



PERGAMON

Pattern Recognition 35 (2002) 2125–2139

PATTERN
RECOGNITION

THE JOURNAL OF THE PATTERN RECOGNITION SOCIETY

www.elsevier.com/locate/patcog

Optimal matching problem in detection and recognition performance evaluation

Gang Liu^{a,*}, Robert M. Haralick^b

^a*Cognex Corporation, One Vision Drive, Natick, MA 01760, USA*

^b*Computer Science Department, Graduate Center, The City University of New York, 365 Fifth Avenue, New York, NY 10016, USA*

Received 14 February 2001; received in revised form 3 October 2001

Abstract

This paper proposes a principle of one-to-one correspondence in performance evaluation of a general class of detection and recognition algorithms. Such a correspondence between ground-truth entities and algorithm declared entities is essential in accurately computing objective performance measures such as the detection, recognition, and false alarm rates. We mathematically define the correspondence by formulating a combinatorial optimal matching problem. In addition to evaluating detection performance, this methodology is also capable of evaluating recognition performance. Our study shows that the proposed principle for detection performance evaluation is simple, general and mathematically sound. The derived performance evaluation technique is widely applicable, precise, consistent and efficient. © 2002 Pattern Recognition Society. Published by Elsevier Science Ltd. All rights reserved.

Keywords: Performance evaluation; One-to-one correspondence; Edge detection; Corner detection; Automatic target recognition; ROC curve; Matching problem; Assignment problem; Hungarian algorithm

1. Introduction

Performance evaluation of computer vision systems and algorithms is an important subject of computer vision research [1–12]. It concerns setting up conventions and developing techniques by which we compare different algorithms and gauge the state-of-the-art in a particular subfield of computer vision research.

A broad class of computer vision applications concern detecting and recognizing particular entities in scenes represented by digital images. There is significant amount of similarity in the evaluation of systems developed for such applications. Some commonly used performance metric, such as detection, recognition, and false alarm rates, are important performance indicators in most of these

applications. In empirical performance evaluation, such performance measures can be computed from the proper comparison of algorithm output against ground-truth. In order for the computed performance measures to be correct, accurate and consistent, a systematic approach is needed.

For evaluating detection and recognition performance, we believe that establishing a maximal one-to-one correspondence in conducting the comparison is the key to precise and consistent computation of such performance measures. Thereby, we propose it as a principle in such performance evaluation, and develop a methodology that exemplifies the principle. The exact meaning of the principle is captured in a mathematical formulation of an optimal matching problem. The development of the methodology, i.e., how an optimal matching problem is set up, transformed into an optimal assignment problem, and solved by the efficient Hungarian algorithm, is carried out in a discussion of evaluating edge detection performance [6–10,13–16]. This

* Corresponding author. Tel.: +1-508-650-3285; fax: +1-508-650-3327.

E-mail address: gang.liu@cognex.com (G. Liu).

particular application is chosen because it is adequate for showing most aspects of the methodology, yet simple and familiar enough for a very broad audience to understand the essence of the methodology without the requirement of other application specific background knowledge. The result, however, is readily applicable to evaluating algorithm performance in many other detection and recognition applications, such as corner detection, line segment detection, building detection, and vehicle detection.

Experimental results are reported for performance evaluations in edge detection and automatic target recognition (ATR) applications [17,18]. Compared with a popular existing technique, the new methodology is found to be more precise and accurate. Because it is also general, consistent, conceptually clear, and mathematically sound, we encourage the research community to adopt this methodology as a standard practice in the empirical evaluation of all detection and recognition related applications.

The rest of this paper is organized as follows. Section 2 reviews literature in performance evaluation. Section 3 takes a close look at edge detection performance evaluation. Section 4 provides some background information for the mathematical foundation of the proposed methodology. Section 5 develops and solves the mathematical problem that models the detection performance evaluation scenario. Section 6 derives from the results of Section 5 and presents a practical technique for establishing a maximal one-to-one correspondence between ground-truth and declared entities. Section 7 presents an experimental comparison between the proposed technique and a distance transform based technique for edge detector evaluation. This section also describes how the proposed technique is used in comparing target detection algorithms on a real image sequence. Section 8 extends the proposed technique to provide support for recognition performance evaluation in addition to detection performance evaluation. An example in ATR evaluation is presented to help the reader better understand the details of the technique. Section 9 gives a summary and some concluding remarks. A pointer to the relevant software source code on the Internet is given at the end.

2. Literature review

Performance evaluation is an important and active area of research in computer vision [1–12]. Both theoretical and empirical approaches are being taken. Haralick takes one theoretical approach [1,19,20] and uses the covariance propagation technique [2] to characterize algorithm performance. This approach was also used by Yi et al. [21] to study the uncertainty in the distance between 2D corner points, by Ramesh and Haralick [10] to study the performance of an edge detector, and by Ramesh [3] to study a number of other algorithms.

In empirical performance evaluation [4–9,12,22], algorithms are applied to some test data set. Algorithm performance is summarized by examining the algorithm output. In doing this, both subjective and objective measures of algorithm performance can be used.

Subjective measures are given by human operators who examine the algorithm output and decide how good that result is for a particular application. For example, Heath et al. [6] used subjective measures and studied the performance of five edge detectors. Real images without ground-truth were used in their experiments. Human visual rating of the edge detection result were used to assess the relative performance of the operators. Heath et al. used correlation and analysis of variance (ANOVA) to statistically study the performance data. They also used synthetic data to give an objective characterization in terms of probability of false alarm and misdetection rate. Their test data set of 28 images is made available on the Internet. They gave detailed instructions for others to follow their methodology in evaluating the performance of other edge detection algorithms. Using the subjective measure has the disadvantage of requiring many human operators to assess a significant amount of algorithm output, which can be very costly. Issues of inter- and intra-observer variations also need to be taken care of. Therefore, this methodology is suitable for well-established algorithms, where the significant effort in characterizing the performance is justified. Recently, this research group at the University of Southern Florida works actively on using objective performance measures based on ground-truth. Much effort is put into automating the search for tuning parameters values which correspond to operating points on the receiver operation characteristic (ROC) curve.

Objective measures are some (usually quantitative) values computed from the algorithm output. In many cases, these measures are computed from comparing algorithm output with the ideal result which one wishes the algorithm would produce. The ideal result is called the ground-truth. This is the method that is used in a large number of works in computer vision literature. For example, Borra and Sarkar [4] used 50 aerial images to study the performance of three feature grouping algorithms. The images had manually created ground-truth on the detected edges. Five performance measures were used to compare the algorithms. They used ANOVA to analyze the results. Interpretations of the performance were based on the ranking of the algorithms by the respective measures, as well as on a combined measure. Burlina et al. [23] used empirical objective performance measures to select tuning parameters and performed a sensitivity analysis. Cyganski et al. [24] created artificial test data from real images and clutter. From this, they obtained empirical ROC curves to compare the performance of algorithms. O’Kane et al. [25] reported a study on the effect of image enhancement on human operators’ target identification ability. They studied three kinds of histogram modification methods and different zooming methods. Young

and Cook [26] compared the performance of matched filter with a human performance model in a target detection application. Shufelt [27] used a data set of 83 test images with ground-truth to evaluate monocular building extraction algorithms. Some pixel-based performance measures were used to quantitatively evaluate four algorithms. He also provided an analysis for the impact on building detection performance by scene complexity, image obliquity, and object complexity.

There are also empirical performance evaluation methods which do not require ground-truth. Among these, Kitchen and Rosenfeld [9] used the local edge coherence measure to study edge detectors, Cho et al. [5] used bootstrap methods to study the reliability of edge detector output. Since the criteria used here were arbitrarily set up, there are concerns that these measures may have biased characterization that does not correspond to requirements in real applications.

