SPIE Proceedings, Nonlinear Image Processing III, Volume 1658, pages 76-93, February 12-13, 1992.

Morphological Algorithm Development Case Study: Detection of Shapes in Low Contrast Gray Scale Images with Replacement and Clutter Noise

Larry R. Rystrom[1,2] Philip L. Katz[1,2] Robert M. Haralick[2,1]
Christian J. Eggen[1]

[1]Applied Physics Laboratory
[2]Department of Electrical Engineering
University of Washington, Seattle, Wa 98195.

Abstract

We present a case study of the design of a fully autonomous morphological detection algorithm. Grayscale input images contain objects to be detected among difficult clutter, replacement noise, and background tilt. We include our criteria for choosing algorithm structure, with associated grayscale and binary structuring elements based upon comparing the geometry of target and noise/clutter objects. We discuss background cancellation, and histogram-based techniques for final thresholding to binary detection images.

Finally we present a performance characterization methodology for the detection algorithm. In addition to conventional detection statistics we consider the "quality" of the hits and false alarms, vis-a-vis the feature set and classifier used in classification downstream of the detector in the overall system design.

1 Introduction

We describe the design process, algorithm structure, and performance results for a fully autonomous morphological detector. The design is aimed at a class of images from a particular scanner system. In such images, objects to be detected ("targets") are geometrically relatively well defined (though with a random component), but embedded in an often severe random background environment characterized most easily with heuristics and by examples.

Typical input images may contain randomly sized and located replacement noise and clutter objects, as well as grayscale background tilt far exceeding local target/background contrast. Targets vary in size by a 2:1 factor in linear dimension, and may appear distorted or occluded by the physical processes involved. Inspection of images reveals that the attributes distinguishing targets from replacement/clutter "objects" are mainly geometric (shape/size). Because of the predominance of random sized replacement noise, which does not fit the additive linear noise model associated with classical signal processing, and because of the geometric distinction between target and clutter/replacement noise objects, we turned to Mathematical Morphology as is appropriate to such situations. The procedures for designing morphological filters are not as advanced in "state of the art" as is the case for linear filters. For instance, the tools for linear filter design are described in many signal processing texts including [1] and [2]. When the signal is deterministic and the additive noise is random a filter (Matched filter) can be designed that optimizes the signal to noise ratio. When the signal is random the mean square error between the filtered noisy signal and the original signal can be minimized by another filter (Wiener filter) design. The technology for implementing these filters is also well developed and includes FIR, IIR, and FFT based techniques.

By comparison the tools for designing optimum morphological filters are still a current research topic. Haralick, et. al. in [3] study optimal filtering with openings that applies to binary images with objects that are restricted to membership in an ordered basis set. Optimal filter design is also discussed in [4] but there are assumptions made that exclude us from using their

results. Our data set is also discussed in [5] but only in the context of finding features to train a neural net classifier. Early and significant work on this particular detection problem was done by Ballestrasse [6].

2 A Brief Review of Morphology

The two basic operations of Mathematical Morphology are erosion and dilation. References [7], [8], and [9] each give a comprehensive review of the mathematical properties associated with each. Binary dilation of a set F by a set L is defined as $F \oplus L = \{x | x = f + l \; for \; some \; f \in F \; and \; for \; some \; l \in L\}$. Binary erosion of a set F by a set L is defined as $F \ominus L = \{x | x + l \in F \; for \; all \; l \in L\}$. In binary morphology, erosion and dilation are set operations with both F and L being arbitrary sets. But in image processing a clear distinction is made between an image and a structuring element. Hereafter f will represent a grey tone image and l a structuring element. Grey scale dilation and erosion of an image f by structuring element l are respectively defined as $(f \oplus l)(x) = \max_{\substack{z \in L \\ x - z \in F}} \{f(x - z) + l(z)\}$ and $(f \ominus l)(x) = \min_{z \in L} \{f(x + z) - l(z)\}$ where the output image domain changes size according to $F \rightarrow F \oplus L$ for dilation and $F \rightarrow F \ominus L$ for erosion. This is the underlying mathematical definition which assumes that the original image exists only within its domain, F.

Change of image size is a mathematical solution of the edge condition problem common to shifting mask algorithms (including linear filters for example). Mathematically, results are left undefined where the lack of image data at the edge would require local modifications of the definitions. In practice, padding is added to the edges of the original images so that sequences of morphological operations never encounter the shrinking and growing, within the original image frame. By so doing we can always display and consider an image size equal to that of the input image. The spatial extent of the padding depends on the extent of the structuring element. Choice of an appropriate padding values is more subtle, and is a design decision equivalent to assumptions upon objects only partly visible at the edges of the original images. Such design decisions, which reflect the detection goals of the algorithm, are properly segregated from the precise mathematical definition of morphological operations.

The primitives of erosion and dilation can be combined into more complex operations. An opening is an erosion followed by a dilation, $f \circ l = (f \ominus l) \oplus l$, and closing is a dilation followed by an erosion, $f \bullet l = (f \oplus l) \ominus l$.

Since openings are antiextensive, i.e. $f \circ l < f$, and since closings are extensive, i.e. $f < f \bullet l$, it makes sense to define two additional operations based on subtraction. Define the opening residue operation as $f \; openres \; l = f - (f \circ l)$ and define the closing residue operation as $f \; closeres \; l = (f \bullet l) - f$ Residues have the effect of extracting the portion of the image that is smaller than L.

The primitives of opening and closing and their corresponding residues have an easy geometric visualization as described in Figure 1. Our design process is really based on such visualizations and what can be accomplished geometrically in successive grayscale surfaces with a sequence of morphological operations.

3 Description of Random Images

One might conceive of the several hundred images in our database as the observed realizations of a random process. If the random process were known then a statistical description would completely suffice to describe the images. However, since no such statistical description is easily available, we will describe the images by examining some of their typical characteristics. Figure 2 shows a fairly representative image. The dimensions of the images were roughly 200 rows by 300 columns.

3.1 Objects Being Detected (Targets)

The objects we wish to detect are round or ovaloid in shape and are generally darker than the surrounding pixels. Since it was possible to clearly identify what the "ground truth" should be for each image, we created a binary mask image for each image in our data set. The pixels that are a 1 value in the mask correspond to object pixels while 0 valued pixels in the mask correspond to background pixels. The mask images were created with computer graphics software that interactively allows a user to approximate objects as polygons and simultaneously overlays the result. The ground truth information was collected at quite some time and expense; its significance is that having been collected once, it allows algorithm results to be evaluated automatically for repeated algorithm design modifications. It also allowed other characteristics of the objects such as shape and size to be studied further. For instance, we were able to get some of idea of the distribution of sizes of targets by estimating a radius for each ground truth mask image connected component. A histogram of target radius is shown in Figure 3.a.

