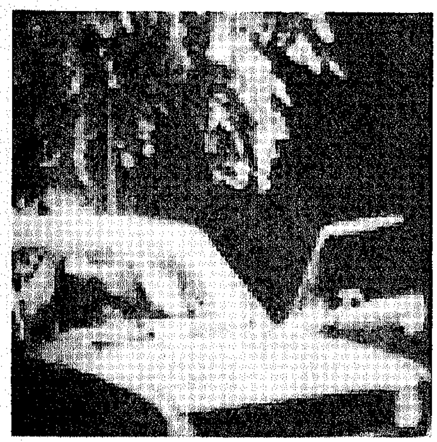
Results

Figure IOS14-2

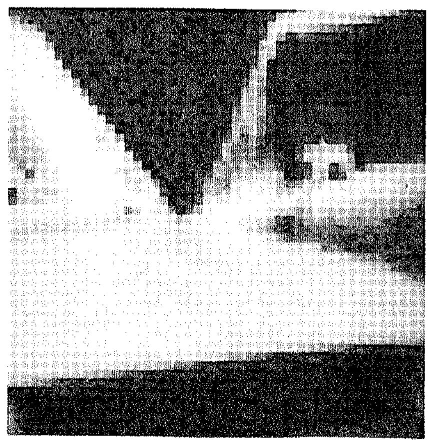
(a) Original image.



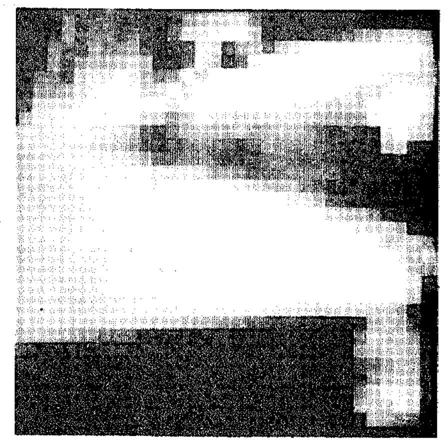
(b) Various spatial image size reductions.



(c) Times 2 enlargement.



(d) Times 4 enlargement.



(e) Times 6 enlargement.

Image Operation Study # 15

Operation—Image Rotation

Description—Image rotation allows spatial rotation of an input image. The angle of rotation, θ, is selected. The rotation algorithm produces an output image geometrically rotated counterclockwise through the angle θ. The rotation occurs about the image center point. The pixel coordinate origin is defined as (0,0) at the center of the image. Hence, for a 256 × 256 image, pixel coordinates range from −127 to 128, left to right, and the line coordinates range from −127 to 128, top to bottom.

Application—Image rotation is used in the geometric manipulation of images prior to or following various processing operations. Often dual image/pixel point processes use rotations to bring the two input images into spatial registration prior to operating upon them.

In graphic arts, image rotation facilitates the combination of various input images into a composite output image.

Cautions—In rotating an image composed of fixed discrete pixels, the input pixels will typically not transform directly into output pixel locations (except when θ is a multiple of 90°). Because of this, interpolation algorithms are often

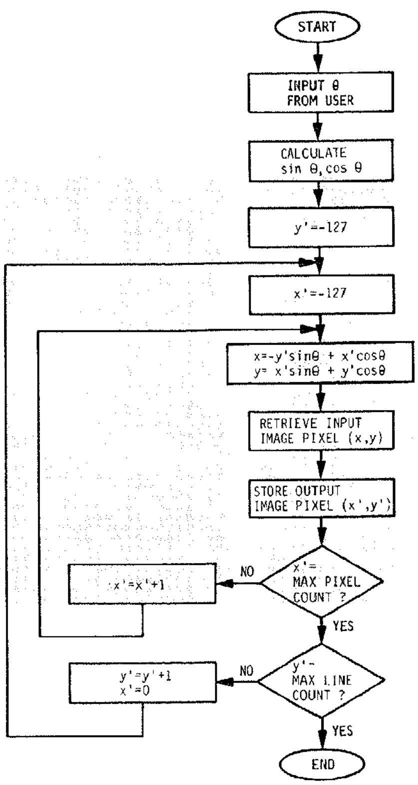


Figure IOS15-1 Image rotation flow diagram (rotation about image frame center point).

employed. These algorithms break a transformed input pixel's brightness into four parts, each contributing to one of the four output pixels surrounding the calculated output image pixel location. Interpolation schemes take on a variety of forms—some rough approximations, some more exact. The tradeoff is often related to required processing time. Depending on the amount of detail present in an image along with the final requirements of the image, an appropriate interpolation scheme may be employed.