For the performance evaluation methods that do require ground-truth, there has not been a widely accepted methodology that dictates exactly how the comparison between the ground-truth and algorithm output should be conducted. As a result, it may be very difficult or confusing to directly compare performance characterization results in different studies, simply due to the lack of a common principle or methodology. This makes performance evaluation results less useful. Therefore, a pressing issue in performance evaluation is to answer the question of how to conduct the comparison between algorithm output and the ground-truth. A common platform should be built that is based upon clear, sound and easy-to-follow principles. It will facilitate the development of particular performance evaluation techniques suitable for different applications.

In this paper, we focus our attention on the evaluation of a general class of detection and recognition algorithms. Although characterizing their performance is much simpler a task compared with developing such algorithms, such characterization is often by no means trivial. For example, consider performance evaluation of automatic target recognition (ATR) algorithms [24,28–37]. Performance of such algorithms are customarily characterized by the detection rate, false alarm rate, as well as the recognition rate for the correctly detected targets. These values need to be determined by properly comparing the algorithm output against the ground-truth. Frequently, there are multiple ground-truth target instances in a single test image, and an ATR algorithm may also produce multiple target declarations on that image. The declared targets may or may not be close enough to ground-truth targets, and the ones close enough may or may not have the right identification of the target type. Also, algorithms may claim more than one target around a single ground-truth target, and may also claim targets around clustered ground-truth targets where it is not obvious which ground-truth target(s) should be associated with the declarations.

Faced with this complicated situation, we need to develop a general, concrete and consistent technique that unambiguously and reasonably classifies ground-truth targets into detected and misdetected targets, and classifies ATR algorithm target declarations into correct declarations and false alarms. Also, for the detected ground-truth targets, it distinguishes those that are correctly recognized and those that are not. As a matter of fact, the need for doing such classifications is not unique of ATR performance evaluation. It is a common task in evaluating algorithms for a wide variety of applications, such as building [27] and vehicle detection [22] in aerial image understanding, and biological cell counting in medical image analysis.

3. Performance evaluation problems in edge detection

In assessing the performance of edge detectors [6,9,13–16], we care about the detection rate, false alarm rate and the average localization error for the correctly detected edge pixels. A performance assessment technique needs to classify the edge pixels in the ground-truth into two distinct classes, namely detected and misdetected edge pixels, and to classify the edge pixels in the declared edge map into two distinct classes, namely correct declarations and false alarms. To tolerate a certain amount of localization error, ground-truth edge pixels and declared edge pixels do not have to be at exactly the same pixel location in order for them to be declared as detected ground-truth edge pixels and correct declarations, respectively. However, each detected ground-truth edge pixel needs to be associated with at least one of the correct declarations, and vice versa. Because of the relaxed requirement on the localization, the classification and association between ground-truth edge pixels and declared edge pixels are not trivial problems. Different ways of doing these lead to different performance characterizations.

A distance transform based technique [6,7] has been customarily used for this purpose. A distance map is first obtained where each pixel location is assigned a value which is the distance from this pixel location to the closest ground-truth edge pixel. A threshold on this distance is chosen. The correct declarations are all those edge pixels in the declared edge map whose distance values are not larger than the threshold. The detected ground-truth pixels are those which have declared edge pixels within their neighborhoods. The size of the neighborhood is also determined by the threshold on the distance. The problem with this technique is that it allows multiple-to-one and one-to-multiple correspondence between ground-truth and declared edge pixels. Consider the example shown in Fig. 1, where ground-truth edge pixels are marked by circles, and declared edge pixels are marked by solid squares. According to a distance transform based method, all ground-truth pixels are detected and there is no false alarm declarations.

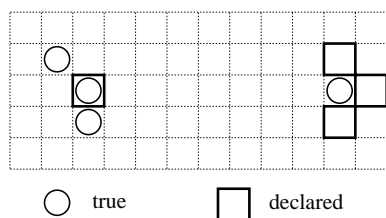


Fig. 1. An example where a distance transform based method fails to give accurate performance measures.

Just like in other detection applications, one would intuitively want to have an exact one-to-one correspondence between detected ground-truth edge pixels and correct declarations. If this is established, the numbers of detected and misdetracted ground-truth edge pixels and the numbers of correct declarations and false alarms will be more informative of edge detection performance. According to this consideration, for the example in Fig. 1, only two ground-truth pixels, the one in the middle on the left half and the one on the right half, are regarded as being detected. The other two on the left half are regarded as being misdetracted. Similarly, only two of the declared pixels, the one on the left half and one of the three on the right half (in this example, it does not matter which one of the three is selected), are regarded as correct declarations. The rest two on the right half are regarded as false alarm declarations. Therefore, a more appropriate performance evaluation method should declare two misdetractions and two false alarms.

Here we propose to establish the desired one-to-one correspondence by setting up a combinatorial optimal matching problem. This problem is transformed into an unconstrained optimal assignment problem, which can be efficiently solved by the Hungarian algorithm [38,39]. The solution yields a maximal one-to-one correspondence between ground-truth edge pixels and declared edge pixels. Performance characterizations based on this correspondence follow the intuition and are more informative of the edge detection performance.

In our experiment, we choose to use a synthetic image to show the difference between two evaluation techniques. Although the use of synthetic imagery in comparing the performance of edge detectors (using a chosen evaluation technique) is questionable [6], it is appropriate in comparing different performance evaluation techniques. This is because that it allows human observers to make a clear and unambiguous judgment of edge detection quality with minimal inter- and intra-observer variations. This judgment acts as a gold standard to which the characterization given by different performance evaluation techniques should be compared. The best performance evaluation technique is the one whose output is closest to that judgment.

3.1. A note on the relevance of performance characterization for edge detection modules

The following development on the edge detector performance evaluation procedure not only serves as an example for how to establish a one-to-one correspondence in the general performance evaluation framework, it is also very important in its own right. This is due to the wide presence of edge detection modules in various computer vision systems, and to the numerous edge detection techniques proposed in the literature.

Since edge detection is usually at earlier stages of computer vision algorithms and is almost never the final goal of any practical system, there is legitimate concern about its performance characterization without the context of a particular vision system. We agree that, due to the many different ways in which edge detection output is used by subsequent vision algorithm modules, the impact of edge detection quality on the entire vision system performance will vary among different vision systems. However, we still insist on the need for a general framework for precise characterization of edge detection quality. This framework needs to adopt a set of quality measures that are general enough to be used in as many different applications as possible. At the same time, the quality measures needs to be capable of carrying enough specific information in order to be useful. For most applications, we believe that the detection rate, false alarm rate, and average localization error constitute a good set of measures for this purpose.

In addition, algorithm performance sensitivity analysis is an important subject in performance evaluation [2,3,19,23]. Highly precise quality measures are essential in studying the sensitivity of the entire vision system performance to the edge detection module. Precision here refers to the sensitivity in the values of the measures to changes in the edge detector output. A quality measure is considered precise if very small changes in the edge detector output are captured and reflected by small changes in the value of the quality measure. As supported by experimental analysis, the performance measures computed from the methodology proposed in this paper is much more precise than a distance-transform based technique, which is being customarily used. We realize that this level of precision might be so high that it is not even necessary for certain computer vision systems, which can be true because of the relative insensitivity of the entire vision system performance to the edge detection module performance. However, we insist on promoting a general framework over ad hoc techniques, especially when such a general framework provides very high precision with fairly low cost.

