

Machine vision applications using binary and grayscale morphology

GERALD W. BUDD

MACHINE VISION TECHNOLOGY has been available for industrial use for over 15 years. The image processing computers of today offer the engineer a variety of products from which to choose. The selection of image processing equipment ranges from board level products that require a host computer to self-contained dedicated systems. Most advancements in image processing hardware usually come about due to advancements or additional requirements dictated by image processing software. In many applications, the software drives the application of machine vision technology.

Early image processing applications were designed to perform fairly simple tasks. Line scan sensors are designed to operate on line or edge information. Advanced line scan systems can apply image processing code (algorithms) to determine if logical connections exist between information obtained from multiple scans, thus connecting or joining lines to form objects. The first-generation machine vision applications used television cameras to acquire images. The images were often compared using a technique referred to as "template matching." The template matching approach compares a "reference" or "golden" image to the sample image. Typical applications performed these operations using "binary" images. A binary image is one that is composed of light and dark pixel locations that can be stored as ones and zeros within a computer memory. Image noise and object movement often limit the use of this technique to systems that uti-

lize well-controlled material handling, i.e., electronic components, circuit boards, web inspection, and printing applications. It is becoming more difficult to implement a simple template-matching technique with the desire to isolate small imperfections. Often the image processor will implement some type of "edge detection" algorithm to reduce the amount of data that must be processed in an image. Even with the use of an edge detection algorithm the template matching approach can encounter subtle variations between a "golden" image and the sample image.

It is common to employ some type of "filter" on information within a limited area where variations are expected to occur between the "golden" and sample images. The region in which the variations occur is often referred to as the "boundary region." The use of a filter in the boundary region is an effort to reduce or eliminate decisions based on pixel information that is considered unstable. The template-matching approach requires little image processing power and can operate near frame rates or, in some specialized systems, at field rates. This approach also suffers from the fact that it does not allow for rotation of the image with respect to the "golden" image.

SRI CONNECTIVITY ALGORITHM
The next logical step in the analysis of image information was to classify or identify individual segments of an image. In 1973 an algorithm was developed by Rosen *et al.* (1) at the Stanford Research Institute under a grant from the National Science

ABSTRACT

This paper gives an introduction to binary morphology for recognition of objects by shape, applies grayscale morphology to improve signal to noise ratio, and uses array sensors and parallel processors for high-speed inspections.

Foundation. This algorithm became known as the SRI connectivity algorithm. It became the basis for many of today's commercial machine vision systems. The SRI algorithm provides an image analysis technique that operates on binary images to determine connected regions of light or dark pixels. These light and dark regions are often referred to as "blobs." From the connectivity information, the SRI algorithm is able to determine geometric features of each blob in the image space. The SRI algorithm is able to extract a considerable amount of "high level" information about the blob(s) in the image space. Typical information that can be extracted from the blob feature set may include: (a) area, (b) center of mass, (c) perimeter, (d) major axis, (e) minor axis, (f) bounding box, (g) number of holes, etc.

All implementations of the SRI connectivity algorithm must extract "useful" information from binary blob information. This approach assumes that one can obtain a good binary image on the information that one is seeking. The basic algorithms work well on high contrast images but produce limited success with low contrast images. The operational speed of a system based on the SRI connectivity algorithms are directly proportional to the number of blobs within the image space. In general, an image space that contains 10 blobs will take 10 times as long to process the information as an image with a single blob.

The SRI connectivity algorithm was written as a general purpose image analysis tool that could be used on a number of machine vision platforms. The algorithm was written and implemented in a machine independent "C" language. A good description of a portable implementation of the algorithm was presented by Milligan (2).

MATHEMATICAL MORPHOLOGY

The dictionary defines *morphology* as the study of form, structure and shape excluding an object's function. Mathematical morphology as applied to image processing is a technique that utilizes form, structures or shapes as the fundamental operators on images. There are many good references on the subject of mathematical morphology (3-5) and morphological filters (6). The forms, structures and shapes are often referred to as "structuring elements." The structuring elements are used by mathematical morphology to operate on the data in an image. The structuring elements are used as operands to perform morphological operations such as erosion, dilation, opening, closing, and many others.

The use of mathematical morphology (hereafter referred to simply as morphology) is very effective when one is creating machine vision algorithms and is able to think in terms of shapes. Morphology is logical operations on images instead of arithmetic operations. Schmitt (7) states that a structuring element is a shape that the computer uses to probe the image, i.e., to test where the structuring element does and does not fit within the image. Using this technique, the computer can separate an image into data that are expected from an operation from data that are different from what is expected.