Implementation—Frame process This operation is generally carried out in software by a host computer. The basic flow chart is given, where

θ = angle of counterclockwise rotation
x = input image pixel address
y = input image line address
x' = output image pixel address
y' = output image line address

It should be noted that this algorithm uses reverse pixel mapping. Input pixel coordinates are calculated for each and every possible output coordinate. Due to the non-one-to-one pixel transformation of the rotation operation, this scheme insures that all output pixels will acquire a new rotated value.

Results

Figure IOS15-2
(a) Original image.
(b) 330 ° rotation.
(c) 60 ° rotation.

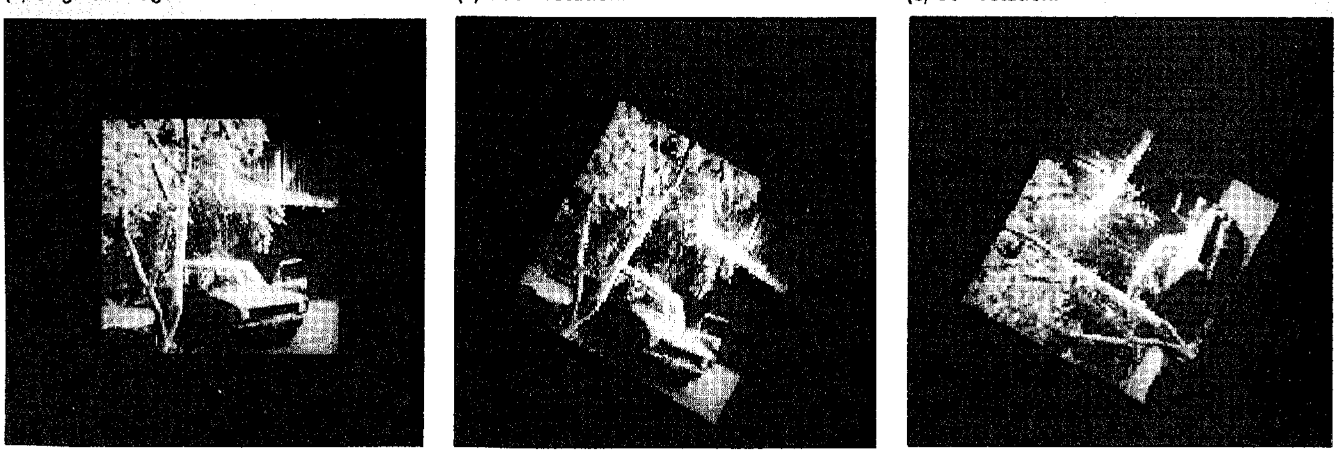


Image Operation Study # 16

Operation—Image Translation

Description—Image translation allows the side to side, up and down spatial movement of an image. The translation offset values, T_x and T_y are specified. T_x is the number of pixels, positive or negative, that the image is to be shifted in the horizontal direction. T_y represents the number of pixels that the image is to be shifted vertically. The output image is identical to the original except spatially shifted horizontally and vertically.

Application—Image translation is used in the geometric manipulation of images prior to or following various processing operations. Often dual image/pixel point processes use translations to bring the two input images into spatial registration prior to operating upon them.

In graphic arts, image translation facilitates the combination of various input images into a composite output image.

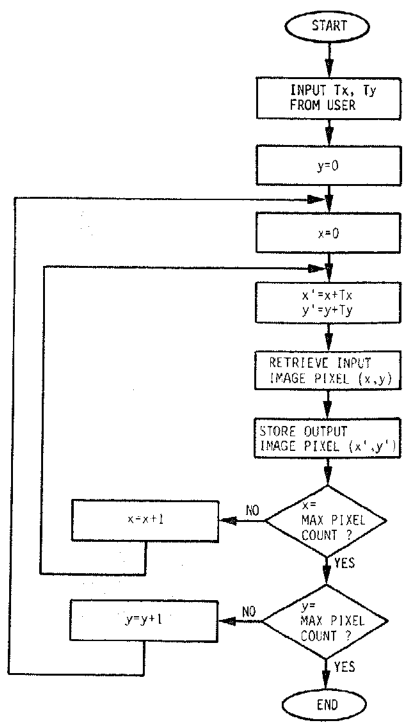


Figure IOS16-1 Image translation flow diagram.