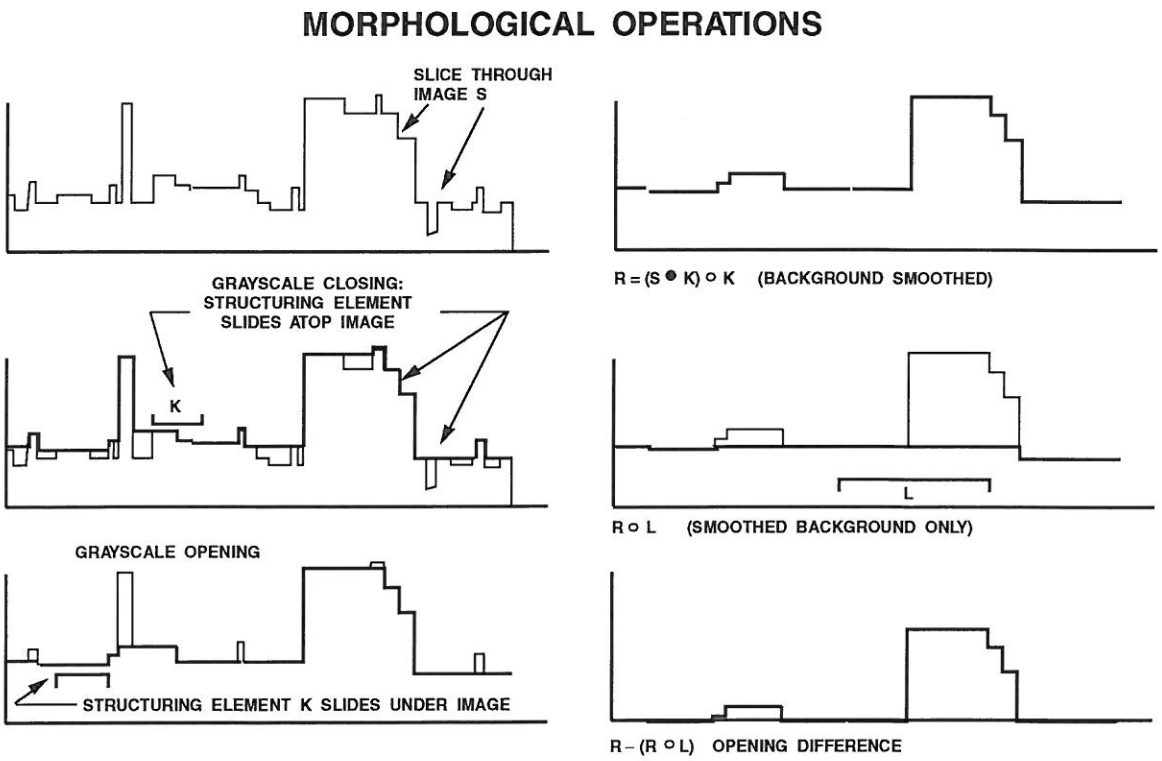
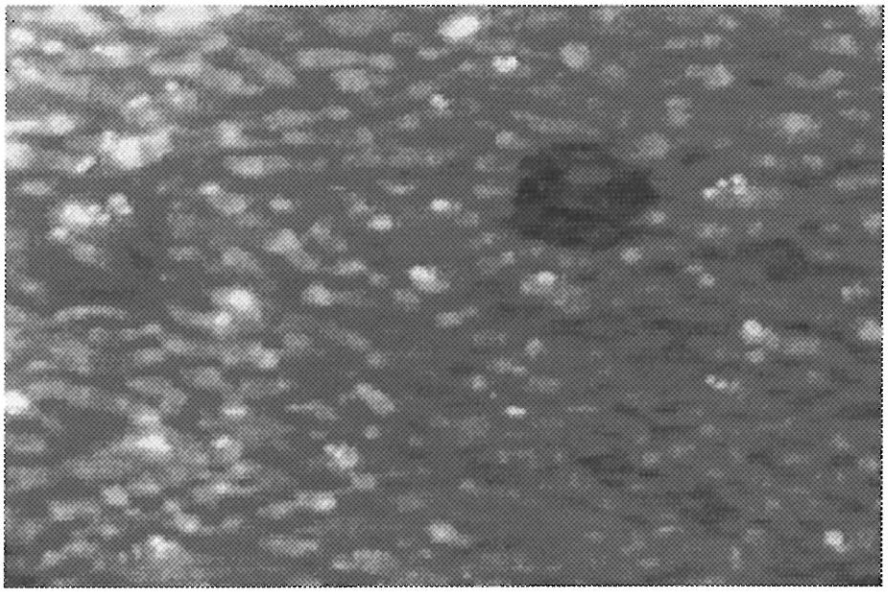
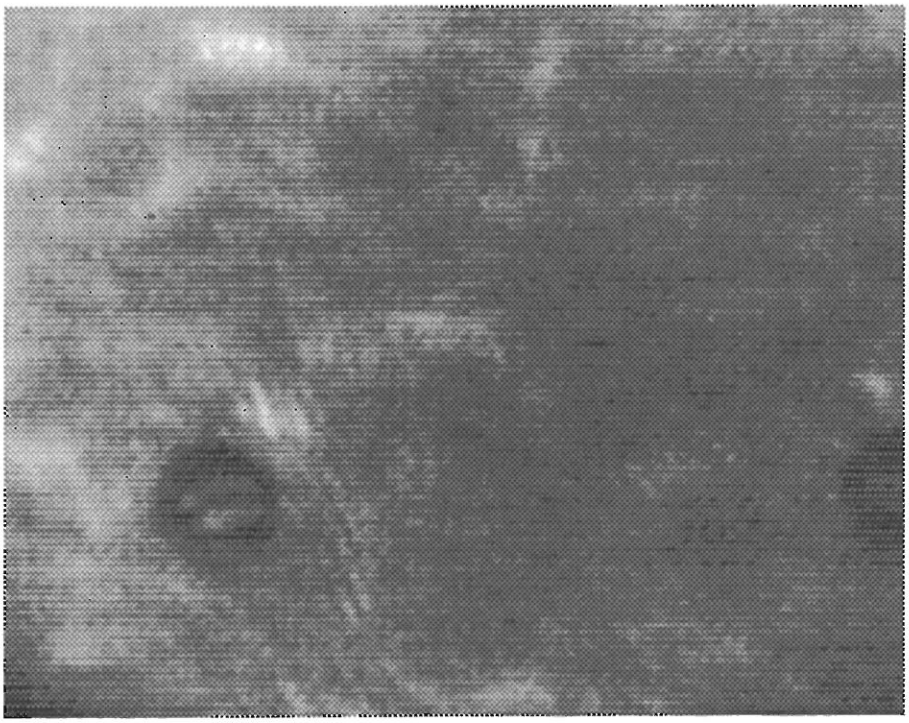


Figure 1: Illustrating Opening, Closing, Opening Residue, Closing Residue, by a flat structuring element. Note the zero background levels in the residues.