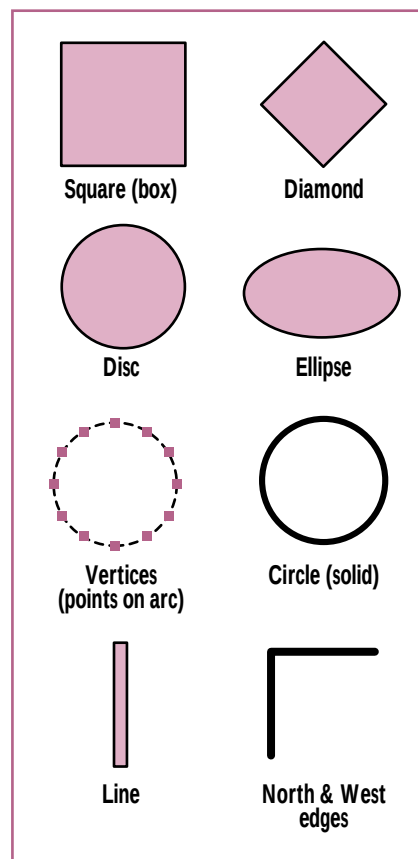
BINARY STRUCTURING ELEMENTS
A structuring element can be defined as any shape or even a group or discrete points in two-dimensional space (2 space). However, it is most useful to think of structuring elements in terms of geometric shapes. Symmetric structuring elements are less sensitive to the orientation of objects within the image shape. Figure 1 illustrates several common structuring elements.

Binary morphology

Dilation and erosion are the principal binary morphology operators. The dilation operation will "grow" a binary object using a structuring element. This operation can be thought of as placing a copy of the structuring element at every pixel position of the original binary object. It is common to define the origin somewhere within the boundaries of the structuring element. When using symmetric structuring elements, the origin is typically its center. The dilation operation is very useful for "knitting" together defects that are in close proximity to each other.

Figure 2 illustrates the principle of dilating several small components (blobs) into a single large component. The individual points are operated on with a disc-shaped structuring element. The points are dilated using the disc structuring element to form larger objects each with a diameter equivalent to that of the structuring element plus the individual point. The resultant is an object comprising an area at least the size of the area between the individual dots and often much larger.

If the individual points are farther apart than the diameter of the structuring element, the operations will yield a resultant in which the discs do not touch each other. If the diameter of the structuring element is exactly equal to the distance between points, the resultant will yield an object that is not completely

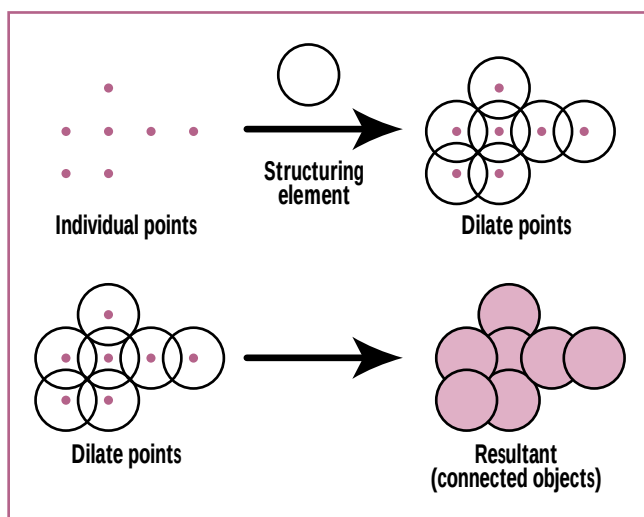


1. Binary structuring elements

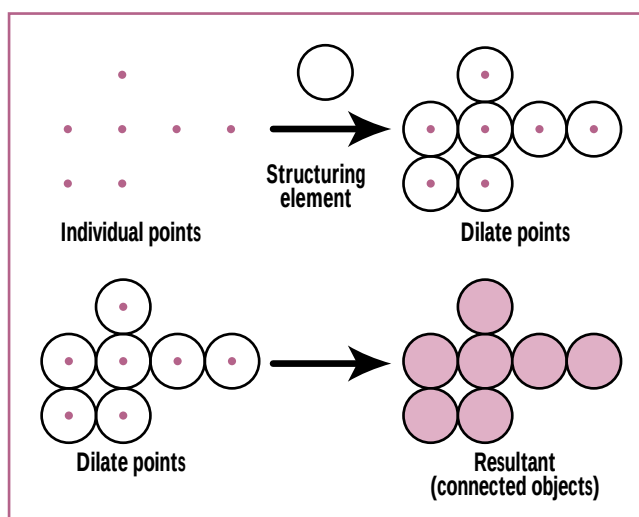
filled in, e.g., voids will be present between discs (Fig. 3).

The erosion operation is the opposite of the dilation operation. The erosion operation collapses (shrinks) the original binary object. It does this by placing an indicator at the origin of the structuring element at every location in which it "fits" in the original binary object. The erosion operation is a practical method of eliminating information based on size that is not desired. The erosion operation is a binary "filter" to remove objects that are smaller than the minimum acceptable size. If one were to apply the erosion operation as a subsequent step to the previous pair of examples, different results would be obtained.