Cautions—Loss of image data cannot result from a translation operation. Wraparound effects are prevalent, though. Normally, a reduced image is translated from one area of the frame to another. When the image is full size, or large with respect to the amount of translation, often a side will translate off the image frame. The pixels that go off one side come up on the adjacent side of the frame. The image may be thought of as on a special loop that may roll up and down or side to side. The image will roll off one side and wrap around to the other.

Implementation—Frame process This operation is generally carried out in software by a host computer. The basic flow chart is given, where

T_x = horizontal translation
T_y = vertical translation
x = input image pixel address
y = input image line address
x' = output image pixel address
y' = output image line address

Results

Figure IOS16-2
(a) Original image.
(b) Various spatial image translations.
(c) Wraparound effect on a large format image.

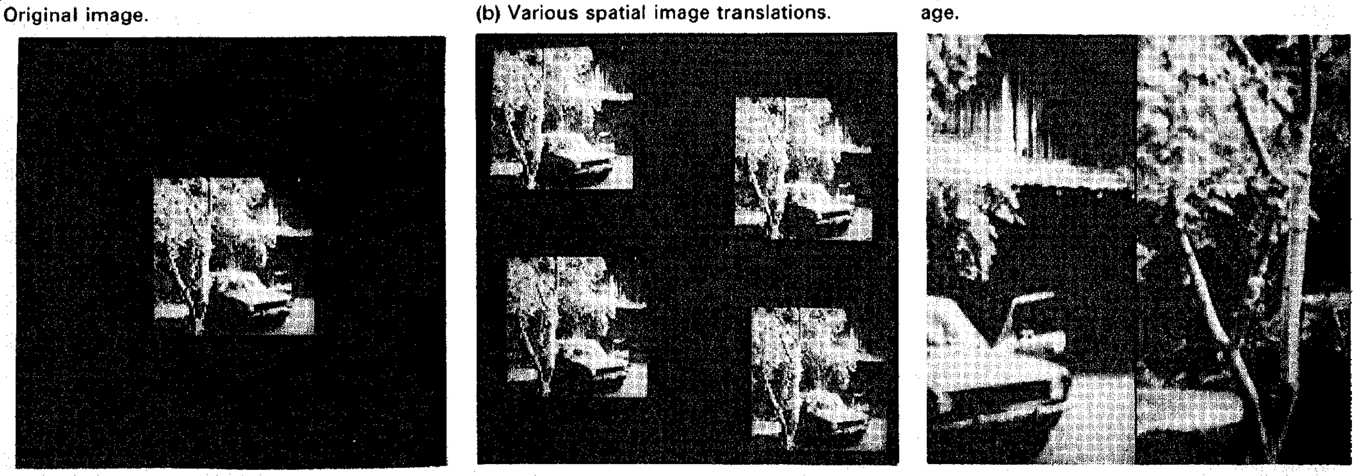


Image Operation Study # 17

Operation—Image Averaging

Description—The image averaging operation produces an output image that is the average of two input images. Images are combined on a pixel-by-pixel basis. Each pixel brightness in the first image is averaged, with its corresponding pixel brightness in the second image. When applying this operation to two completely different image scenes, the result is a composite image of both. If the scenes are identical, the resultant image will be similar, with a decrease in any present random noise.

Application—The predominant use of this operation is in image noise reduction. Random "snow" noise present in an image changes spatial position from frame to frame. Providing that more than one frame of the same scene is available, reduction of this type of noise is possible. Averaging two or more of the images diminishes the presence of the "snow" because the noise becomes averaged with good pixel data. If enough frames are averaged together, the net result will be an image visually devoid of the noise.