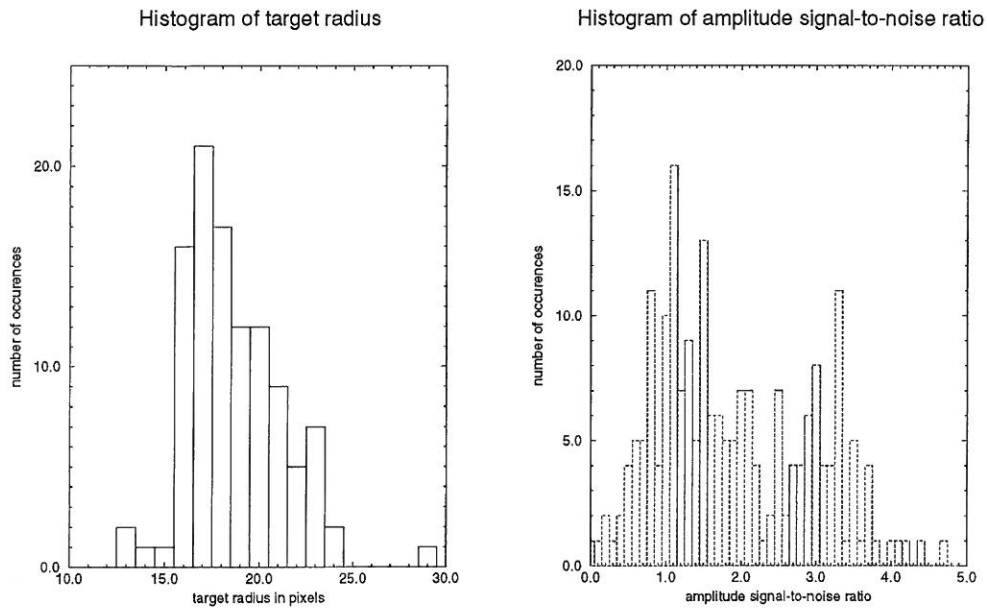


a)



b)

Figure 2: a) Illustrating a typical image (c073696). b) Illustrating a typical image with physical occlusion (s011268).



a) b)

Figure 3: a) Histogram of object radius derived from ground truth. b) Histogram of object signal-to-noise ratio derived from ground truth.

The objects are often very hard to see due to low contrast. Indeed, careful inspection of Figure 2.a is necessary to ascertain that there are not one but two target objects visible. One way to quantify the contrast is given by

$$S^2 = \frac{(\hat{\mu_o} - \hat{\mu_b})^2}{(\hat{\sigma_o^2} + \hat{\sigma_b^2})/2}$$

where $\hat{\mu}$ and $\hat{\sigma^2}$ are estimates of the mean and variance for the portion of the image denoted by the subscript, either object pixels (o subscript) or background pixels (b subscript). The background pixels associated with the object are selected in the following way. The ground truth mask image is dilated with an appropriately sized disk and then the result of the dilation xor'd with the original mask image to form a background pixel mask image. In this way the background pixels in a ring around the object are selected; the number of pixels in the ring is approximately the same as the number of pixels in the object. S for the rightmost object in the image in Figure 2.a is 2.1 which is very low because S can be interpreted as an amplitude signal to noise ratio. The corresponding value of the signal to noise ratio in dB is 6.4. A histogram of S for our image data base is shown in Figure 3.b.

3.2 Background Characteristics

The most significant background characteristic is the presence of cloudlike patches in the image, hereafter called "glint" (That term in fact refers to the agglomeration of effects of several physical phenomena). Glint is readily apparent in Figure 2.a. It is known from the image creation process that glint is replacement noise rather than additive noise. This being the case, we were strongly motivated to employ Mathematical Morphology rather than classical signal processing techniques in our algorithms.

We were able to get some idea of the size of the glint by calculating a pattern spectrum of some representative images. The pattern spectrum is discussed in [9], [10], and [11]. Essentially a pattern spectrum consecutively applies finer and finer sieves to an image. The difference between the amount sieved through at a given size and at the next smaller size is the value of the pattern spectrum at the given size. It corresponds in some sense to the amount of energy in the image for shapes of that size. Because our images are grey tone we normalized them by volume sieved through rather than area sieved through. The pattern spectrum for the image in Figure 2.a is shown in Figure 4. From Figure 4, confirmed in visual inspections, glint appears to be a lighter than background phenomenon generally smaller than typical target objects.

Another characteristic of the background is that it often had a significant tilt. An example of an image with background tilt is shown in Figure 5.a.

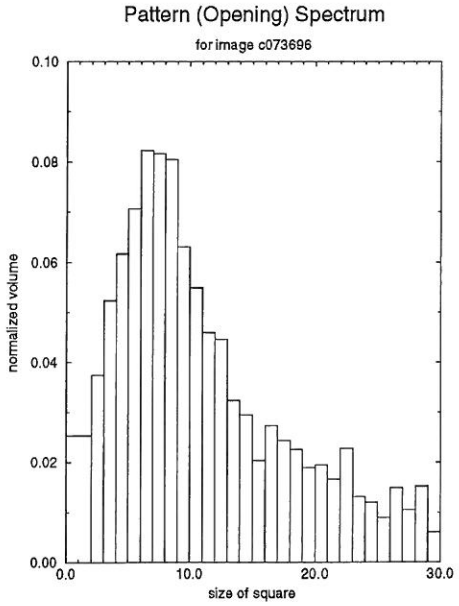


Figure 4: Pattern Spectrum for image c073696.

Figure 5.b is a cross section along a row of the image in Figure 5.a and it demonstrates there is a definite decrease in overall intensity as one moves to the right in the image. Figure 5 shows that the combination of glint and background tilt preclude effective use of simple thresholding. The contrast between object and background is often less than the background tilt; the rightmost object in the image of Figure 5.a, which has extremely low contrast demonstrates this point.

The final property we will describe is another kind of physical occlusion that operates only on target objects because it comes from a physical process different from that which produces "glint." An example of this is in Figure 2.b.

3.3 Representative Image Sets

For purposes of algorithm development, we were provided several hundred images, each containing one or more verified target objects, and a greater number containing no targets. During initial algorithm development we inspected visually, and deliberately chose an unrepresentative set of some 200-300 images with a large number and wide variety of difficult attributes. This was our "test-to-break" data, upon which initial development was performed and failures traced.

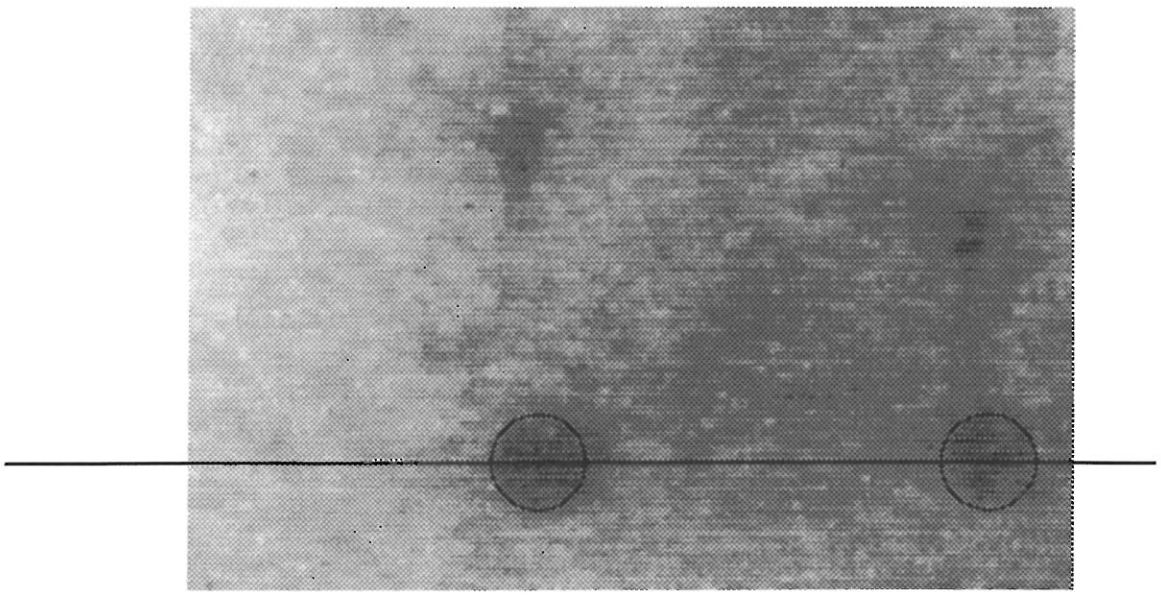
Thereafter, a set of difficulty likely to approximate normal operating conditions was obtained. Technical representatives of the mission sponsor provided a set of 84 frames, containing 210 verified target objects, certified as representative of the average level of difficulty of target-containing frames likely to be encountered in practice. Thereupon we obtained a random selection of approximately the same number (90) of non-target-containing frames, by sampling every n'th frame from all the images we had been supplied, and discarding those containing targets. This set of 174 frames (84 containing 210 targets and 90 containing none) became our "representative set" of images for performance evaluation.