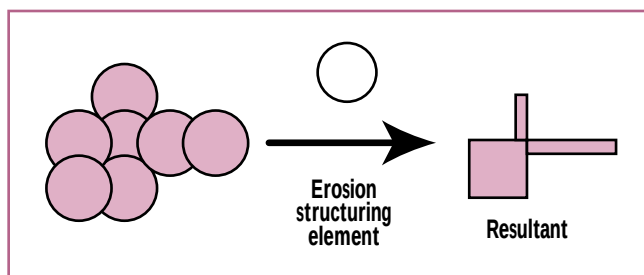
Figure 4 illustrates the result of the erosion operation when the original object has a considerable



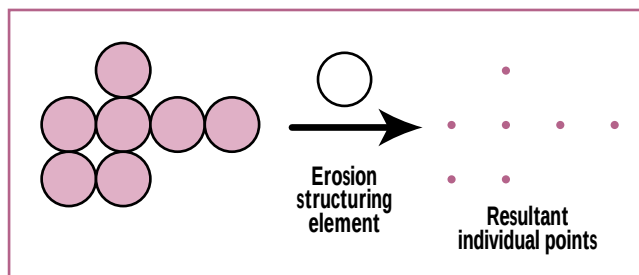
2. Dilation using structuring element with diameter equal to point spacing



3. Dilation using structuring element with diameter less than point spacing



4. Erosion of object with overlapping regions



5. Erosion of object without overlapping regions

amount of overlapping regions. The structuring element is able to fit inside the object in many locations producing a resultant with substantial area remaining. The operations performed on a set of individual points in close proximity using a combination of dilation/erosion will result in an object with an area equal to approximately the area bound by the points.

Figure 5 illustrates the erosion operation performed on a set of objects that have no overlap region. If the image is eroded using a structuring element of approximately the same diameter as the objects, the resultant will be a set of individual points, each positioned at the center of the discs. If the dilation/erosion combination were performed on the

set of points using a structuring element with a diameter less than the distance between the individual points, the result would yield the original points. This series of operations would identify regions in the image with scattered points farther apart than the size of the structuring element.

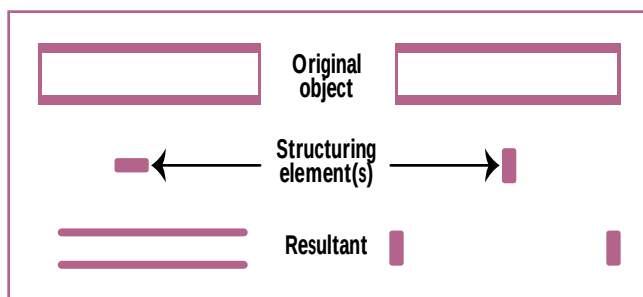
Shape identification using binary morphology

There is a special extension of the dilation and erosion operators known as conditional operators. A conditional operator performs an operation until a certain criterion is met. The conditional operators can perform the testing criteria on either ones or zeros in the image area. In addition, the direction of the operation may be specified; for example,

one may dilate in the north, south, east, west, or a combination of directions*. The careful selection of directional operators can be used to extract information that has a specific orientation. The converse is also true, the directional operators can be used to remove information that is common in all images.

Figure 6 illustrates how the selection of operators can extract or remove information from the image using structuring elements. Both objects, the rectangular outlines, are

*The nomenclature that Phoenix Imaging has adopted includes the acronym for the operator followed by the direction of operation. A dilation in the north, south, east, west direction is referred to as d_nsew. A conditional dilation in the north, south, east, west direction using positive binary values as the test criteria is referred to as cd1_nsew.



6. Filtering with binary structuring elements

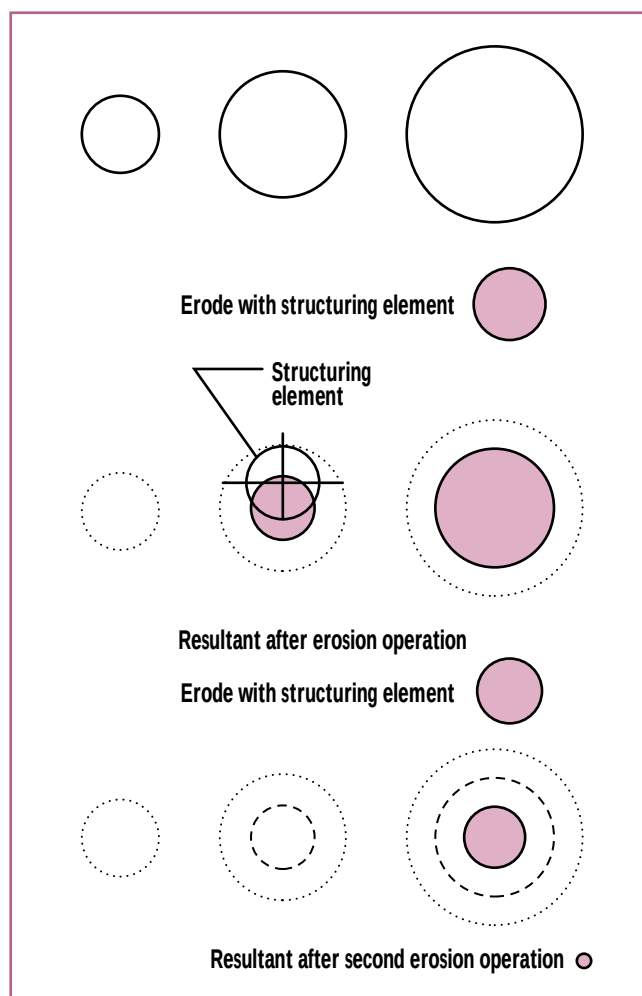
“operated on” by performing an erosion operation. The object on the left was eroded using a horizontal structuring element, and the object on the right was eroded using a vertical structuring element. Using a small horizontal structuring element will remove any information that has a vertical or nearly vertical orientation that has a smaller dimension than the structuring element. The small vertical structuring element will eliminate any information with a horizontal orientation that is smaller in size than the structuring element. The horizontal structuring element is created, in effect, by performing an erosion in the east-west direction. The vertical structuring element is created performing an erosion in the north-south direction. The simple linear structuring elements can be expanded to create two-dimensional shapes.