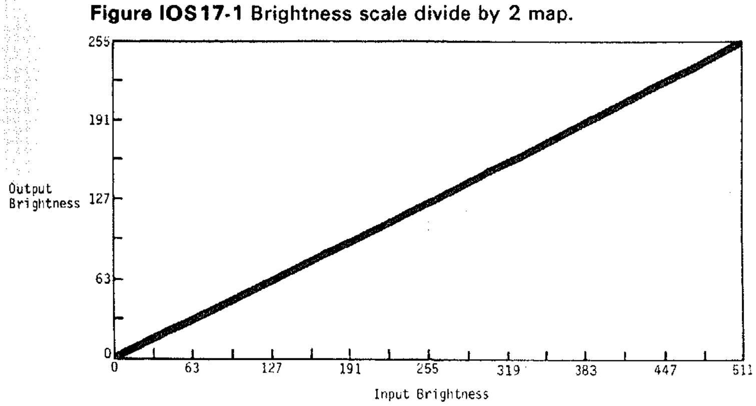
Averaging two images of totally different scenes produces a result where both scenes appear superimposed upon one another. This effect is found useful in some applications.

Cautions—This operation may generally be implemented without regard to downfalls.

Implementation—Dual image/pixel point process The image averaging operation uses a dual image/pixel point process to carry out the pixel-by-pixel addition. In the addition of two 8-bit pixels, a 9-bit result is produced. This result is then brightness scaled by a factor of two, to a valid 8-bit pixel value, by the dual image/pixel point processor output map, *M*. The equation for this operation is given.

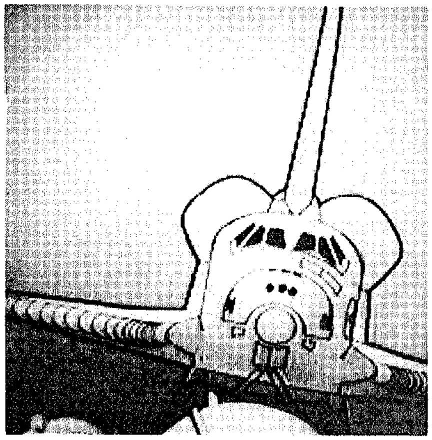
$$O(x,y) = M[I_1(x,y) + I_2(x,y)]$$

Figure IOS17-1 Brightness scale divide by 2 map.

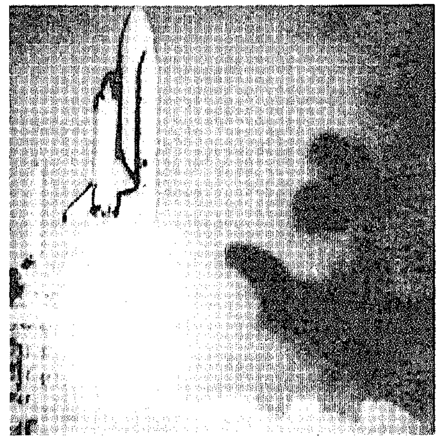


Results

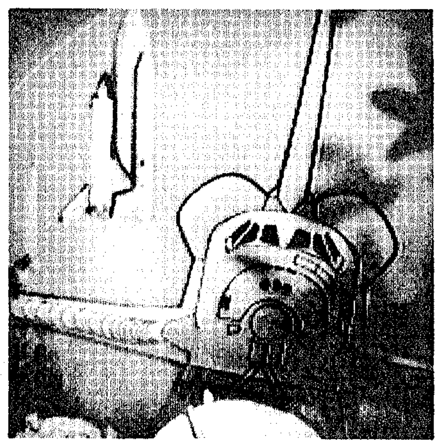
Figure IOS17-2



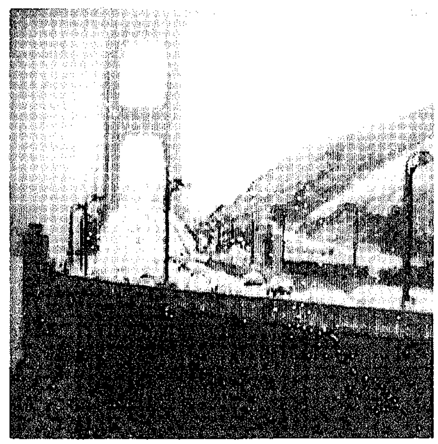
(a) Original image #1.



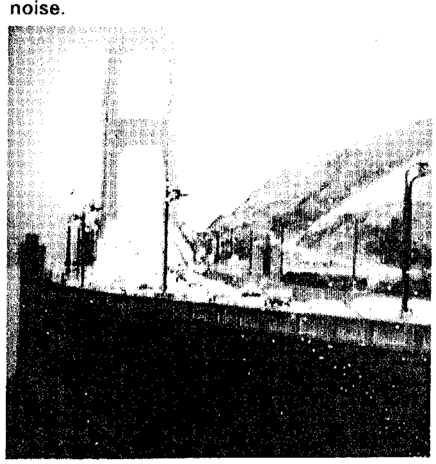
(b) Original image #2.