4. Mathematical foundation—assignment problem and its solution

This section provides a brief introduction to the optimal assignment problem. It also introduces proper notation to

facilitate later discussions. The optimal assignment problem and its solution are essential in solving the problem formulated in the next section. More details of the assignment problem and its solution can be found in standard textbooks such as Ref. [38].

4.1. Unconstrained optimal assignment problem

In the assignment problem, one is concerned with establishing a full one-to-one correspondence between two discrete sets K and L , both of which have N elements. An assignment is a full one-to-one mapping $a : K \rightarrow L$.

Let the cost for associating $k \in K$ with $l \in L$ be $q(k, l)$. The total cost of an assignment a is

$$\sum_{k \in K} q(k, a(k)). \quad (1)$$

The assignment problem is to find an optimal assignment $a : K \rightarrow L$ with the smallest cost among all possible full one-to-one mappings. If there is some constraint between the k 's and the l 's, a constraint which prohibits certain pairings, then the problem becomes the constrained optimal assignment problem. Otherwise, it is the unconstrained problem.

If $q(k, l)$ is finite for all $k \in K$ and $l \in L$, an optimal solution to the unconstrained problem always exists, but might not be unique. However, from the optimization point of view, all optimal solutions are equivalent.

For an integral or rational valued cost function $q(\cdot, \cdot)$, an optimal assignment can be found by the Hungarian algorithm [38,39]. This is a numerical search algorithm that guarantees to arrive at one optimal solution. The computational complexity is $O(N^3)$ [39].

In the context of edge detection performance evaluation, K is the set of ground-truth edge pixels, and L is the set of declared edge pixels. The cost $q(k, l)$ is a measure of the distance between a ground-truth edge pixel k and a declared edge pixel l . Different forms of distance measure can be used, e.g., the L_1 norm (city-block distance), L_2 norm (Euclidean distance), and L_∞ norm. In our experiments, the Euclidean distance squared is used.

In edge detection evaluation, most likely the number of ground-truth edge pixels is not equal to the number of declared edge pixels. It appears that the assignment problem does not apply. However, this can be fixed easily. Conceptually, we can think of adding imaginary pixels to the set with the smaller number of elements. The distance from any imaginary pixel to all pixels in the other set is assigned a finite but very large value. This way, in the assignment the imaginary pixels will not compete with any of the original real pixels for any pixel in the other set. In

the end, all imaginary pixels will also be associated with pixels in the other set. All those pixels that are the counterparts of the imaginary pixels are considered unmatched, and therefore are either misdetected ground-truth edge pixels (if the imaginary pixels were added to the declared edge pixel set) or false alarms in the declared edge map (if the imaginary pixels were added to the ground-truth edge pixel set).

4.2. The Hungarian algorithm

An unconstrained optimal assignment problem is essentially determined by its cost function $q(\cdot, \cdot)$. An exhaustive search for an optimal solution would require a prohibitive $O(N!)$ computations of the total cost in Eq. (1). Fortunately, this can be avoided.

Let the cost function be represented in a matrix form. Adding a constant to any row of the matrix does not change the optimal solutions. (Although adding a constant changes the total cost of the resulting optimal assignment, it is an optimal assignment, not its associated total cost, that is of interest to us). Similarly, adding a constant to any column of the matrix does not change the solutions, either. Therefore, two matrices are considered as in different equivalent forms of a same matrix if one can be obtained from the other via a sequence of additions of constants to various rows and columns. The Hungarian algorithm [38–40] works by finding the maximum number of independent zeros in all equivalent forms of the cost matrix. Due to the König-Egerváry Theorem [38,39,41,42], this is equivalent to finding the smallest number of lines (rows and columns) to cover all zeros.

In this research, we take a black-box approach to the Hungarian algorithm. It is simply used as a proven and efficient tool to solve the assignment problem. In performance evaluation, it is sufficient to know the existence of such a solution, to know its computational complexity, and to have available its software implementation. The details of the algorithm are not of major interest to this research and, therefore, are omitted here. Interested readers can find the details of the algorithm in Refs. [38–42] and the references therein.

In the situation where every element in K is allowed to be paired with every element in L with a finite cost, an optimal solution always exists. Furthermore, if all entries in the cost matrix are integral or rational numbers, the Hungarian algorithm is guaranteed to arrive at one optimal solution with finite iterations. The Knuth implementation [39] of the algorithm handles the integral cost matrix. This implementation also handles the case where the numbers of elements in K and L are not equal, in which case an optimal subset of the bigger set appears in the final pairs. The computational complexity is $O(m^2n)$ where m is the smaller of the cardinalities $|K|$ and $|L|$, and n is the larger.

5. Detection performance evaluation—an optimal matching problem

In edge detection performance evaluation, we know that corresponding ground-truth edge pixels and declared edge pixels cannot be very far apart spatially. It makes sense to set up a threshold on the maximum tolerable localization error, i.e., the maximum tolerable distance between a ground-truth edge pixel and a declared edge pixel that are to be associated with each other. The threshold on the distance between them can be conveniently set by the neighborhood size used in edge detection. To enforce this requirement, all we need is to put a constraint in the optimal assignment problem:

$$q(k, a(k)) \leq \tau \quad \forall k \in K, \quad (2)$$

where τ denotes the maximum tolerable distance. The constrained optimal assignment problem is then to find an assignment $a: K \rightarrow L$ which minimizes the cost in Eq. (1) among all assignments that satisfy the constraint in Eq. (2).

For the constrained problem, the finiteness of $q(k, l)$ does not guarantee the existence of a solution. However, in order to determine misdetections and false alarms in performance evaluation, we have to find some reasonable association, maybe compromised in some way, between the k 's and the l 's.

Taking a closer look at the situation, we realize that the constrained optimal assignment problem in its original form is not an appropriate model for our detection performance evaluation problem. Our problem, however, can be well modeled by an *optimal matching problem*.

The matching problem is similar to the assignment problem in that it tries to establish a one-to-one pairing between elements in K and L . Here the sizes of K and L may be different. There is a compatibility relationship $H \subset K \times L$. Any $k \in K$ and $l \in L$ for which $(k, l) \notin H$ are not allowed to be associated with each other. Only a subset of K needs to be associated with distinct elements in L . The matching problem is to determine a match with the largest number of pairs.

Naturally, for our problem at hand, the compatibility relationship is

$$H = \{(k, l) \in K \times L \mid q(k, l) \leq \tau\}.$$

For any particular match $m: K \rightarrow L$, let $K_s(m)$ be a subset of K containing all the members for which the constraint is satisfied.

$$K_s(m) = \{k \in K \mid q(k, m(k)) \leq \tau\}. \quad (3)$$

The rank of a match m is the number of valid pairs it establishes between the elements in K and those in L . Formally, it is defined as the cardinality (number of

elements) of $K_s(m)$

$$r(m) = |K_s(m)|. \quad (4)$$

In edge detection evaluation, we are only interested in the largest ranking matches, which are called the *maximal matches*. Let M be the set of all possible matches between K and L subject to the compatibility relationship H . Then the set of maximal matches can be formally written as

$$M_x = \left\{ m \in M \mid r(m) = \max_{n \in M} r(n) \right\}. \quad (5)$$

The cost for such a maximal match m is defined as

$$C(m) = \sum_{k \in K_s(m)} q(k, m(k)). \quad (6)$$

The optimal matching problem is to minimize this cost over all maximal matches.

$$m_0 = \arg \min_{m \in M_x} C(m). \quad (7)$$

In edge detection performance evaluation, the solution to the optimal matching problem gives the largest possible number of associated pairs between ground-truth and declared edge pixels. Among all the different ways for making such a match, it picks the one with the smallest total localization error. In a sense, it is trying to give the most positive interpretation of the declared edge map. This is the right attitude, since all computer vision algorithms using the edge map should strive to make the most positive use of it.