4 Algorithm Design

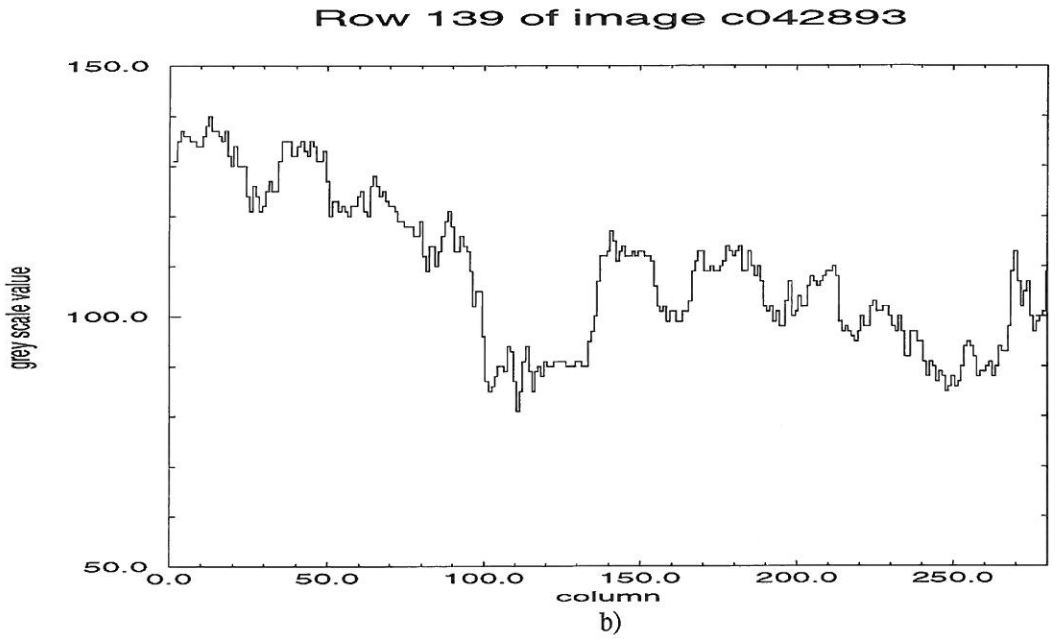
Before proceeding to describe our algorithm in detail, we provide an overall figure, Figure 6, that shows each processing step of our algorithm including glint removal, occlusion removal, grayscale detection, and 1-bit detection.

4.1 Glint Removal

Glint removal is the first algorithm step. We decided to attack it first based on two considerations: 1) glint was smaller in size than the target objects that occurred and morphological filtering naturally progresses from smaller to larger objects; 2) glint removal simplifies the later problem of background normalization/target detection. Our initial attempt at a glint removal algorithm was to apply a small closing with a disk of diameter w_1 to the input image (B) then open this closed image with a disk of diameter w_2 to produce the glint cleaned image (K), i.e. $K = (B \bullet \text{disk}(w_1)) \circ \text{disk}(w_2)$. Before discussing selection of sizes for w_1 and w_2 we must change our attention to a practical consideration.

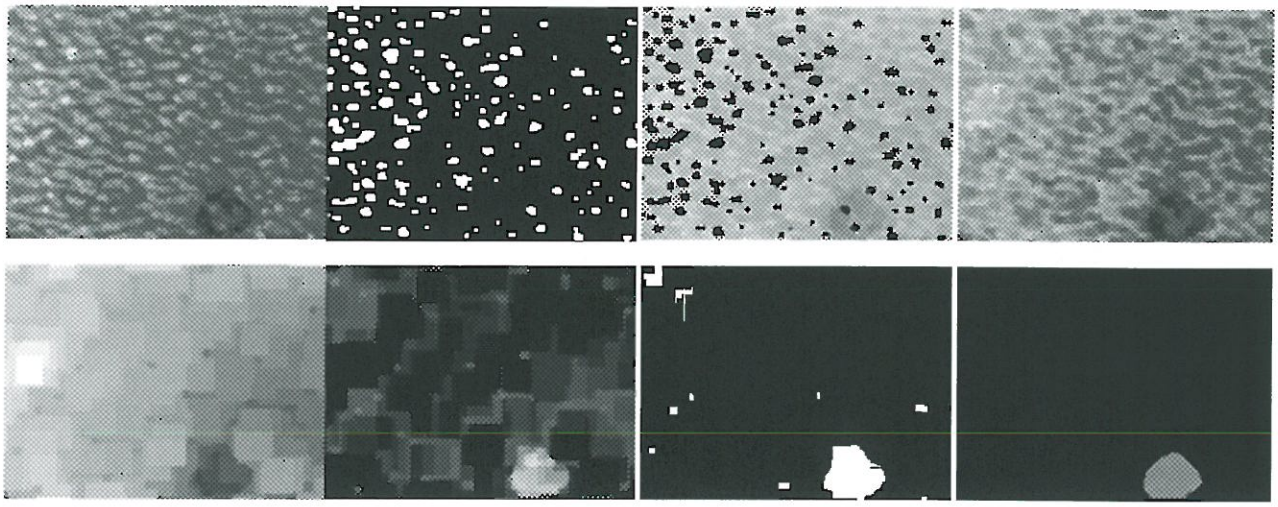


a)



b)

Figure 5: a) Illustrating an image with background tilt (c042893). Size is 183 rows by 279 columns. b) Row 139 of the above image (c042893).



ORIGINAL	GLINT MASK	GLINT NULLED	DE-GLINTED IMAGE
OCCLUSIONS REMOVED	GRAYSCALE DETECTIONS	1-BIT TARGET DETECTIONS	GROUND TRUTH

Figure 6: Illustrating the overall algorithm

We knew that the algorithm we were going to develop would be eventually required to run in real time (see [12]). Thus even at this early stage we determined that we would restrict ourselves to square structuring elements because they can be simply decomposed into two point structuring elements to gain a speedup of approximately $\frac{n^2}{2\log_2 n}$ to 1. With this consideration the glint cleaning algorithm became $K = (B \bullet \text{square}(w_1)) \circ \text{square}(w_2)$.