A disc-shaped structuring element is created by performing erosion or dilation operations in the correct proportions.

Figure 7 illustrates how one might apply binary morphology to the classification of objects. First we assume that the objects are all binary circular objects of three distinctive diameters. The following series of operations can be used to differentiate objects of various sizes. The objective is to isolate and identify each of the three different size discs.

The first step is to erode the objects with a structuring element that is about the diameter of the smallest object. The resultant of this operation is the center illustration. Two discs remain, each with a smaller diameter than the original objects. If you were to slide the structuring element around the interior of the original object and turn “on” a pixel at every location that origin of the structuring element could reach, the resultant would be the gray disc.

A second erosion operation similar to the first would isolate the larger disc. This operation could be repeated as many times as necessary to identify all the various size discs within an image space. Unlike the SRI connectivity algorithms, the morphological operators operate on all of the objects present in the image space simultaneously,



7. Filtering with disc structuring elements

regardless of the number of objects present or their position.

The sequence of operations is very important. The reversing of morphological operations does not necessarily result in the same information (image) that the process was started with. To illustrate this point, Fig. 8 shows a box (gray object). The box will be eroded using a circular (disc) structuring element. The structuring element moves along the interior of the box and generates an indicator at the center of the disc for every position in which the disc fits within the box.

The resultant of the first step is a box with similar proportions as the original box except that it is smaller in size. It is important to note that the box has square corners just as the original box.

The next operation is to dilate the smaller box with the same structuring element that was used in the reduction operation. In effect, this operation will generate a disc the same size as the structuring element positioned with its center at every point of the small box. The small box is represented in the dilation operation as the dot-

ted line. The resultant of this operation is a larger box with the same proportions as the original box. However, there is a major difference between the two boxes; the box created by dilation with a circular structuring element has rounded corners.

If we subtract the dilated box from the original box we are left with only the difference between them, namely the corners. In this way, binary morphology can be used for preparation of image data before it is passed to connectivity algorithms. This technique will increase the rate at which a system can process information because the data will be reduced to only include what is necessary.

The information could have been reduced even farther by employing a *conditional erosion* operation referred to as a reduction. A reduction will erode an object until it reaches a width of one pixel. Once it reaches a width of one pixel, it will not erode the object farther because the object will no longer exist. Each additional erosion operation after the object has reached a width of one pixel will eliminate one pixel from each end of the object until it reaches its limit, an object of one pixel in length. The end result of a large number of reduction operations on the four corners of the previous illustration would be a single pixel positioned at the center of mass of each corner.

There are many binary morphological image processing tools that can be applied in combination to produce the desired result. As previously mentioned, it is helpful to think in terms of geometric shapes. It is also very useful if the person developing the machine vision algorithm is familiar with Boolean logic operators *such as, and, or, not, not and,* etc. The algorithm development proceeds in logical steps of isolating one object or area of interest at a time. When the entire algorithm has been developed, it may be optimized by combining operations and similar procedures.

GRAYSCALE MORPHOLOGY

All of the techniques applied in binary morphology can be expanded into grayscale morphology. Binary morphology can be thought of as operations on two-dimensional objects. Grayscale morphology can be thought of as operations on three-dimensional objects. Think of a grayscale image as being a representation of 3 space, the X,Y dimensions corresponding to the dimensions of the video monitor and the Z dimension corresponding to the brightness of the image. It can visualize the grayscale objects by rotating the screen of a video monitor so that it were lying coplanar to a desktop. The dark grayscale values (near zero) would be at the level of the desktop and bright grayscale values (approaching 255) would be peaks extending above the desktop, a grayscale topology if you will.

The binary structuring elements could add the height dimension to create grayscale (three-dimensional) structuring elements. Figure 9 is an illustration of the corresponding grayscale structuring elements of the binary shapes. The box becomes a brick, the diamond becomes a pyramid, and the disc becomes either a cylinder (flat top) or a sphere (round top). Special structuring elements may be created by combining attributes of several structuring elements. For example, one could create a rod with a round top by combining a cylinder with a hemisphere. The greater the complexity of the structuring element, the more image processing time required to use it as a filter.