5.1. Solving the optimal matching problem

We use the idea of “ghost pair” to transform the optimal matching problem back to the unconstrained optimal assignment problem, solve the optimal assignment problem using the Hungarian algorithm, and then apply some simple post-processing to enforce the constraint and obtain the solution to the optimal matching problem.

Let d denote a very large but finite value. For example,

$$d = |K| \times \tau. \quad (8)$$

The cost function is selectively modified by

$$q(k, l) = \begin{cases} d & \text{if } q(k, l) > \tau, \\ q(k, l) & \text{otherwise.} \end{cases} \quad (9)$$

Then the matrix form of this modified cost function is augmented to a square matrix by padding the shorter dimension with the value d . This corresponds to adding imaginary elements to the smaller set to make the two sets of equal size. We now have an unconstrained optimal assignment problem that is defined by this augmented cost matrix. Each pair of elements in K and L whose cost gets changed to d is called a “ghost pair”. Since d is such a large value, the two elements in a ghost pair do not compete for each other against other elements in the assignment process. Notice that, each ground-truth edge pixel is typically involved in many ghost pairs since there are almost always declared edge pixels that are faraway from it. The same is true for each declared edge pixel. However, some detected ground-truth edge pixels and some correctly declared edge pixels are also involved in pairs that are not ghost pairs. In the association process, it is these non-ghost pairs that are of interest to us. Nonetheless, the provision for the ghost pairs is necessary in order to change the hard constraint on the compatibility relationship into penalty terms in the total cost. This is the key element in transforming the constrained matching problem into an unconstrained assignment problem.

At least one optimal solution to this new unconstrained assignment problem exists due to the finiteness of the cost function. Such a solution can be found by the Hungarian algorithm. In the resulting solution, we examine again the cost between the assigned pairs, to enforce the compatibility relationship. The pairs whose cost are not larger than τ are good associations. They give corresponding detected ground-truth edge pixels and correct declarations. The cost of each of the rest of the pairs is larger than τ , and these pairs must either be ghost pairs, or involve the imaginary pixels added in augmenting the cost matrix.

Remember that the original distance between the two elements in a ghost pair is larger than the threshold. In edge detection performance evaluation, we do not allow pixels that are farther apart than the threshold to be associated with each other. The two elements in a resulting ghost pair are then determined to be a misdetected ground-truth edge pixel and a false alarm in the declared edge map.

The other kind of pairs whose cost is larger than τ involve imaginary pixels. The real elements in such pairs come from the set whose number of elements is larger than that of the other set. Therefore, these are the “left-overs” that did not succeed in competing for the limited number of counterparts from the other set. If the imaginary elements were added to the set of ground-truth pixels, that means there are too many edge pixels in the algorithm output. Therefore, these left-overs reflect false alarms. Otherwise, if the imaginary elements were added to the set of declared edge pixels, that means there are not enough edge pixels in the algorithm output. The left-overs reflect misdetections.

6. Practical technique—association procedure

6.1. Edge detection result classification and association procedure

Note that the Hungarian algorithm can be implemented to handle cases where the numbers of elements in K and L are different [39]. The conceptual addition of imaginary elements in augmenting the cost matrix does not need to be literally carried out. An implementation can achieve the same effect without the extra computational burden required in doing that. Therefore, the cost matrix augmentation step does not need to be explicitly carried out.

The following is an outline of the edge detection result classification and association procedure.

Step 1: Resolving the simple cases:

- The ground-truth edge pixels which coincide at exactly the same pixel locations with some declared edge pixels are automatically determined as detected ground-truth edge pixels. They are associated with those declared edge pixels at the same locations, hence they have no localization error. The declared edge pixels that these ground-truth pixels are associated with are automatically determined as correct declarations.
- The ground-truth edge pixels which do not have any declared edge pixel within a distance of the chosen threshold τ are automatically determined as misdetected ground-truth edge pixels.
- Similarly, the declared edge pixels which do not have any ground-truth edge pixel within a distance of the chosen threshold τ are automatically determined as false alarms.

The rest of the procedure deals only with the ground-truth and declared edge pixels that are left undetermined.

Step 2: Let K be the set of ground-truth edge pixel locations, L be the set of declared edge pixel locations. Let N_K and N_L be the numbers of elements in them, respectively. Create an $N_K \times N_L$ matrix Q of distance values between each pair in $K \times L$.

Step 3: Selectively modify Q by applying the rule in (9), with N_K as $|K|$ in (8).

Step 4: Apply the Hungarian algorithm on the modified Q .

Step 5: Examine the resulting assignment.

- For each assigned pair
 - if the value in the distance map Q is not larger than τ , this is a valid association. The involved ground-truth and declared edge pixels are paired up and to be counted as a detected ground-truth pixel and a correct declaration;
 - otherwise, this is a ghost pair. If this pair is denoted by (k, l) , k is counted as a misdetected ground-truth edge pixel, and l is counted as a false alarm.

- The left-over elements in K (if $N_K > N_L$) or L (if $N_K < N_L$) that are not paired up are all misdetected ground-truth edge pixels (if $N_K > N_L$) or false alarms (if $N_K < N_L$).

6.2. Corner and line segment detection performance evaluation

The exact procedure just described can be applied directly without any change in evaluating algorithms for many kinds of detection applications. Two simple examples are corner detection and line segment detection. These are important because many complex vision systems use corners and line segments as basic image feature units. The performance of corner and line segment detection modules can significantly affect the overall performance of the entire system.

Evaluating corner detection performance is almost identical to evaluating edge detection performance. In most applications, the Euclidean distance squared is suitable for constructing the distance matrix Q . This was used in a study to compare three different corner detection algorithms [43–45]. To limit the length of this paper, details of this evaluation is not included here.

Evaluating line segment detection performance just requires a different distance measure in constructing the distance matrix Q . A weighted Hausdorff distance [46] could be used in this case.

$$q(S_1, S_2) = \max \left(\frac{\|S_1\|}{\|S_2\|}, \frac{\|S_2\|}{\|S_1\|} \right) \max(h(S_1, S_2), h(S_2, S_1)), \quad (10)$$

where S_1 and S_2 denote two line segments, and the double vertical bars denote the Euclidean norm (length) of a line segment. The function $h(\cdot, \cdot)$ denotes the (directed) Hausdorff distance [46] defined as

$$h(S_a, S_b) = \max_{x \in S_a} \left\{ \min_{y \in S_b} \|x - y\| \right\}. \quad (11)$$

$q(S_1, S_2)$ in Eq. (10) is the balanced Hausdorff distance between the two line segments S_1 and S_2 weighted by the ratio

in the length of the two segments. Therefore, it penalizes both the difference in length and the difference in location and orientation of two line segments.

7. Experiment

7.1. Comparing with distance transform based evaluation technique

We use the synthetic test image used in Ref. [6] to compare the proposed method with a distance transform based method for edge detection performance evaluation. This test image is 64×64 , and has a brighter disk of a constant gray value against a darker constant background. White Gaussian noise is added to obtain a noisy version of the image with $\text{SNR} = 4$. Fig. 2(a) shows the ground-truth edge map obtained by following the boundary (using 8-connectivity) of the disk on the noise-free image. There are a total of 132 ground-truth edge pixels and 3964 background pixels. Notice that this ground-truth is different from that used in Ref. [6], where a three-label (true-positive, don't-care, and false-positive) ground-truth is used. Here we want to give sharper performance measures and do not specify the "don't-care" region in the ground-truth. Our method, however, can easily accommodate the provision for the "don't-care" zone in situations where it is desired. This is done by re-examining the false alarms and omitting the ones falling in that zone.