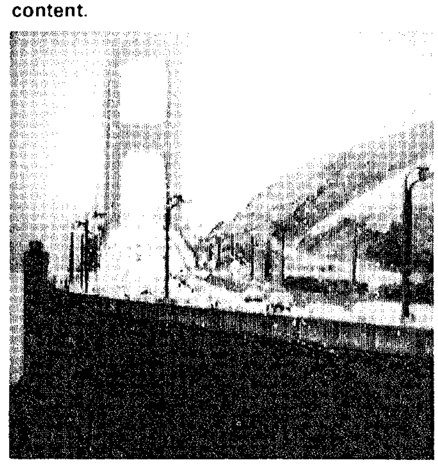
(c) Average of #1 and #2.



(d) Original image corrupted by noise.



(e) Second original image corrupted by noise.



(f) Averaged image pair with reduced noise content.

Image Operation Study # 18

Operation—Image Subtraction: Motion Detection, Background Subtraction

Description—Image subtraction is accomplished by taking the pixel-by-pixel difference of two input images. This process is normally applied to similar images of basically the same scene. The resulting output image is predominantly black where elements of the scene are identical in the two images. Objects that are not present in one image or have changed spatial locations show up upon the black background.

Application—Two predominant uses for image subtraction are motion detection and background subtraction operations. In motion detection, two images of the same scene are used. Identical portions of the images will subtract out to black in the output image. Differences in the two images, such as objects that have moved, will show up clearly in both locations of the output image. Hence, motion and direction are made evident. If the time between image acquisition is known, then speed of the moved object may also be calculated. One particular application of motion detection is in medical blood flow analysis.

Background subtraction is an identical operation to motion detection; however, it is thought of in different terms. Complicated scenes with a lot of detail may make it difficult to detect subtle changes from frame to frame. Here, the idea is to subtract out common background image information, leaving only differences for analysis. Subtraction of two images, one of only the background, allows the differences to be made immediately detectable. Medical X-ray imagery often uses this type of operation. Complicated surrounds in an image are eliminated by taking an image prior to the patient's ingestion of a radio-opaque liquid. A second image is then taken once the areas of interest have been highlighted by the liquid. When the first image is subtracted from the second, only the highlighted features are left. The complexity of analysis is greatly reduced when working with an image enhanced through this process.

Cautions—This operation may generally be implemented without regard to downfalls.

Implementation—Dual image/pixel point process Image subtraction uses a dual image/pixel point process to carry out the pixel-by-pixel subtraction. In the subtraction of the two 8-bit pixels, a 9-bit result is produced. The ninth bit represents a borrow, or arithmetic underflow. To faithfully reproduce all differences in pixel brightness, an absolute value map is used to insure that all results are positive, and 8-bit. This function is handled by the dual image/pixel point processor output map, M. The equation for this operation is given.

$$O(x,y) = M[I_1(x,y) - I_2(x,y)]$$

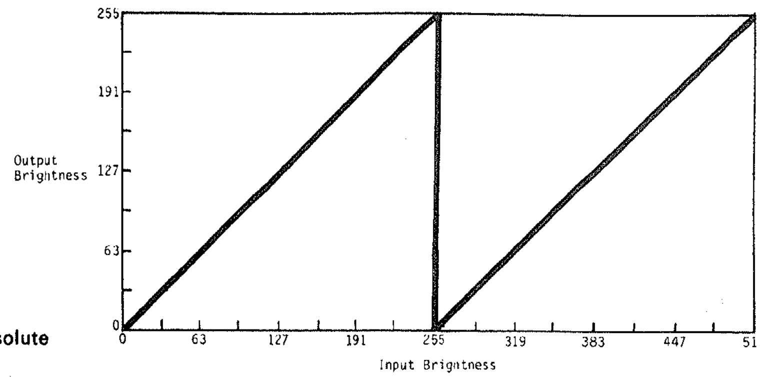


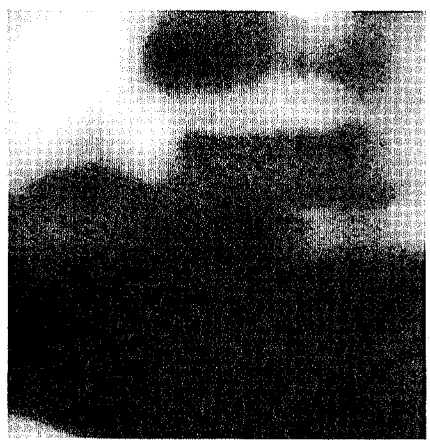
Figure IOS18-1 Brightness scale absolute value map.