The size of w_2 is based on ensuring that the cleaning step both cleans glint of all sizes but does not bridge together adjacent objects in the same image and thus distort their size and shape for the next stage of the algorithm. Let s_{obj} be the smallest observed interobject distance. If the opening structuring element is to barely fit between two such objects, in the limiting case when they are diagonally s_{obj} apart, then $w_2 = \frac{s_{obj}}{\sqrt{2}}$. Adding a safety factor gives the working expression $w_2 < \frac{2s_{obj}}{3\sqrt{2}}$. For our data this meant $w_2 = 15$. The purpose of the closing with disk(w_1) is to "protect" the opening and prevent it from getting "pushed down". A secondary consideration is that it not remove target objects. These considerations give $w_1 = min\{\frac{w_2}{3}, \frac{d_{min}}{2}\}$ where d_{min} is the smallest observed target diameter. For our data this meant $w_1 = 5$.

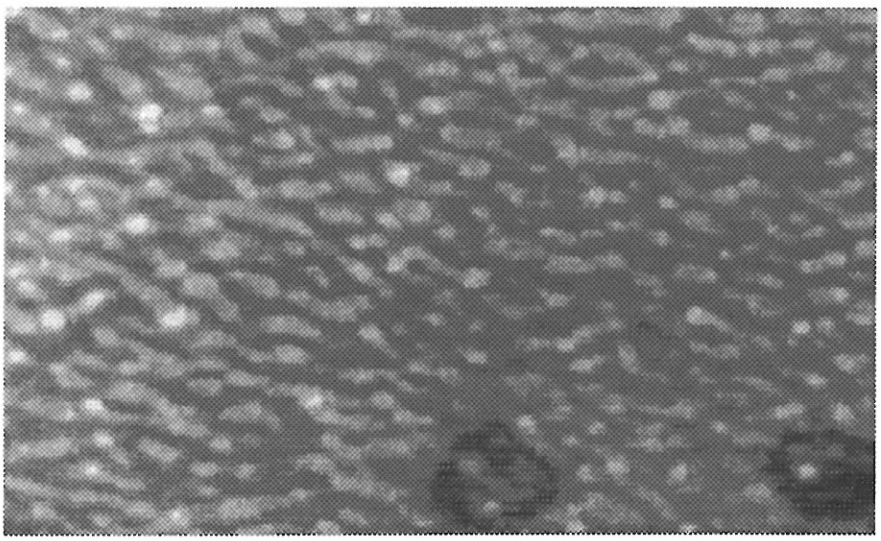
4.2 Enhanced Glint Removal

In testing the initial glint removal algorithm described above the results were not as expected. We expected the glint to be removed and the image mainly unchanged except for being smoother. Instead, in many images we observed a pattern of replication of the structuring elements that is best described as "tiling". Figure 7 depicts the results of our initial algorithm and shows the phenomenon of tiling.

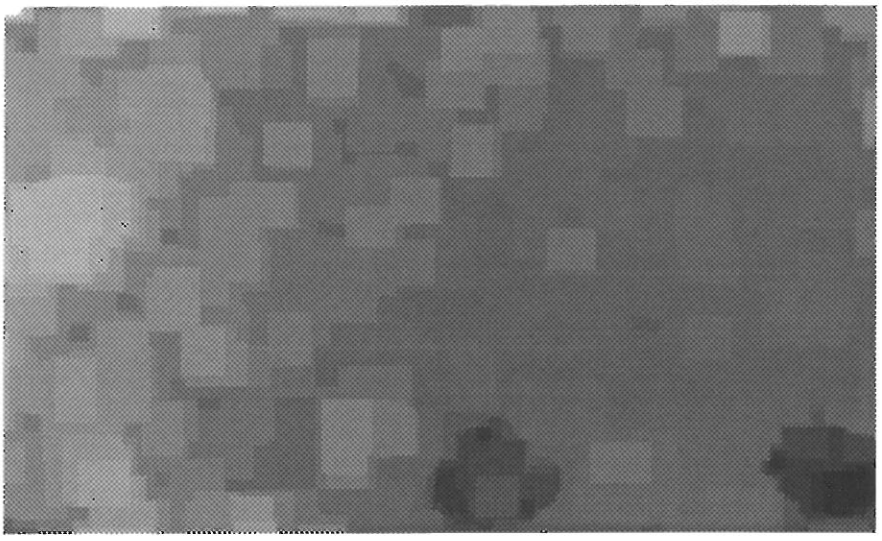
We discovered after some experimentation that the tiles were based upon the size of square(w_2) and were thus an artifact of performing an opening. The tiles made automatic thresholding at the object detection stage of the algorithm difficult because the "objects" created by tiling often tended to become agglomerated together and look very much like the objects we were trying to detect. These problems called for a more sophisticated glint removal algorithm that did not tile the image.

Our approach was to retain the first operation, namely $D = B \bullet \text{square}(w_1)$, in the initial glint removal algorithm of section 4.1, but replace the second operation, namely $\circ$ square(w_2), by an operation that was inherently smoother. Specifically, we created a mask I approximating the glint regions. We then masked the (partly-cleaned) D image where I indicated glint, and filled in the masked pixels from surrounding values, creating a "glint-cleaned" image K; $K = Fill(D, I)$. While there were some "seams" where filling from different edges of masked regions met, such structure was mild and generally not target-like, compared to the results from opening by square(w_2).

It remains to describe the formation of the glint mask I. Glint was considered to be the union of "very bright glint" region C and a region G of pixels brighter than surrounding background, but not agglomerated into regions larger than



a)



b)

Figure 7: a) Image c584910 b) Image c584910 after glint cleaning with initial algorithm

a maximum size. Specifically $C = 1(B > (\mu + 3\sigma))$ is found by thresholding all pixels more than 3σ above the mean. $G = 1(D\ openres\ \text{square}(w_2)\ > T_2)$ is the set of regions lighter than their surroundings where structuring element square(w_2) (chosen as in section 4.1) will not fit. (So G is in a sense the collection of regions that the $\circ$ square(w_2) operation would have cleaned in the initial algorithm). Finally, actual glint mask $I = (C \bigcup G) \oplus$ square(3) where the final small dilation, independent of the size of targets or clutter in images, is an engineering adjustment to account for the tendency of the undilated glint mask to "miss" single-pixel corners and edges of glint regions.

It can be seen, then, that except for the inclusion of "very bright" regions (which have no size limit when they occur) the enhanced glint removal algorithm operates upon the same region as the $\circ$ square(w_2) step in the unsuccessful algorithm of section 4.1; the main difference is a smoother glint-removal step (edge-filling), less prone to tiling.

The T_2 threshold is set at $\max\{\mu + 2\sigma, 0.25 * \max\{D\ openres\ \text{square}(w_2)\ \}, T_3\}$ where the adaptive T_3 threshold process for $openres$ and $closeres$ images will be described in the object detection section of this paper, and the μ and σ values are measured for the opening residue image (not for the original raw image, as was the case in determining the C mask). The upward adjustments in the T_2 threshold, compared to our "standard" thresholding procedure, were motivated by a few instances of dramatically setting the threshold too low, resulting in too much of a "good" image being blanked out as glint and thus ignored.