We apply the implementation of Canny's edge detector used in Ref. [6] to the noisy image with different tuning parameters. Fig. 2(b) shows the edge map produced from the tuning parameters of (1.05, 0.77, 0.88) given in Ref. [6]. We also choose two other sets of parameters which give more obvious misdetections and false alarms, respectively. Figs. 2(c) and 2(d) show the resulting edge maps.

The proposed optimal matching based method and the distance transform based method are used to compare the

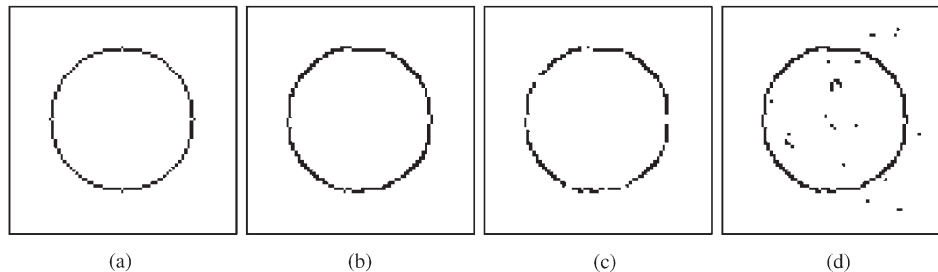


Fig. 2. Edge maps in comparing the proposed and a distance transform based performance evaluation methods. (a) Ground-truth (b) Edge map 1. (c) Edge map 2. (d) Edge map 3.

Table 1
Performance measures for the declared edge maps by two performance evaluation methods

Edge map	Optimal matching			Distance transform		
	#MD	#FA	RMS loc error	#MD	#FA	RMS loc error
1	0	42	0.5	0	0	1.0
2	9	28	1.39	0	0	1.27
3	0	74	1.0	0	28	1.0

declared edge maps with the ground-truth edge map. The distance transform based method used in Ref. [6] has a circular search radius of three pixels for edge (true-positive) pixels. Our implementation of this method used in the experiment reported here uses a square search region of 5×5 pixels with the origin at the center. The Euclidean distance squared is used to form the distance matrix of the proposed method. The threshold τ in inequality (2) is set to 8, corresponding to a circular search radius of $2\sqrt{2}$. Table 1 gives the performance measures of number of misdetections (#MD), number of false alarms (#FA), and the root-mean-squared (RMS) localization error for the detected ground-truth edge pixels from these two methods.

It should be noted that the purpose of the current experiment is to demonstrate the appropriateness of enforcing one-to-one correspondence for counting misdetections and false alarms and calculating localization error for correct detections. It is not our purpose here to carry out a full empirical evaluation of any edge detector. No effort is made to tune the detector to achieve its best performance according to the performance measures.

When examining the data in Table 1, the emphasis should be on comparing the same performance measures given by the two different methods.

In visually comparing edge map 1 with the ground-truth, we see no misdetection. However, the thickness of the edge is not even, with some parts being 8-connected and

some parts being densely 4-connected. According to our 8-connected thin ground-truth edge map and the one-to-one correspondence principle, these thicker parts of the declared edge (where edge pixels have more than two 8-neighbors) contain false alarms. The proposed method precisely picks out these declared pixels as false alarms, which are shown in Fig. 3(a). On the other hand, the distance transform based method leniently accepts all declared edge pixels as being appropriate. If, however, this level of detail is not of interest, a “don’t care” zone can be specified in the ground-truth. The false alarms declared by the proposed method falling into that zone will then be tolerated and not counted as false alarms. If that were done, both methods would claim no false alarm.

The difference between the two methods is clearer on their evaluation results on edge maps 2 and 3. In edge map 2, we clearly see gaps in the declared edge map which we certainly want to call misdetections. At the same time, we also see some false alarms on thicker parts of the declared edges. Both of these are reflected in the evaluation result by the proposed method. The identified misdetection and false alarm pixels are shown in Fig. 3(b) and 3(c), respectively. The distance transform based method is overly lenient again and declares no misdetection nor false alarm.

In edge map 3, the proposed method treats not only the stray edge pixels as false alarms, but also some on the thicker parts of the edges around ground-truth edges. These pixels are shown in Fig. 3(d). The distance transform based method treats only those stray edge pixels as false alarms, and tolerates all declared edge pixels around ground-truth edge pixels.

In general, we observe the tendency of the distance transform based method to be overly reluctant to declare false alarms around ground-truth edge locations, and overly reluctant to declare misdetections around any declared edge pixel. It gives inappropriately high performance measures. The optimal matching based method is more precise, and gives performance measures agreeing with subjective evaluation.

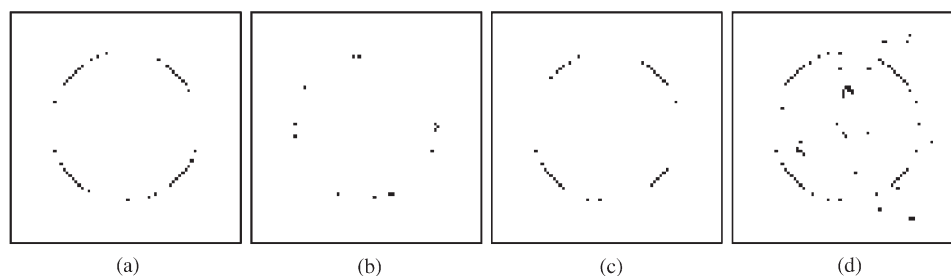
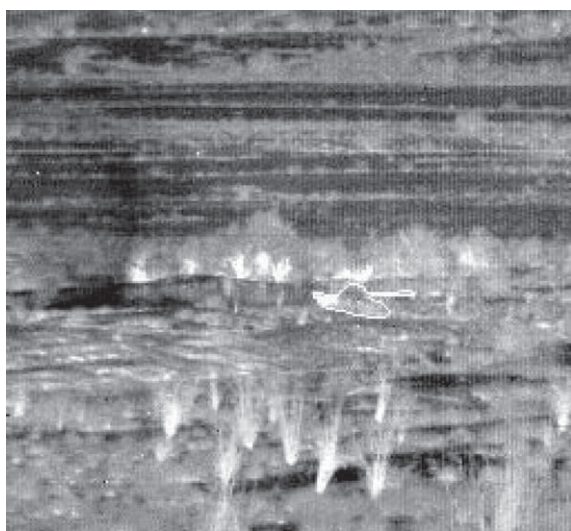
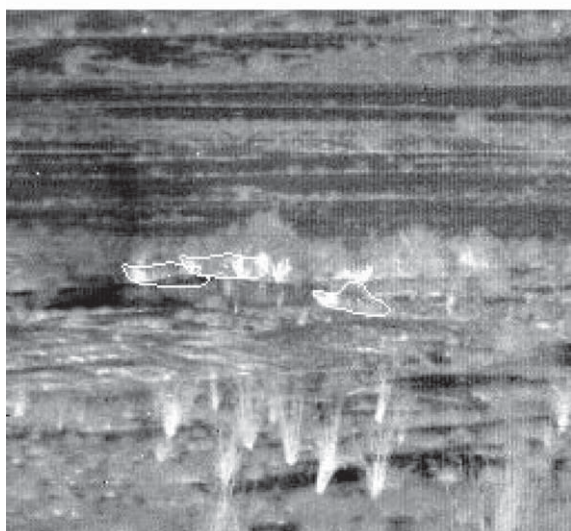


Fig. 3. Misdetection and false alarm edge pixels in the edge maps. (a) False alarms in map 1. (b) Misdetections in map 2. (c) False alarms in map 2. (d) False alarms in map 3.