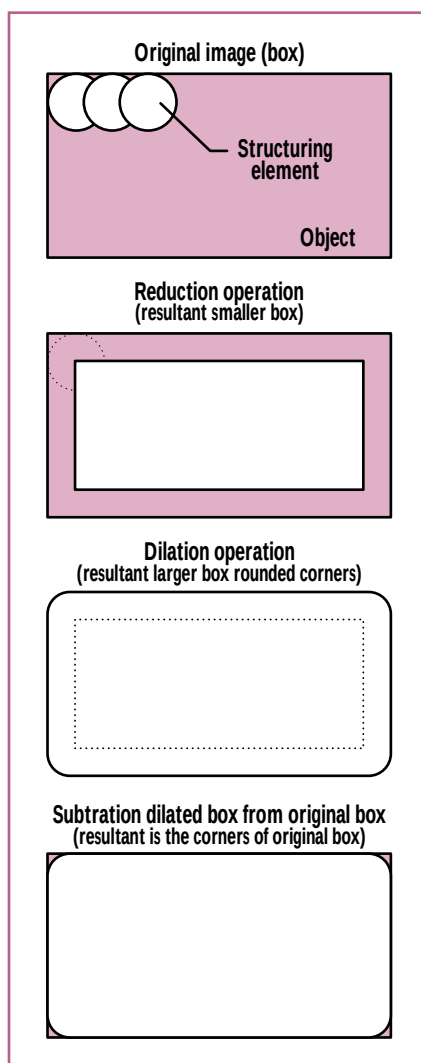
Most image processing systems have the capability to differentiate 256 levels of grayscale ($2^8 = 256$). A good quality CCD video camera with 50–52 dB signal to noise may be able to distinguish 180–200 levels of grayscale. For most surface inspection applications, the difference in grayscale between the defect and the surrounding area should be at least 5–10 gray levels.

Figure 10 shows a hypothetical situation of a sensor viewing a surface area with non-uniform illumination. This is an example of the type of grayscale response that might be obtained from a curved surface. The grayscale values are exaggerated for the purpose of this discussion. The light boxes represent high grayscale levels (bright objects in the image area), and the dark boxes represent low grayscale values (dark objects in the image area).

If one were to apply a global threshold in an attempt to isolate dark defects in the described image area, the results would most likely be inconsistent. Figure 11 shows two defects; each is four pixels square and has a grayscale value of 56. The first defect is positioned in the lower right section of the image and is easy to locate. The second defect is positioned in the upper left on a diagonal from the first. The second defect is approximately 24 gray levels darker than its surrounding area. This is a substantial difference and is easily detectable.

The problem with a global threshold approach under these conditions is that a good possibility exists for additional information to be extracted which may not be desirable. As we select lower and lower threshold values, the defects will be isolated. However, during this procedure, other regions in the image area with the same grayscale values as the defects will also be isolated. The second illustration in Fig. 11 shows the result of selecting a value of 56 and below as the threshold. The defects are present in the information but so is a large section of the image.

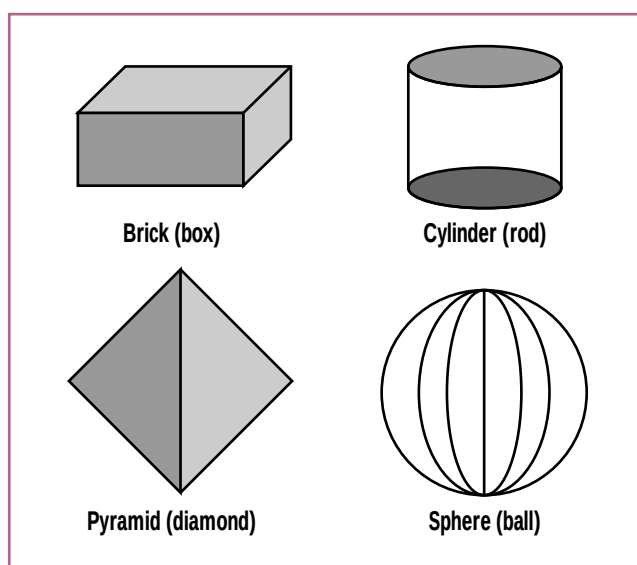
When the differential between the defect gray level and the background gray level is large (>20 grayscale values) the global threshold will generate acceptable results. Uniform illumination is very difficult to obtain on flat surfaces and nearly impossible to achieve of curved surfaces. When attempting to perform an inspection using a simple thresh-



