

Assignment Problem in Edge Detection Performance Evaluation

Gang Liu and Robert M. Haralick
Intelligent Systems Laboratory
Department of Electrical Engineering
University of Washington
Seattle, WA 98195-2500, USA
{gliu,haralick}@isl.ee.washington.edu

Abstract

We propose to use the combinatorial assignment problem to model the issue of associating ground-truth and declared edge pixels in the objective empirical performance evaluation of edge detectors. The assignment problem is adapted to the maximal assignment problem to incorporate the need for tolerating certain amount of localization error for the detected ground-truth pixels. The solution to this problem yields a maximal one-to-one association between ground-truth and declared edge pixels. Performance evaluation based on this association has the attitude of making the most positive interpretation of the declared edge map. Synthetic test data is used in the experiment to allow unambiguous subjective judgement of edge detection performance. The preciseness and reasonableness of the performance evaluation from the proposed method is observed. The usefulness of this method in other performance evaluation applications is also discussed.

1. Introduction

Performance evaluation in computer vision is an important and active research area [4, 5, 10, 1, 6, 7]. Both theoretical and empirical approaches are being taken, and both have their strength and weaknesses. In empirical performance evaluation, one major approach is to get ground-truth for the image data and compute the objective performance measure by comparing the algorithm output with the ground-truth.

In assessing the performance of edge detectors [2, 3, 9, 12], we care about the detection rate, false alarm rate and the average localization error for the correctly detected edge pixels. A performance evaluation technique needs to classify ground-truth edge pixels into two distinct classes, namely detected and misdetected edge pixels, and to classify edge pixels in the declared edge map into two distinct classes, namely correct declarations and false alarms. To

tolerate certain amount of localization error, ground-truth and declared edge pixels do not have to be at exactly the same pixel location for them to be declared as detected ground-truth edge pixels and correct declarations, respectively. However, each detected ground-truth edge pixels 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 of the ground-truth and declared pixels are not trivial problems. Different ways of doing these leads to different performance measures.

A distance transform based technique has been proposed 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 which have a distance value not larger than the threshold. The detected ground-truth pixels are those which have declared edge pixels within their neighborhood, which 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. For example, consider the example shown in Figure 1, where ground-truth edge pixels are marked by circles, and declared edge pixels are marked by solid squares. According to distance transform based method, all ground-truth pixels are detected and there is no false alarm declarations.

Intuitively, one would want to have an exact one-to-one correspondence between the detected ground-truth edge pixels and the correct declarations. If this is established, the numbers of detected and misdetected ground-truth edge pixels and the numbers of correct declarations and false alarms will be more informative of the edge detection performance. According to this consideration, only two ground-truth pixels, the one in the middle on the left half and the one on the right are regarded as being detected, and

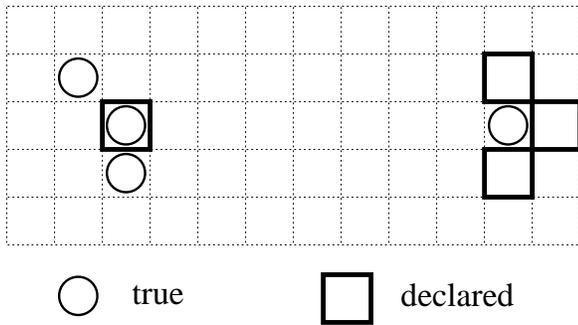


Figure 1. An example where distance transform based method fail to give accurate performance measures.

the other two on the left half are regarded as being misdetected. Similarly, only two of the declared pixels, the one on the left half and one of the three on the right (which one does not matter in this example), are regarded as correct declarations. The remaining two on the right half are regarded as false alarm declarations. Therefore, the more appropriate performance evaluation method should declare two misdetections and two false alarms.

Here we propose to establish the desired one-to-one correspondence by solving a combinatorial maximal assignment problem. The solution yields such a correspondence between the ground-truth and declared edge pixels. Performance measures based on this correspondence follow the intuition and are more informative of the edge detection performance.