(a)



(b)

Fig. 4. First frame in the tank image sequence and detection result by the combined algorithm. (a) True target. (b) Detection result by one ART algorithm.

7.2. Evaluating tank target detection performance

The proposed method was applied to comparing the performance of six algorithms in detecting a moving tank target in a sequence of 340 forward-looking infrared (FLIR) images. This sequence was acquired from a newly developed low-cost uncooled sensor, which is known to produce low quality imagery. In many of the 340 images, one single tank target is present and not significantly occluded. In others, the tank target is either present but with some parts occluded by the terrain, or not present in the image

at all. Out of the 340 frames, there are 244 frames where the target is either not occluded, or not significantly occluded. Fig. 4 shows the first frame of the sequence with the true target boundary outlined. Also shown in the figure is the detection result on this image by one of the algorithms studied.

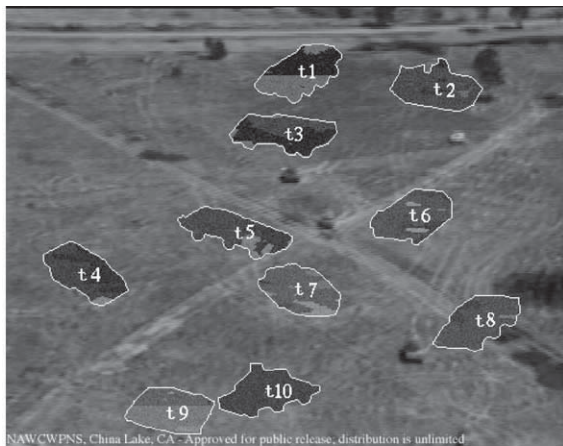
Six target detection algorithms were run independently on each of the images in the sequence. Thresholds in the algorithms were changed over a range of values. At each threshold value, the output from each algorithm on each image was compared with the ground-truth for that image using the technique described in the previous section. The comparison results for all 340 images together were used to compute the detection and false alarm rates for that algorithm at a detection performance level corresponding to that threshold value. This gave one point on the ROC curve for that algorithm. The points corresponding to all tested threshold values make up the entire ROC curve for that algorithm. The statistical Z-test [47] was conducted to examine the significance of the difference between the performance of the algorithms. More details of the algorithms and the evaluation results can be found in Ref. [48].

8. ATR performance evaluation

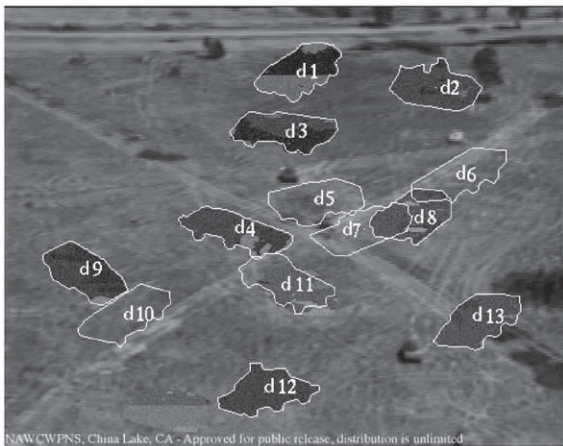
The proposed methodology is useful in many applications where a maximal one-to-one correspondence is to be established between two sets, a distance-like dissimilarity/penalty measure can be made between the elements in the two sets, and the pairing process is constrained by some threshold on the dissimilarity/penalty measure. We choose to base our discussion first on the edge detection application not only due to the importance of edge detection as a major feature extraction module in many computer vision algorithms, but also due to the relative simplicity of its performance evaluation. This way, the main idea of the proposed methodology can be described more clearly without confusion from other unrelated application-specific issues.

Now that the methodology has been described clearly, we use it to develop a procedure for ATR performance evaluation. The purpose of this section is two-fold. First, it shows how the proposed methodology is easily extended to support recognition performance evaluation. Second, it gives a detailed example to let the reader walk through the procedure in order to help the understanding.

In characterizing ATR performance, not only do we need to determine the detection rate and false alarm rate, we also need to compute the recognition rate for the correctly detected targets. In this situation, a two-stage procedure needs to be used. This procedure first identifies targets that are both correctly detected and correctly recognized. It then identifies targets that are correctly detected but incorrectly recognized. In the first stage, the distance matrix is constructed so that only targets of the same class ID can po-



(a)



(b)

Fig. 5. Ground-truth and declared targets compared by the two-stage matching procedure to determine the detected, recognized and false alarm targets. (a) Ground-truth targets. (b) Declared targets.

tentially form pairs. The targets paired up by this stage are those both correctly detected and correctly recognized. The left-overs go through the second stage. In this stage, the target class ID does not play any role in the construction of the distance matrix. The pairs formed in this stage involve only targets that are correctly detected but incorrectly recognized.

The number of correctly detected targets is the sum of the numbers of pairs produced by the two stages. Among these pairs, those produced by the first stage are the ones for the targets also correctly recognized. The left-overs from the second stage are the misdetrcted and false alarm targets. All the pairs and left-overs provide the precise information required in constructing the confusion matrix to show the recognition performance of an ATR system. This procedure

Table 2

Targets in the example on using the proposed method for ATR performance evaluation

Ground-truth			Declared		
Reference	Class	Centroid	Reference	Class	Centroid
t_1	21	(268, 60)	d_1	21	(269, 59)
t_2	1	(391, 74)	d_2	1	(392, 74)
t_3	18	(253, 114)	d_3	18	(255, 113)
t_4	26	(72, 243)	d_4	24	(211, 205)
t_5	24	(209, 205)	d_5	19	(284, 177)
t_6	31	(370, 191)	d_6	25	(414, 154)
t_7	29	(268, 259)	d_7	15	(327, 203)
t_8	23	(429, 287)	d_8	31	(371, 191)
t_9	27	(149, 367)	d_9	26	(73, 243)
t_{10}	9	(236, 351)	d_{10}	3	(111, 281)
			d_{11}	4	(256, 251)
			d_{12}	9	(237, 351)
			d_{13}	23	(431, 286)

is used in the experiments in Refs. [18,48] for evaluating ATR algorithm performance.

An example is provided here. Fig. 5(a) shows one of the test images used in this study. There are ten target instances in this image, with their silhouettes overlaid and an instance number given by which they are referenced. These targets are referred to as t_1, \dots, t_{10} . The first “class” column in Table 2 gives the class identification of these targets. The first “centroid” column in the table gives the positions of the centroids of these targets in image column and row coordinates.

Fig. 5(b) shows the output from one algorithm with the silhouettes of the thirteen declared targets overlaid on the image. The image also shows the instance numbers for these declared targets, which are referred to as d_1, \dots, d_{13} . The second “class” and “centroid” columns in Table 2 gives the classes and locations of these declared targets.

The Euclidean distance squared is used in constructing the distance matrix in the proposed performance evaluation method. In tolerating the location error, a threshold on the squared distance of $\tau = 625 \text{ pixel}^2$ is used, corresponding to a circular search region of radius 25 pixels.