An example figure describing the result of running both the initial glint algorithm and the enhanced glint algorithm through to detection is Figure 8. It is apparent that the initial glint algorithm performs substantially poorer.

4.3 Occlusion Removal

$L = K \circ \text{square}(w_3)$. w_3 is taken to be larger than $0.5d_{max}$ to ensure that occlusions up to 50% of target diameter are closed over. For our values of d_{max}, this resulted in $w_3 = 21$ pixels.

4.4 Background Normalization and Object Detection

The background is normalized at the same time as extracting the round target objects by performing a closing residue operation. So N, the background normalized image, is calculated by $N = L\ closeres\ \text{square}(w_4)$. The size of w_4 is chosen to be comfortably larger than the diameter of the largest expected target since its job is to fail to "fall" into any object-sized indentations occuring in the glint cleaned image. A reasonable quantification is $w_4 > 1.25d_{max}$ where d_{max} is the largest diameter of object expected to occur. These considerations give $w_4 = 64$ for our data.

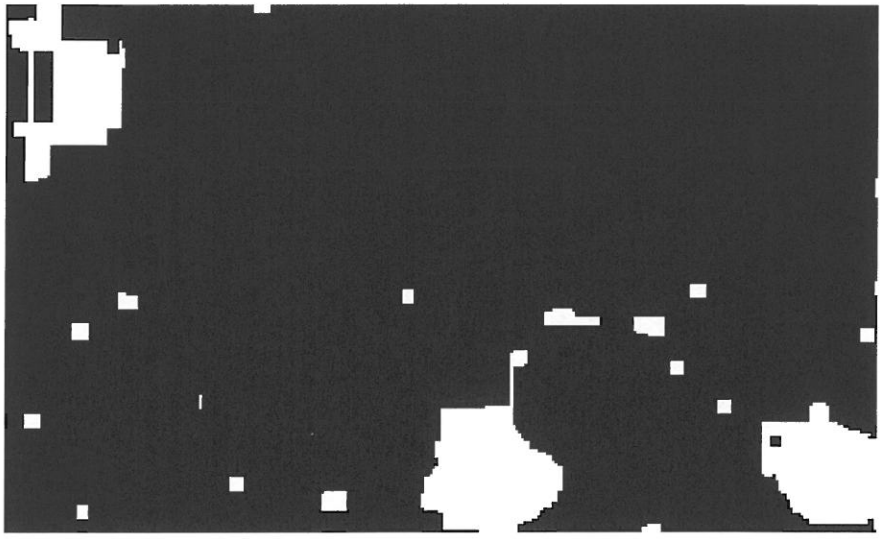
After the glint was cleaned, occlusions removed and the background normalized, the final detection image O containing target objects was found by thresholding $O = 1(N > T_3)$. Because of the highly nonlinear character of morphological operations and the consequent geometry of closing residue image N, it is clear that: 1) N will be identically zero where a square(w_4) can fall perfectly, and close to zero where it can "almost" fall. This dictates that the intensity histogram of image N, on which T_3 must be based, will have the majority of pixels close to zero (where closeness depends on the underlying contrast spreads of the original image absolutely, and of detected objects (in image N) against the surrounding background). The pixels to be retained are thus to be found in a flat histogram "tail" or a second (or greater) mode or both.

The somewhat complex procedure for setting T_3 is motivated by the need to negotiate "blind" (i.e. automatically, and for all cases) in any histogram having the possibilities just defined. In overview, the T_3 threshold is performed as follows.

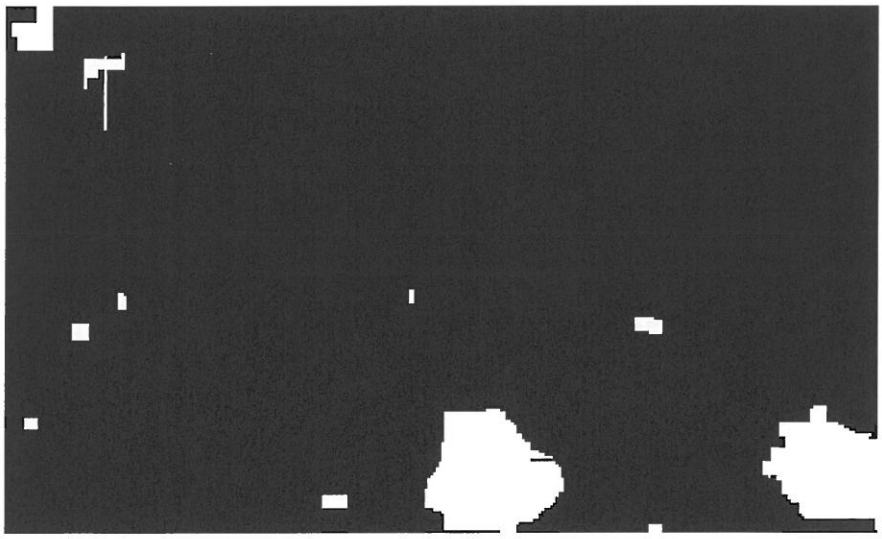
First two separate models of the actual histogram for image N are created, each resulting in an autothreshold. The higher is taken. Then further automatic upward adjustments in the threshold level (decreases in the number of pixels detected) are applied; the algorithms for these are based upon systematic failures of earlier autothreshold algorithm attempts, lacking such adjustments.

Specifically the T_3 threshold begins with the highest of T_{3A} and T_{3B} defined as follows. T_{3A} is the threshold of the Otsu algorithm described in [13]. To obtain T_{3B} an exponential model density curve $P(x) = ae^{-ax}$ is fitted to the first three bins of the histogram. Then T_{3B} is taken to be the threshold corresponding to a false alarm rate of 0.001 in that model, namely $T_{3B} = x_{0.001} = \frac{ln(0.001)}{a}$.

Experiments upon actual data frames showed that even the maximum of the above two thresholds resulted in a value lying in the initial histogram peak, for a non-trivial number of images. (The "symptom" of such a threshold is a "saturated" detection frame with large (0.2 and more) fractions of background detected, mainly random shapes). As a result the threshold must be forced upward until a clear indication of a flat spot or an upturn is obtained. Specifically, beginning with the $max\{T_{3A}, T_{3B}\}$, the threshold is increased by single gray levels until both of the following stopping criteria are fulfilled. (Note that each consists of a major criterion with a "safety valve" to ensure that it will be satisfied before the threshold rises to the maximum



a)



b)

Figure 8: a) Detection with initial glint cleaning method for image c584910. b) Detection with enhanced glint cleaning method for image c584910.