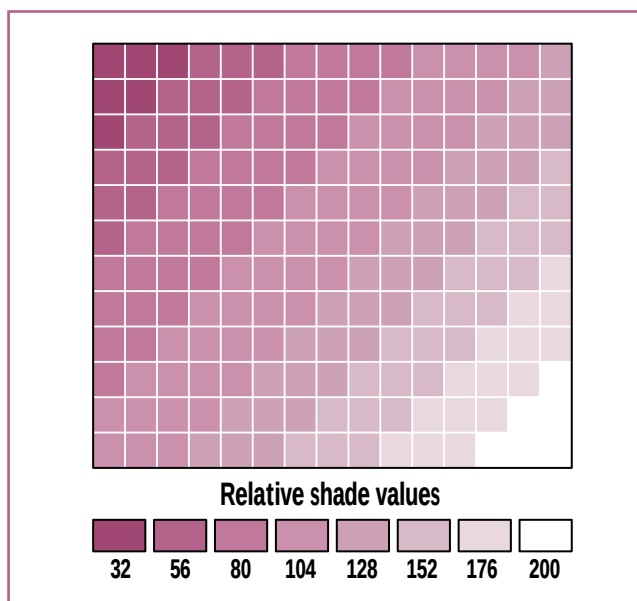
8. Data reduction with structuring elements

old, the generation of a uniform illumination field is critical. This technique does have an advantage in processing speed because it is the most basic image processing function. Typically, most surface inspection applications exhibit small changes in the grayscale level between the defects and the surface that they are found.

The uniform illumination of a curved surface is very difficult, such as the interior walls of the can. Therefore, we have selected to use an image processing system that has the ability to background normalize.



9. Representative grayscale structuring elements



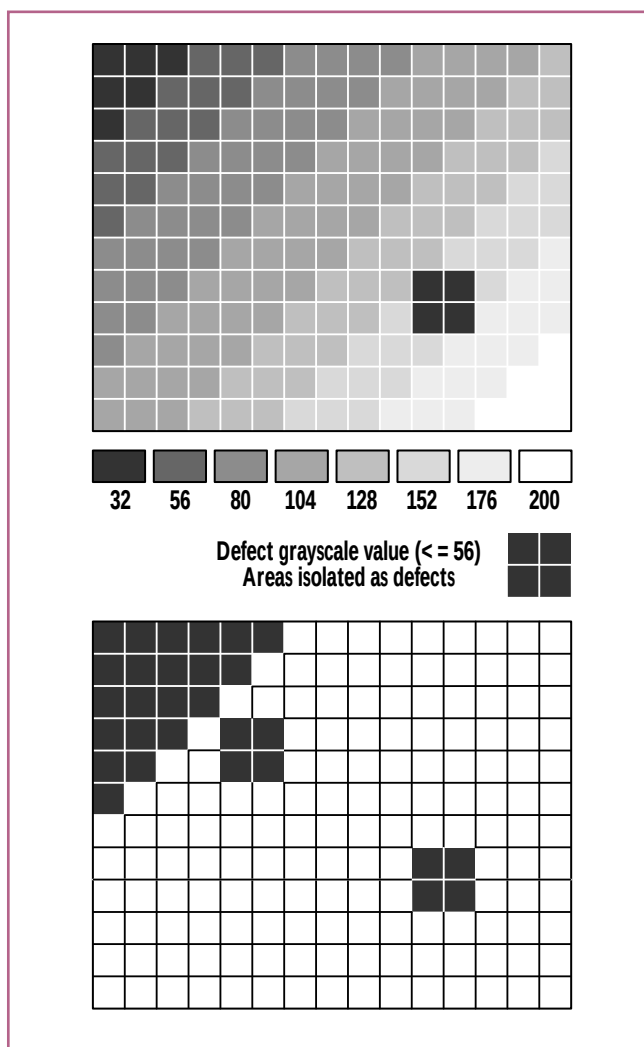
10. Sensor view of good surface with no defects (reference image)

This operation allows the system to determine a "normal" value of the grayscale information within a localized region. This normal value and the original grayscale value of each pixel location are compared to determine the relative contrast of features in the image.

The normalized grayscale value is calculated for every pixel in the image using the following algorithm. For the purpose of this discussion, we assume that a defect (foreground

information) is represented by a dark spot in the image and that the base material (background information) is light (higher grayscale values).

Figure 12 on this page shows the sequence of operations in the background normalization procedure. The diagram on the top represents a portion of a grayscale image with a defect (four dark squares) within the gradient of a lighter background. The algorithm will "cover over" the dark defects with the



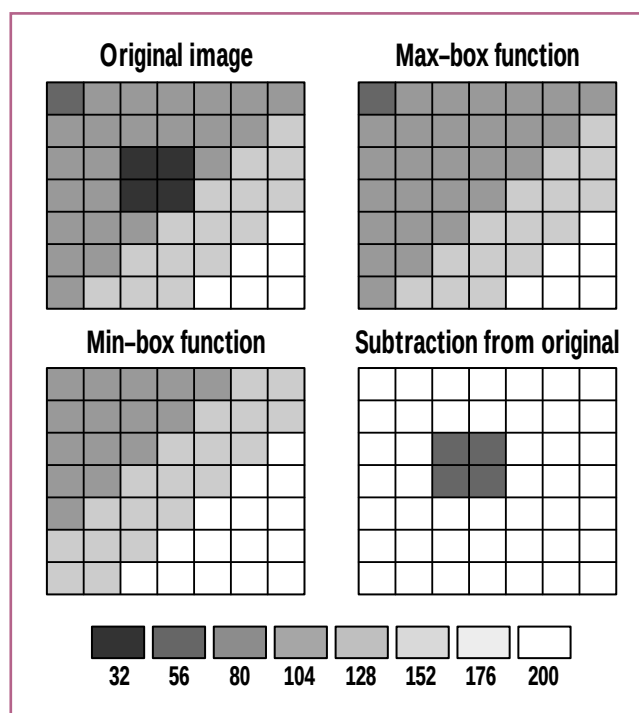
11. Isolation of defect using global threshold

lighter grayscale values of the surrounding pixels. The size of the operator used for this procedure is selected to be as large as or larger than the size of the largest defect that the system is to isolate.