Results

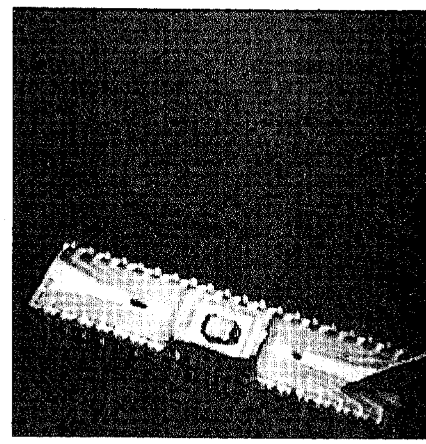
Figure IOS18-2

(a) Original image with foreground and background.

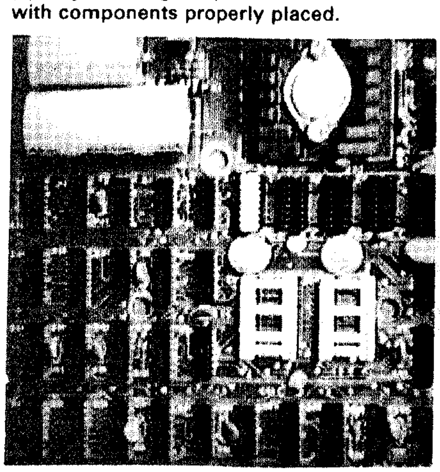
(b) Image of only background.



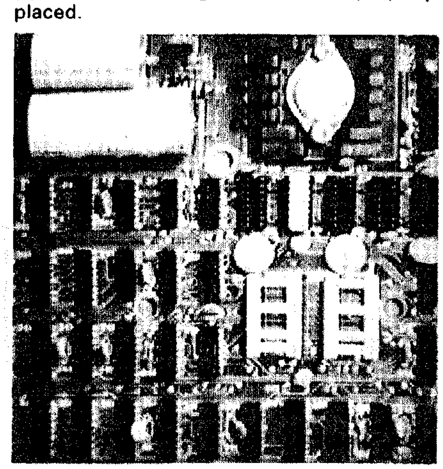
(c) Image subtraction yielding only foreground object.



(d) Original image of printed circuit board with components properly placed.



(e) Board with intergrated circuit improperly placed.



(f) Subtraction yielding only placement discrepancy with background detail removed.

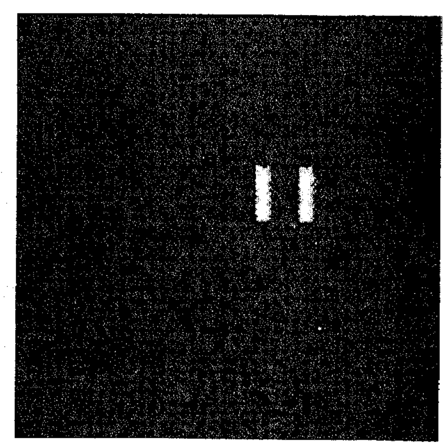


Image Operation Study # 19

Operation—Logical Image Combination: AND, OR, EXclusive-OR

Description—This operation provides for the logical pixel-by-pixel combination of two input images. Such operations between images include AND, OR and EXclusive-OR functions. An output pixel value is calculated by logically combining the two respective input pixels in a bit-by-bit fashion.

Application—The AND and OR image combinations are normally used to mask and add together images. We use the AND function to mask off portions of an image. Given an image of which we wish to retain only a small section, a second image may be generated to be a mask. The mask image is composed of pixels having a binary value equal to all 0s (black), where the original image should be masked, or 1s (white) where it should be allowed to appear in the output image. The AND combination of the two images produces the final masked image.

Image OR combinations are used to add together subimages into a composite output image. Given that two subimages do not spatially overlap and are masked, they may be ORed together combining both into a single output image.