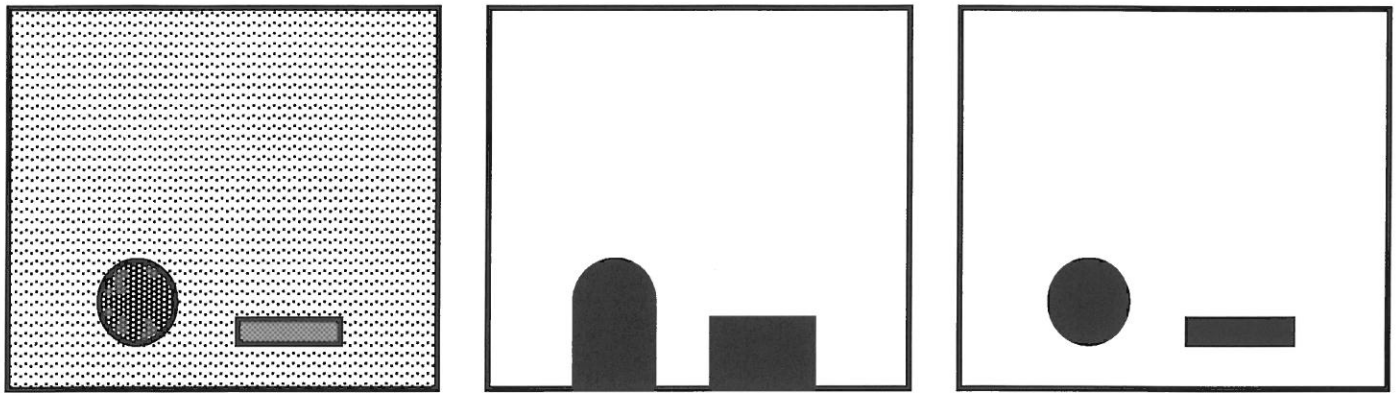


Figure 9: Edge effects in detection. a) Grayscale image; b) Binary detector output, in absence of edge padding/trimming; c) Binary detector output when edge padding/trimming is used.

gray level.) Let $h(i)$ be the the number of pixels in the histogram bin at the i'th threshold iteration, and $htotal$ the total number of pixels in the histogram (total area). Then stop on [C_1 and C_2].

C1: $[h(i+1) - h(i) < 0.0025htotal]$ OR
[threshold is above 99.75% of the total pixels]

C2: $[h(i+1)/h(i) > 0.65]$ OR
$[h(i+1) < 0.000025htotal]$

With the above definition of the T_3 threshold for closing differences, the overall detection algorithm is seen to require no human operator intervention. The three thresholds are set based on the amplitude statistics of the original image and intermediate processed images. The structuring element sizes depend heuristically upon the sizes of objects being detected, and upon assumptions concerning their inter-object distances.

4.5 Edge Effects

In the formal definition of grayscale dilation and erosion, dilation expands, and erosion contracts the domain of a grayscale image. Since the algorithm here is made of openings and closings (each of which erodes, then dilates, or the converse) one might suspect no net effect of growing and shrinking. This is not the case. Edge effects of shrinking and growing images manifest themselves in "edge bridging" as exemplified in Figure 9.

The "cure" for the edge-bridging problem is straightforward (though potentially costly in computation and storage requirements if implemented directly). The initial image B is padded outward in each direction by the diameter of the largest structuring element in the algorithm sequence. Only results for the original domain of B are retained (by trimming) at completion. Padding values are a design decision. We assigned initial values in the rim (padding region) equal to values at the nearest edge point of the original domain of B. The effects in the final detection image are to eliminate bridging (preserving shape), to make partially visible edge targets detectable as if they always extended full-size outward into the rim, but to make partially visible corner targets undetectable. Since the relative sizes of images and target objects dictate that corner targets are infrequent but edge targets are relatively frequent, and shape/size of interior targets would be important in later classification, we were satisfied with the design tradeoffs.

5 Performance Measures

Having described the rationale behind the form of the algorithm the values of parameters, as well as iterative re-designs based on experience with parts of the algorithm, it is desirable to evaluate how well we did. We are able to formulate objective performance measures at this stage only because of the existence of the binary ground truth images.

5.1 Per Image Performance

We wrote software that automatically compares a binary ground truth image and an algorithm output image and produces in tabular form a comparison of the two. Figure 10 is an instance of the output of the "autocomparator" software for the image of Figure 8.a

The first part of Figure 10 shows the number of false alarm pixels, the number of misdetected pixels, and the number of correctly detected pixels. These numbers, while possibly helpful are not so meaningful for morphology which is concerned about shape. Thus the second part of the figure makes the same comparison by connected components; the basis of the comparison is the requirement that at least one pixel must overlap. Finally the third part cross references ground truth connected components to overlapping algorithm connected components and vice versa. The cross reference information on overlap area provides a measure of the quality of the detection and multiple detections. To aid in cross referencing between images and the table, location information about the upper left corner and the bottom right corner of each connected component was included.

In reporting performance of the detector alone, we desired a measure of difficult vs trivial false detections. Since the area of the true target objects being sought was known (in our case between 250 and 1200 pixels), it was obvious (as exemplified in Figure 8) that a large number of false detections were simply too small ("flyspecks") even to be comparable to partially detected targets. Similarly, we observered numerous false detections that were spindly (see Figure 11), likely as a result of residual tiling in the last two steps of the algorithm. Accordingly a spindliness feature was defined, namely $\text{spindliness} = \frac{(\text{major axis length})^2}{\text{area}}$. The major axis length is calculated from the corners of the (extremal point, see [7]) bounding box.

By way of intuitive understanding, spindliness will have values $4/\pi$ for a disk, 2 for a square, N for a 1 X N pixel rod, and N for an "L" shaped object one pixel wide each leg having length N. It was found that by far the majority of connected components that hit targets had spindliness ≤ 1.5

On this basis we designated a false detection "difficult", for purposes of predicting significant work for the classifier, if area ≥ 40 and spindliness ≤ 1.8. In Figure 8.a, one can see that only 2 out of 19 false detections are "difficult" namely algorithm connected components numbered 1 and 8.

5.2 Performance on the Ensemble of Representative Images

We summarize performance on the 174 representative images described in section 3.3. Statistics are presented on detections (of true targets) and false detections.

To summarize detections, one needs to know the number of true targets, the number that were hit (at least one pixel overlap), the total number of hits. From these can be obtained the number of targets missed and the number of excess hits (total number of excess hits beyond one per true target). From these can be obtained p_d, p_{md}, and p_{excess}. In addition, because of potential application criteria entailing an all-or-nothing criterion in judging an image frame to contain one or more targets and a local concentration-of detections criterion, we computed probabilities of detections based on detecting at least one true target per frame, and based on detecting at least 50% of all objects in a frame. We designate these as $P_{D(any)}$ and $P_{D(50\%)}$

In terms of false detections it is necessary to know the total number of false detections, and the number of difficult false detections, which can be expressed per representative pixel area (1000 pixels is the area of the most "typical" target by comparison) or per frame. In addition, since the false detections are by no means uniformly distributed, some measure of the distribution of false detections among frames is desired. Figure 12 summarizes detector performance.

6 Conclusion

We were successful in developing a morphological algorithm that separated the target from the background nearly 90% of the time on a per target basis and well in excess of 90% on a per frame basis. Considering the difficult signal-to-noise ratio and clutter environment this is really quite remarkable. Our success was possible because both the objects of detection and the clutter had substantial shape characteristics that Mathematical Morphology could exploit.