In the first stage of the procedure, only targets of the same class (same class ID, not instance number) can be paired up. For any pair of ground-truth and declared targets whose class ID numbers are different, a distance value of 626, which is greater than the threshold $\tau = 625$, is assigned, regardless of the actual distance between the centroids of these targets. The actual distance is used only for pairs of targets whose class ID numbers are the same. Fig. 6 shows the resulting distance matrix. Subjecting this matrix to the procedure in Section 6, we find eight pairs—eight out of the ten ground-truth targets are both correctly

	d_1	d_2	d_3	d_4	d_5	d_6	d_7	d_8	d_9	d_{10}	d_{11}	d_{12}	d_{13}
t_1	2	626	626	626	626	626	626	626	626	626	626	626	626
t_2	626	1	626	626	626	626	626	626	626	626	626	626	626
t_3	626	626	5	626	626	626	626	626	626	626	626	626	626
t_4	626	626	626	626	626	626	626	626	1	626	626	626	626
t_5	626	626	626	4	626	626	626	626	626	626	626	626	626
t_6	626	626	626	626	626	626	626	1	626	626	626	626	626
t_7	626	626	626	626	626	626	626	626	626	626	626	626	626
t_8	626	626	626	626	626	626	626	626	626	626	626	626	5
t_9	626	626	626	626	626	626	626	626	626	626	626	626	626
t_{10}	626	626	626	626	626	626	626	626	626	626	1	626	626

Fig. 6. Distance matrix used in the first matching procedure for determining targets both correctly detected and correctly recognized. Threshold to be used is 625.

Table 3
Targets that are both correctly detected and correctly recognized

Ground-truth	t_1	t_2	t_3	t_5	t_6	t_4	t_{10}	t_8
Declared	d_1	d_2	d_3	d_4	d_8	d_9	d_{12}	d_{13}
Class ID	21	1	18	24	31	26	9	23

detected and correctly recognized. These are shown in Table 3.

The targets left unpaired are t_7, t_9 on the ground-truth side and $d_5, d_6, d_7, d_{10}, d_{11}$ on the declaration side. Any valid pairing between them that satisfies the distance constraint identifies correctly detected but incorrectly recognized targets. The distance matrix for the second stage of the procedure contains only the actual distances, as shown in Fig. 7. The solution is obvious: only (t_7, d_{11}) is a valid pair whose distance is smaller than the threshold.

Now the full characterization of the algorithm output in Fig. 5(b) can be summarized as follows:

- Of the 10 ground-truth target instances
 - nine are correctly detected, eight of which are also correctly recognized;
 - (redundant) one was misdetected; one was correctly detected but incorrectly recognized.
- Of the 13 declared target instances

	d_5	d_6	d_7	d_{10}	d_{11}
t_7	6980	32341	6617	25133	208
t_9	54325	115594	58580	8840	24905

Fig. 7. Distance matrix used in the second matching procedure for determining correctly detected but incorrectly recognized targets. Threshold to be used is 625.

- nine are correct declarations, eight of which also have the correct class ID;
- (redundant) four are false alarms which do not correspond to any ground-truth target; one correct declaration has a wrong target class ID declared.

9. Summary and conclusion

We propose enforcing a maximal one-to-one correspondence as a general principle to follow in evaluating detection performance. We argue that it leads to a most reasonable association between the ground-truth entities and the entities declared by algorithms. This applies to a broad class of detection and recognition applications. We

develop the principle mathematically by formulating an optimal matching problem. This problem is transformed into an unconstrained optimal assignment problem, which is known to have an efficient solution technique, namely the Hungarian algorithm. The solution to the problem establishes a one-to-one correspondence between the detected ground-truth entities and the correct declarations in the algorithm output. The correspondence also determines misdetected ground-truth entities, false alarms in the declaration, and the computation of localization error for the detected ground-truth entities. We showed examples where the appropriateness, preciseness and generality of this technique are verified.

Although the main development of the principle and the derived technique is carried out in an edge detector evaluation scenario, this methodology is readily applicable to evaluating performance of other applications, such as corner and line segment detection, and building and vehicle detection in aerial images. These applications share the common feature that both detection performance and localization performance are of interest, and that they are closely tied together.

We extend the technique to support recognition performance evaluation. This requires a two-stage procedure, which is outlined for the ATR application. This procedure can be applied with marginal or no modifications for other detection and recognition applications, such as evaluating the performance of multiple category biomedical cell counting algorithms.

From the development of the methodology, as well as from its successful application in a number of different detection and recognition evaluation experiments, we believe that the one-to-one correspondence principle is general, clear and mathematically sound. The technique derived from this principle is precise, consistent and efficient. Therefore, we highly recommend the principle and technique be adopted by the research community as a standard practice in empirically evaluating detection and recognition performance.

Acknowledgements

Gang Liu would like to thank Lixin Gong, Dr. Selim Aksoy and Dr. Terry Rockafellar for helpful discussions related to this topic. He would also like to thank Dr. Harry Schmitt and Mary Cassabaum of the Raytheon Company for providing the tank image sequence data. Anonymous reviewers' comments on earlier versions of this manuscript were extremely helpful in improving the quality of this paper.

A package of C source code for the software used in the experiments is available on the Internet at URL: (isl.ee.washington.edu/~gliu).

References

- [1] R.M. Haralick, Dialogue: performance characterization in computer vision, *CVGIP, Image Understand.* 60 (2) (1994) 245–249.
- [2] R.M. Haralick, Propagating covariance in computer vision, *Int. J. Pattern Recognition Artif. Intell.* 10 (5) (1996) 561–572.
- [3] V. Ramesh, Performance characterization of image understanding algorithms, Ph.D. Thesis, University of Washington, 1995.
- [4] S. Borra, S. Sarkar, A framework for performance characterization of intermediate-level grouping modules, *IEEE Trans. Pattern Anal. Mach. Intell.* 19 (11) (1997) 1306–1312.
- [5] K. Cho, P. Meer, J. Cabrera, Performance assessment through bootstrap, *IEEE Trans. Pattern Anal. Mach. Intell.* 19 (11) (1997) 1185–1198.
- [6] M.D. Health, S. Sarkar, T. Sanocki, K.W. Bowyer, A robust visual method for assessing the relative performance of edge-detection algorithms, *IEEE Trans. Pattern Anal. Mach. Intell.* 19 (12) (1997) 1338–1359.
- [7] K. Bowyer, C. Kranenburg, S. Dougherty, Edge detector evaluation using empirical ROC curves, in: *CVPR'99 Proceedings*, 1999, pp. 354–359.
- [8] M.C. Shin, D. Goldgof, K. Bowyer, Comparison of edge detectors using an object recognition task, in: *CVPR'99 Proceedings*, 1999, pp. 360–365.
- [9] L. Kitchen, A. Rosenfeld, Edge evaluation using local edge coherence, *IEEE Trans. Systems, Man, Cybernet.* 11 (9) (1981) 597–605.
- [10] V. Ramesh, R.M. Haralick, An integrated gradient edge detector—theory and performance evaluation, in: *Image Understanding Workshop Proceedings*, 1994, pp. 689–695.
- [11] T. Kanungo, M.Y. Jaisimha, J. Palmer, R.M. Haralick, A methodology for quantitative performance evaluation of detection algorithms, *IEEE Trans. Image Process.* 4 (12) (1995) 1667–1674.
- [12] G. Liu, R.M. Haralick, Assignment problem in edge detection performance evaluation, in: *CVPR 2000 Proceedings*, 2000, pp. 26–31.
- [13] J.F. Canny, A computational approach to edge detection, *IEEE Trans. Pattern Anal. Mach. Intell.* 8 (6) (1986) 679–698.
- [14] R.M. Haralick, Digital step edges from zero crossing of second directional derivatives, *IEEE Trans. Pattern Anal. Mach. Intell.* 6 (1) (1984) 58–68.
- [15] V.S. Nalwa, T.O. Binford, On detecting edges, *IEEE Trans. Pattern Anal. Mach. Intell.* 8 (6) (1986) 699–714.
- [16] S.M. Smith, J.M. Brady, SUSAN—a new approach to low level image processing, *Int. J. Comput. Vision* 23 (1) (1997) 45–78.
- [17] B. Bhanu, Automatic target recognition: State of the art survey, *IEEE Trans. Aerospace Electron. Systems AES-22* (4) (1986) 364–379.
- [18] G. Liu, R.M. Haralick, FLIR ATR using location uncertainty, *J. Electron. Imaging* 9 (2) (2000) 178–193.
- [19] R.M. Haralick, Overview: computer vision performance characterization, in: *Image Understanding Workshop Proceedings*, 1994, pp. 663–666.
- [20] R.M. Haralick, Performance characterization protocol in computer vision, in: *Image Understanding Workshop Proceedings*, 1994, pp. 667–673.