In our experiments, 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 performance evaluation techniques. This is because it allows a clear and unambiguous judgement of edge detection quality with minimal inter- and intra-observer variation. This judgement acts as a standard to which the output of different performance evaluation techniques should be compared. The best performance evaluation technique is the one whose output is closest to that judgement.

2. Assignment problem

2.1. Unconstrained case

In the unconstrained assignment problem, one is concerned with establishing a full one-to-one correspondence between two sets K and L , both of which have N elements.

An assignment is a 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 optimal assignment is a mapping $a : K \rightarrow L$ which yields the smallest cost. The assignment problem is to find such an optimal assignment.

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

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 the distance between a ground-truth edge pixel k and a declared edge pixel l .

Most likely, the numbers of ground-truth edge pixels and declared edge pixels are not equal. This does not cause a problem. Conceptually, we can think of adding “ghost” pixels to the set which has smaller number of elements. The distance from any ghost pixel to all pixels in the other set is assigned a finite but very large value. This way, the ghost pixels will not compete with any of the original pixels in the assignment. In the end, all ghost pixels will also be assigned to pixels in the other set. All those pixels that are the counterparts of the ghost pixels are considered unmatched, and therefore are either misdetected ground-truth edge pixels (if the ghost pixels are added to the declared edge pixel set) or false alarms in the declared edge map (if the ghost pixels are added to the ground-truth edge pixel set.)

2.2. Hungarian algorithm

The unconstrained optimal assignment problem is essentially determined by the cost function $q(\cdot, \cdot)$. Exhaustive search for the optimal assignment would require $O(N)$ computations of the total cost in Equation (1). Fortunately, this need not be done.

Let the cost function be represented in a matrix form. Adding a constant to any row of the matrix does not change the the optimal assignment. (Although the total cost of the resulting optimal assignment gets changed, it is the 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 solution, either. The Hungarian algorithm [11, 8] works by adding proper constants to properly chosen rows and columns of the matrix, and finding the maximum number of independent zeros in all equivalent forms of the cost matrix. Due to the König-Egerváry Theorem, this is equivalent to finding the smallest number of lines (rows and columns) to cover all zeros.

In the situation where every element in K is allowed to be paired with every element in L , the optimal solution always exists. 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 [8] of the algorithm handles the integral cost matrix. The number of elements in K and L need not be equal. The computational complexity is $O(m^2n)$ where m is the smaller of the cardinalities $|K|$ and $|L|$, and n is the larger.

2.3. Constrained case

In edge detection performance evaluation, we know that corresponding ground-truth edge pixels and declared edge pixels cannot be spatially very far apart. The threshold on the distance between them can be conveniently set by the neighborhood size used in the edge detection. We want to put a constraint on the solution of the optimal assignment problem:

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

The constrained optimal assignment problem is then to find a mapping $a : K \rightarrow L$ which minimizes the cost in Equation (1) and satisfies the constraint in Equation (2). Now the finiteness of $q(k, l)$ does not guarantee a solution to the constrained problem.

The constrained problem in its original form is not an appropriate model for our problem at hand. Instead, we consider another model which we call the *maximal assignment problem*.

For any particular assignment $a : K \rightarrow L$, let K_s be a subset of K such that the constraint is satisfied for all its members.

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

Let $K_m(a)$ be the largest among all such subsets.

$$K_m(a) = \arg \max_{\{K_s \subseteq K | q(k, a(k)) \leq \tau, \forall k \in K_s\}} |K_s| \quad (3)$$

where $|K_s|$ is the cardinality (number of elements) of K_s . $K_m(a)$ is called a *maximal subset* of K w.r.t. the assignment a .

The rank of a is defined as the cardinality of its corresponding maximal subset

$$r(a) = |K_m(a)| \quad (4)$$

In edge detection evaluation, we are only interested in the largest ranking assignments, which are called the *maximal assignments*. The cost for a maximal assignment a is defined by

$$\sum_{k \in K_m(a)} q(k, a(k)) \quad (5)$$

The maximal assignment problem is to minimize this cost over all maximal assignments.

In the edge detection evaluation application, the solution to the maximal assignment problem gives the largest possible number of pairs of ground-truth and declared edge pixels. Among all the different ways for making this kind of match, it picks the one with the smallest localization error. In a sense, it tries 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 need to try their best to make the most positive use of it.