The first operator "Max_box" will find the maximum grayscale value in a square neighborhood and place the value in the upper-right pixel of the four pixels that are closest to the geometric center of the square. The result of the first step is to cover the defect with the grayscale value of its neighbors.

The next operator is the "Min_box." This operation is the inverse of the Max_Box. This step is very important because it returns the background information to its original grayscale value. At the conclusion of these two steps, objects smaller than the size of the "Max-Min" operator are removed from the image.

The last step in the sequence is to subtract the processed image from the original image. The resultant is



12. Background normalization procedure

a "difference" image between the two that will yield a grayscale representation of the defect. The grayscale value of the defect will be equal to the grayscale difference between the defect and background. This technique requires a considerable amount of processing power and a special parallel processing architecture to make the approach practical. The parallel image processor used for this type of inspection is of the single instruction multiple data lines (SIMD) (8). The system can process over 2400 million binary operations per second (MOPS). This translates into an operating rate of 9322 binary frames per second.

Application of this technology allows the inspection system to isolate subtle defects even when the illumination conditions are not ideal. The diagrams on this page illustrate the results that can be obtained when the technology is properly applied. The top diagram represents the original defect image used in the global threshold example.

After the background normalization and subtraction operations, the resultant is obtained.

The importance of the resultant is that it represents the relative contrast of the defect with respect to its surrounding area.

The clear areas (pixels) represent locations where the background normalized pixels exactly match the grayscale value of the original image. The areas in which grayscale differences do exist are isolated and given a

KEYWORDS

Algorithms, image analysis, mathematics, pattern recognition, shape, structures.

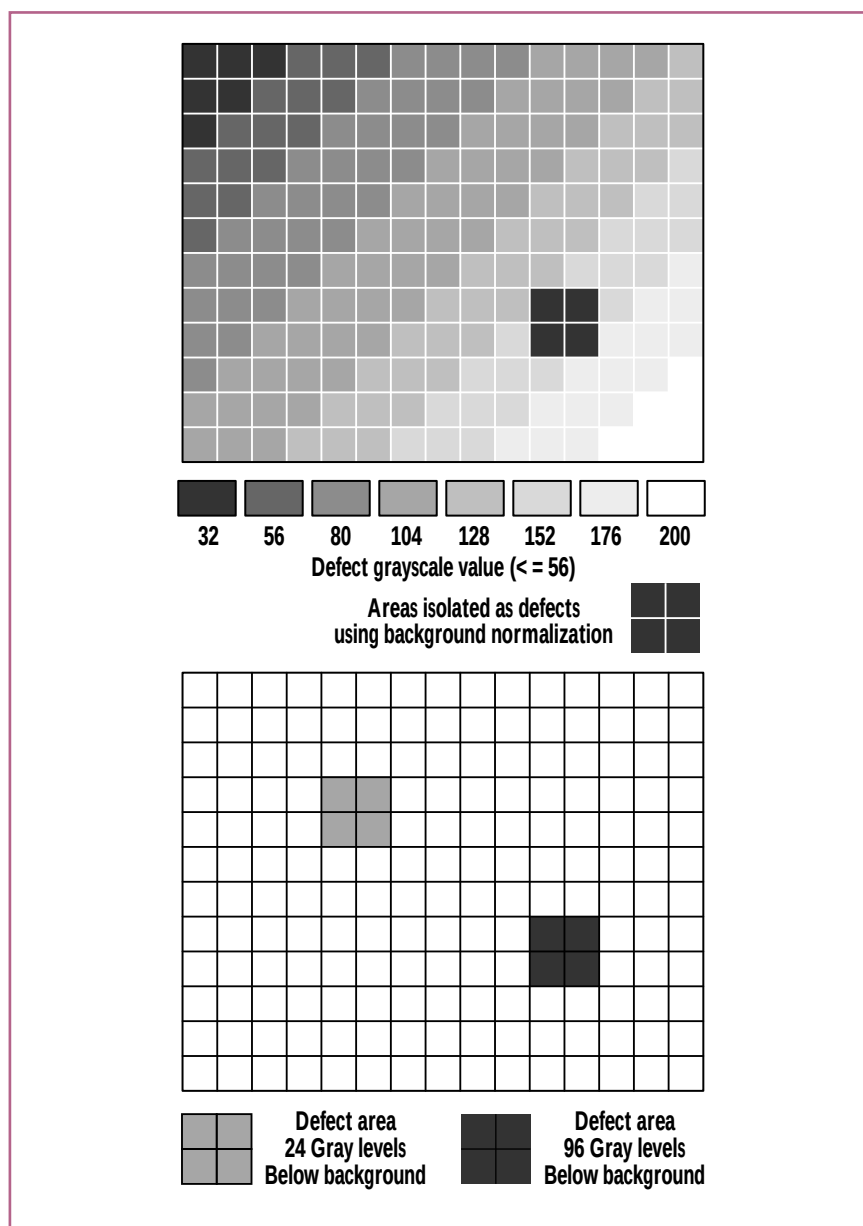
grayscale value equal to the difference. The isolated areas are usually considered as "suspect" regions until they meet farther criteria. Additional criteria include the grayscale level of the resultant, the suspect size, shape, and proximity to other suspect regions. The shape can be either binary (two-dimensional) or grayscale (three-dimensional).