- [21] S. Yi, R.M. Haralick, L.G. Shapiro, Error propagation in machine vision, *Mach. Vision Appl.* 7 (1994) 93–114.
- [22] G. Liu, F. Zhuge, L. Gong, R.M. Haralick, Performance evaluation of the CFAR vehicle detection algorithm, in: *Image Understanding Workshop Proceedings*, 1998.
- [23] P. Burlina, V. Parameswaran, R. Chellappa, Sensitivity analysis and learning strategies for context-based vehicle detection algorithms, in: *Image Understanding Workshop Proceedings*, 1997, pp. 577–583.
- [24] D. Cyganski, B. King, R.F. Vaz, J.A. Orr, ROC analysis of ATR from SAR images using a model-based recognizer incorporating pose information, in: *Algorithms for Synthetic Aperture Radar Imagery II*, *Proceedings of the SPIE*, Vol. 2487, 1995, pp. 190–200.
- [25] B.L. O’Kane, T. Williams, R. Wright, Target-identification performance with enhancements of second-generation FLIR imagery, in: *1994 SID International Symposium Digest of Technical Papers*, San Jose, CA, 1994, pp. 114–117.
- [26] O. Young, M. Cook, Analytical modeling for automatic target recognition performance evaluation, in: *Signal and Image Processing Systems Performance Evaluation*, *Proceedings of the SPIE*, Vol. 1310, 1990, pp. 162–173.
- [27] J.A. Shufelt, Performance evaluation and analysis of monocular building extraction from aerial imagery, *IEEE Trans. Pattern Anal. Mach. Intell.* 21 (4) (1999) 311–326.
- [28] S.Z. Der, R. Chellappa, Probe-based automatic target recognition in infrared imagery, *IEEE Trans. Image Process.* 6 (1) (1997) 92–102.
- [29] M.R. Stevens, J.R. Beveridge, Precise matching of 3-d target models to multisensor data, *IEEE Trans. Image Process.* 6 (1) (1997) 126–142.
- [30] J.H. Kitrosser, An evaluation of a region-growing algorithm performing detection for automatic target recognition, *J. Imaging Sci. Technol.* 38 (4) (1994) 311–320.
- [31] B. Bhanu, R.D. Holben, Model-based segmentation of FLIR images, *IEEE Trans. Aerospace Electron. Systems AES-26* (1) (1990) 2–11.
- [32] Q.H. Pham, T.M. Brosnan, M.J.T. Smith, Sequential digital filters for fast detection of targets in FLIR image data, in: *Automatic Target Recognition VII*, *Proceedings of the SPIE*, Vol. 3069, 1997, pp. 62–73.
- [33] R.M. Matchko, G.R. Gerhart, Target detection for low ambient backgrounds, *Opt. Eng.* 37 (7) (1998) 1945–1950.
- [34] T. Peli et al., Morphology-based algorithms for target detection/segmentation in FLIR imagery, in: *Proceedings of the SPIE*, Vol. 1957, 1993, pp. 85–94.
- [35] J. Boriotti, F. Maali, Automatic thresholding for detection of landing aircraft in FLIR imagery, in: *Proceedings of the SPIE*, Vol. 1957, 1993, pp. 156–159.
- [36] M. Dutkiewicz et al., Fusion of optical, infrared and radar imagery for coastal surveillance, in: *Proceedings of the SPIE*, Vol. 2744, 1996, pp. 165–175.
- [37] D. de Ridder et al., Vehicle recognition in infrared images using shared weight neural networks, *Opt. Eng.* 37 (3) (1998) 847–857.
- [38] R.T. Rockafellar, *Network Flows and Monotropic Optimization*, Wiley, New York, 1984.
- [39] D.E. Knuth, *The Stanford GraphBase: A Platform for Combinatorial Computing*, Addison-Wesley, Reading, MA, 1993.
- [40] H.W. Kuhn, The Hungarian method for the assignment problem, *Naval Res. Logist. Quart.* 2 (1955) 83–97.
- [41] D. König, Graphok és matrixok, *Mat. és Fiz. Lapok* 38 (1931) 116–119.
- [42] J. Egerváry, Matrixok kombinatoricus tulajonságairól, *Mat. és Fiz. Lapok* 38 (1931) 16–28 (translation by H.W. Kuhn, “On combinatorial properties of matrices”, *George Washington University Logistics Papers* 11, 1955).
- [43] D.G. Lowe, Three-dimensional object recognition from single two-dimensional images, *Artif. Intell.* 31 (3) (1987) 355–395.
- [44] Q. Ji, R.M. Haralick, Breakpoint detection using covariance propagation, *IEEE Trans. Pattern Anal. Meth. Intell.* 20 (8) (1998) 845–851.
- [45] G. Liu, R.M. Haralick, Corner detection using location uncertainty, *Tech. Rep.*, Intelligent Systems Laboratory, Department of Electrical Engineering, University of Washington, Seattle, WA 98195, 2000.
- [46] D. Huttenlocher, G. Klanderma, W. Rucklidge, Comparing images using the Hausdorff distance, *IEEE Trans. Pattern Anal. Meth. Intell.* 15 (9) (1993) 850–863.
- [47] G.K. Kanji, *100 statistical Tests*, Sage publications Ltd., Beverley Hills, CA, 1993.
- [48] G. Liu, Automatic target recognition using location uncertainty, *Ph.D. Thesis*, University of Washington, 2000.

About the Author—GANG LIU received his B.Sc., M.Sc. and Ph.D. degrees, all in Electrical Engineering, in 1992, 1995 and 2000, from Xidian University in Xi’an, China, Tsinghua University in Beijing, China, and the University of Washington in Seattle, WA, USA, respectively. Upon receiving his doctorate, Dr. Liu joined the Cognex Corporation in Natick, MA, USA and engaged in industrial machine vision algorithm research and development. His research interests in computer vision include low-level feature extraction (including image gradient estimation, edge and line detection, corner detection and image segmentation), object representation, automatic target recognition, pattern classification and classifier combination, correspondence problem, OCR and bar code applications, and algorithm performance evaluation.

About the Author—DR. HARALICK is a distinguished professor in the Department of Computer Science, Graduate Center, City University of New York. Professor Haralick is responsible for developing the gray scale co-occurrence texture analysis technique and the facet model technique for image processing. In high level computer vision he has worked on robust methods for photogrammetry and developed fast algorithms for solving the consistent labeling problem. He has developed shape analysis and extraction techniques using mathematical morphology, the morphological sampling theorem, and fast recursive morphology algorithms. In the area of document image understanding, Professor Haralick, along with Professor Ihsin Phillips, developed a comprehensive ground-truthed set of some 1600 document image pages

most in English and some 200 pages in Japanese. He has also developed algorithms for document image skew angle estimation, zone delineation, word and text line bounding box delineation. His most recent research is in the area of computer vision performance characterization and covariance propagation. Professor Haralick is a Fellow of IEEE for his contributions in computer vision and image processing and a Fellow of IAPR for his contributions in pattern recognition, image processing, and for service to IAPR. He has published over 550 papers and recently completed his term as the president of the International Association for Pattern Recognition.