The EXclusive-OR combination may sometimes be found useful as a simple image comparing operation. Pixels are combined bit by bit, producing an output image displaying black where the two input pixels are *exactly* identical. Where the two pixels are not perfectly identical, the output pixel will be something other than black, depending on the actual bit-for-bit comparison.

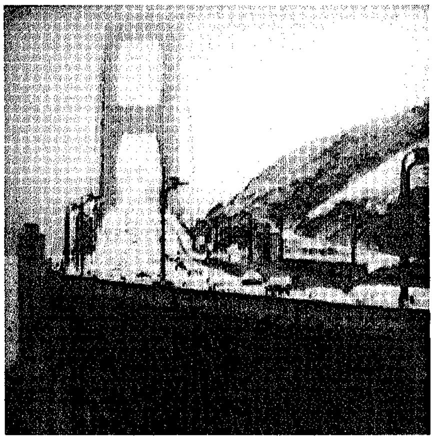
Cautions—Performing the AND or OR combinations on images that have pixels other than black or white in overlapping regions often will produce strange results. This is because the operations happen on a bit-by-bit fashion. The result of an AND or OR of two pixel brightnesses does not really hold any practical significance. In the case of EXclusive-ORing two images, if two respective input pixels are not exactly identical, the resulting value means very little. The significance is found in resulting pixels that are totally black, for these pixels are identical in the two input images.

Implementation—Dual image/pixel point process All logical image combinations are carried out by the image/pixel point processor.

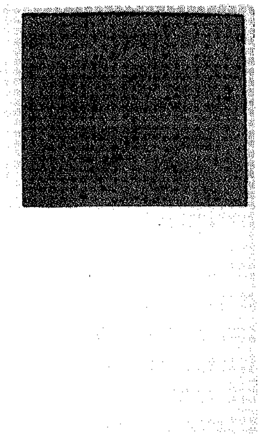
Results

Figure IOS19-1

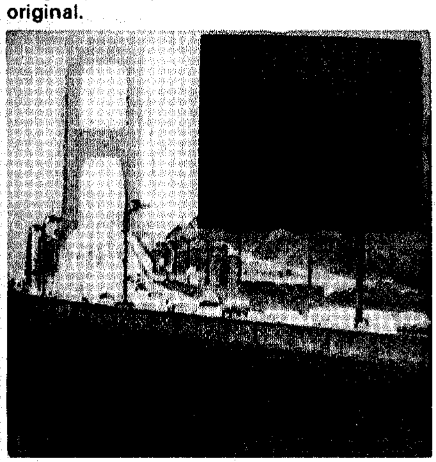
(a) Original image.



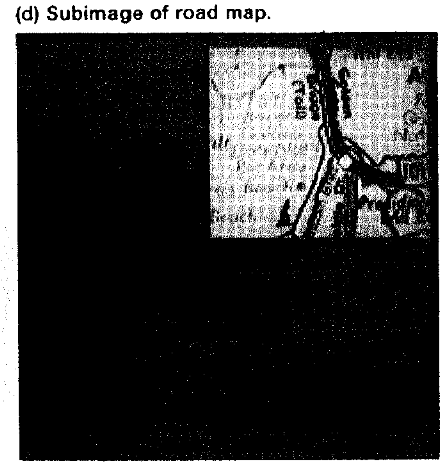
(b) Mask image.



(c) AND combination of original and mask images yielding a masked version of the original.

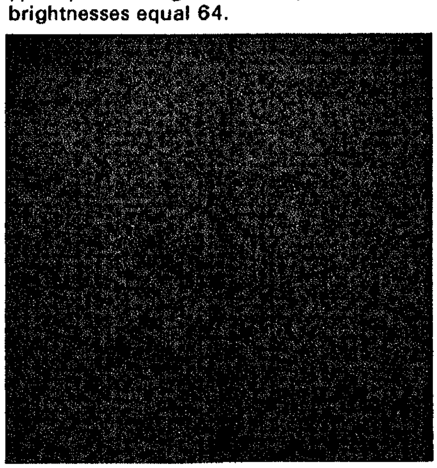


(d) Subimage of road map.



(e) OR combination of subimage and masked original image.

(f) Gray level image where all pixel brightnesses equal 64.



(g) EXclusive-OR combination of original with gray level images, yielding image with white pixels wherever the original image pixels equal 64.