The technique described above will isolate the surface imperfections that appear to be the most difficult task for the image processor. The more obvious defects can be isolated using binary morphology, i.e., thresholding techniques in combination with binary shape identification to isolate actual defects from image noise. The lower contrast defects can be isolated by further analysis of the resulting "normalized" grayscale image. The application of this technology for the isolation of surface imperfections on metal components has been described previously in the literature (9).

BINARY MORPHOLOGY APPLICATION

Determination of solid wood volume

Solid wood volume can be determined by approximation. A series of images must be acquired from the side of the flatbed trailers used to transport the logs. Depending on the size of the trailer (or trailers) as many as six sensors may be required per side. Images are acquired from both sides of the transport trailer. A typical image is used to analyze an area of approximately 10 ft wide by 8 ft high. Each sensor is calibrated with a reference target to provide a



13. Defect isolation using background normalization

correlation between pixel size and area in the sensor's field of view.

The grayscale image information is filtered to isolate the solid wood. The grayscale information is then converted into binary information that represents the shape of the individual stick end sections. The binary information is then used to calculate the total area of the "wood" based on the exposed end sections. This is a very good representation of the actual exposed surface area. The solid wood volume calculation is

based on the "average" surface area of wood from both sides of the trailer multiplied by the "average" stick length taken from the sensors located above the trailer.

The conversion technique used to translate grayscale information into binary information that is used to determine log diameter measurements are not perfect. However, the system can approximate the solid wood volume to better than 99% of the volume calculated by manual techniques. The system has the abil-

ity to isolate and ignore the bark region of the sticks and eliminate it from the solid wood volume calculation. In addition, the system has the ability to determine the distribution of stick diameters.

GRAYSCALE MORPHOLOGY APPLICATION

Surface inspection (texture)

The grayscale morphology procedures described above can be used for surface inspection. The technology provides several important functions that alleviate problems commonly encountered in machine vision applications. One of the most important problems to overcome in "real world" application is to provide uniform illumination. A machine vision engineer can spend countless hours on attempting to create uniform lighting conditions. The grayscale morphology approach is to create adequate illumination conditions and then allow grayscale morphology to "smooth" the image during processing. This type of image processing operation requires a vast amount of image processing power and is best suited for a massively parallel image processor.

The uniform illumination of a cylindrical object such as a log is very difficult to achieve. The desire to increase processing throughput by converting the grayscale information into binary information often produces poor results. The background normalization procedure described previously allows the inspection to compare relative contrast of features to their local neighborhoods rather than global light and dark regions. This technique allows the machine vision engineer to quickly isolate objects in the image area that are lighter or darker

than surrounding information. Common uses include isolation of scratches on the surface of shiny objects, surface roughness indication without the use of "perfect" illumination, and the isolation of bark on turned "clean" sticks. TJ

Budd is group manager, Phoenix Imaging, Farmington, MI 48335.

Received July 19, 1994.

Accepted Dec. 12, 1995.

Presented at the TAPPI 1994 Engineering Conference.

LITERATURE CITED

1. Rosen, C., Agin, G., Berger, J., et al., "Exploratory Research In Advanced Automation," First Semi-Annual Report, Menlo Park, CA: Stanford Research Institute, December 1973.
2. Milligan, R. Jr., 1990 *Vision '90 Conference Proceedings*, Machine Vision Association of Society of Manufacturing Engineers, p. 2.
3. Haralick, R. M., Sternberg, S. R., and Zhuang, X., 1987 *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. PAMI-9, p. 532.
4. Maragos, P., *Optical Engineering* 26: 623(July 1987).
5. Dougherty, E. and Giardina, C., *Morphological Methods in Image and Signal Processing*, Prentice-Hall, Englewood Cliffs, 1988.
6. Loughheed, R. M., *Proceedings of the International Electronic Imaging Conference*, Pasadena, CA. 1989, p. 285.
7. Schmitt, L., 1990 *Vision '90 Conference Proceedings*, Machine Vision Association of Society of Manufacturing Engineers, p. 8.
8. Wilson, S., 1990 *Vision '90 Conference Proceedings*, Machine Vision Association of Society of Manufacturing Engineers, p. 7.
9. Budd, G. W., Machine Vision Association of Society of Manufacturing Engineers, Technical Paper MS93-128, 1993.