In the overall system design, the detection objects from this algorithm were fed to a feature computation process and a binary tree classifier (which are beyond the scope of this paper). The classification results achieved on the objects found by our detector, based on independent training and testing sets, resulted in satisfactory overall system performance.

Open development questions for the particular algorithm include robustness and residual false detections. In presently ongoing work, the algorithm described in this paper was applied to substantially more difficult clutter environments than the

```
detection matrix for image c584910 ( 191 x 312 ) is:
no signal,no detect   false alarm
miss                  true detect

per pixel:
 54564   2688
    37   2303

per ccm:
 0 19
 0  2

ground truth ccms:

gt              top      bottom  alg
ccm       spind left     right   ccm overlap
 #  area  liness (r,c)    (r,c)    #  area
 1  1323  1.3 (151,175) (189,184) 12  1307
 2  1017  1.2 (153,288) (186,306) 16   996

algorithm output ccms:

alg             top      bottom  gt
ccm       spind left     right   ccm overlap
 #  area  liness (r,c)    (r,c)    #  area
 1  1377  1.8 (  1, 12) ( 64, 12)
 2    18  1.5 (  1, 89) (  3, 94)
 3     7  2.7 ( 64,312) ( 70,312)
 4     5  2.3 (101,312) (105,312)
 5    30  1.4 (102,246) (106,251)
 6    20  1.4 (104,143) (108,146)
 7    38  1.4 (105, 41) (110, 47)
 8    89  2.2 (111,197) (116,213)
 9    71  1.5 (114,226) (120,236)
10    36  1.4 (116, 25) (121, 30)
11    25  1.4 (118,307) (122,311)
12  1736  1.6 (126,183) (191,184)  1  1307
13    25  1.4 (130,239) (134,243)
14     5  2.3 (142, 70) (146, 70)
15    25  1.4 (144,256) (148,260)
16  1328  1.5 (145,290) (190,312)  2   996
17    30  1.4 (149,  8) (153, 13)
18    25  1.4 (172, 81) (176, 85)
19    62  1.4 (177,115) (183,122)
20    20  1.4 (182, 27) (186, 30)
21    19  1.7 (189,231) (191,235)

summary:
number of ground truth ccms = 2
number of detected ground truth ccms = 2
probability of detection = 1
number of false alarms = 19
non ground truth area of image = 57252
false alarm rate (per 1000 pixels) = 0.331866
```

Figure 10: Illustrating output report for image c584910.

Figure 11: Illustrating a detection image with spindly false alarms (c310937).

Representative Data

174 Frames
210 True Targets in 84 Frames
No True Targets in 90 Frames

Detection of True Targets (Per True Target)

Targets Hit: 182, Misses: 28, Excess Hits 10
$p_d = 0.87$; $p_{md} = 0.13$; $p_{excess} = 0.05$

Detections of True Targets (Framewise)

$P_{D(50\%)} = 0.93$; $P_{D(any)} = 0.965$

False Detections

633 False Detections
$R_{FD} = 3.6$ /frame
143 "Difficult" False Detections
$R_{FD(difficult)} = 0.81$ /frame

Distribution of False Detections

51% of FD's in worst 13% of frames, $R_{FD} = 14$ /frame
77% of FD's in worst 26% of frames, $R_{FD} = 11$ /frame
23% of FD's in remaining 74% of frames, $R_{FD} = 1$ /frame

Figure 12: Performance results on representative images

"representative images" of section 3.3. Initial performance was encouraging, with the exception that the problem of "bad frames" (a small fraction of frames containing large numbers of false detections) was exacerbated. Upon inspection, the difficulty was found within the *openres* and *closeres* thresholding routine (T_3 threshold, of Section 4.4). The difficulty was not in general concept; rather it lay in failure to account for extremely irregular (very-many-peaked) histograms. A smoothing routine in histogram space was the successful remedy. These most recent developments will be reported in future papers.

More general open research issues, related to this work include the following: methodologies for constructing optimal morphological algorithms; criteria for choosing the shapes and sizes of structuring elements; assessment of performance degradation when a suboptimal choice of structuring element is made instead of the optimal one; and, performance measures that directly relate to morphology instead of detection.

7 Acknowledgements

This work was funded by the Marine Corps Research and Development Acquisition Center (Code AW), with technical direction and task sponsorship via the Signal and Image Processing Branch of the Naval Coastal Systems Center (Code 2230). Components of the work were also supported by Office of Naval Research (Code 1125 OA) under its Applied Research Laboratory Project program (Grant N0014-90-J-1369).

References

[1] A. V. Oppenheim, A. S. Wilsky, I. T. Young, *Signals and Systems*, Prentice Hall, Englewood Cliffs, New Jersey, 1983.

[2] R. A. Roberts, C. T. Mullis, *Digital Signal Processing*, Addison-Wesley, Reading, Massachusetts, 1987.

[3] R. M. Haralick, E. R. Dougherty, and P. L. Katz,"Model-Based Morphology: The Opening Spectrum", SPIE Proceedings, Image Understanding and the Man-Machine Interface II, Darret, E. B. and Pearson, J. J., Editors, Vol 1972, pp 108-117, 1991. Also submitted to CVGIP: Image Understanding.

[4] D. Shonfeld, J. I. Goutsias, "Optimal Morphological Pattern Restoration from Noisy Binary Images", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 13, pp. 14-29, January 1991.

[5] A. C. Dubey, A. J. Nevis, R. A. Kennedy, G. Dobeck, E. Moritz, "Morpholgical Image Processing and Artificial Neural Network for Automatic Target Recognition (ATR)." Presented at the Second Workshop on Neural Networks, WNN-AIND91, SPIE Vol. 1515, pp. 273-283, February 11-13, 1991, Auburn, Alabama.

[6] C. L. Ballestrasse, "Morphological Detection Algorithm for Targets in Noisy Grayscale Images", Masters Project Report, Department of Electrical Engineering, University of Washington, August 17, 1990.

[7] R. M. Haralick and L. G. Shapiro, *Computer and Robot Vision*, Vol. 1, Addison-Wesley, Reading, Massachusetts, 1992.

[8] C. R. Giardina, E. R. Dougherty, *Morphological Methods in Image and Signal Processing*, Englewood Cliffs, NJ, Prentice Hall, 1988.

[9] J. Serra, *Image Analysis and Mathematical Morphology* , Academic Press, Orlando, Florida, 1982.

[10] P. Maragos, "Pattern Spectrum and Multiscale Shape Representation", *IEEE Trans. on Pattern Analysis and Machine Intelligence*, vol. 11, pp. 701-716, July 1989.

[11] E. R. Dougherty, J. B. Pelz and D. G. Marsh, "Morphological Granulometric Analysis of Electrophotographic Images", Proc. SPIE, Vol. 1153, San Diego, California, August 1989.

[12] A. K. Somani et. al., "Proteus System Architecture and Organization", *International Parallel Processing Symposium*, Anaheim, 1991, also submitted to the International Conference on Pattern Recognition and Computer Vision and Pattern Recognition 1992.

[13] N. Otsu, "A threshold selection method from grey level histograms", *IEEE Trans. Systems Man Cybernetics*, SMC-9, 62-66, 1979.