2.4. Solving the maximal assignment problem

We now apply the idea of “ghost pair” to transform the maximal assignment problem back to the unconstrained assignment problem, solve the unconstrained problem, and apply some simple post-processing to obtain the solution to the maximal assignment problem.

Let d be a finite and very large value. For example,

$$d = N\tau \quad (6)$$

We selectively modify the cost by

$$q(k, l) = \begin{cases} d & \text{if } q(k, l) > \tau \\ q(k, l) & \text{otherwise} \end{cases} \quad (7)$$

This modified cost function is used to form an unconstrained problem. Each pair of elements whose cost gets changed to d is called a “ghost pair.” Since d is such a large value, this particular pair does not compete for each other against other elements in the assignment process. Notice that, normally each ground-truth and declared edge pixel is involved in many ghost pairs since there almost always exist edge pixels in the other class that are faraway from it. However, detected ground-truth edge pixels and correctly declared edge pixels are also involved in some pairs that are not ghost pairs. In the association process, it is these non-ghost pairs that are of interest to us. However, the provision for the ghost pairs is needed to change the hard constraint on the pixel eligibility into penalty terms in the total cost. This is necessary for the Hungarian algorithm to be used to find the solution.

The optimal solution to this new unconstrained problem exists due to the finiteness of the cost function. Hence it can be found by the Hungarian algorithm. In the resulting solution, we examine again the cost between the assigned pairs, to enforce the constraint on the pixel eligibility for being paired up. The pairs whose cost are not larger than τ are good associations. They give correspondence between the involved detected ground-truth edge pixels and correct declarations. The cost of each of the rest of the pairs is larger than τ , and these must all be ghost pairs.

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 ghost pair are then determined to be a misdetected ground-truth edge pixel and a false alarm in the declared edge map.

3. Association procedure

It is now clear that the association of ground-truth edge pixels with the declared edge pixels can be obtained with the procedure described below. The result of the association also determines the misdetected ground-truth edge pixels and the false alarms in the declared edge map.

Note that although the original formulation assumes the numbers of elements in K and L are equal, the algorithm can handle unequal cases as well [8]. Therefore, we do not need to make them the same size by adding ghost elements. Let N_K and N_L denote the number of elements in K and L , respectively.

Edge detection result classification and association procedure:

1. Resolving the simple cases:
 - The ground-truth edge pixels which have declared edge pixels right at the same locations 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 ground-truth edge pixels which do not have any declared edge pixels within a distance of the chosen threshold τ are automatically determined as misdetected ground-truth edge pixels.
 - Similar rules apply for the declared edge pixels to determine the automatic correct declarations and false alarms.

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

2. Let K be the set of ground-truth edge pixel locations, L be the set of declared edge pixel locations. Create an $N_K \times N_L$ matrix Q of distance values between each pair in $K \times L$.
3. Selectively modify Q by applying the rule in Equation (7), with N_K as N in Equation (6).
4. Apply the Hungarian algorithm on the modified Q .

5. Examine the resulting assignment. (Recompute the distance for the assigned pairs if Q is destroyed by the implementation of the Hungarian algorithm.)
 - 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 $N_K < N_L$).

4. Experiment

We use the synthetic test image used in [6] to compare the proposed method with a distance transform based method. This test image is 64×64 , and has a brighter disk of constant gray value against a darker constant background. White Gaussian noise is added to obtain a noisy version of the image with SNR=4. Figure 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 [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.

We apply the implementation of Canny's edge detector used in [6] to the noisy image with different tuning parameters. The tuning parameters of (1.05,0.77,0.88) given in [6] produces the edge map shown in Figure 2(b). Another two sets of parameters are chosen to give more obvious misdetections and false alarms, respectively. These are also shown in Figure 2.

The proposed assignment based method and the distance transform based method are used to compare the declared edge maps with the ground-truth edge map. The distance transform based method used in [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 threshold τ in the assignment model is set accordingly to $2\sqrt{2}$. The performance measures of number of misdetections (#MD), number of false alarms (#FA),

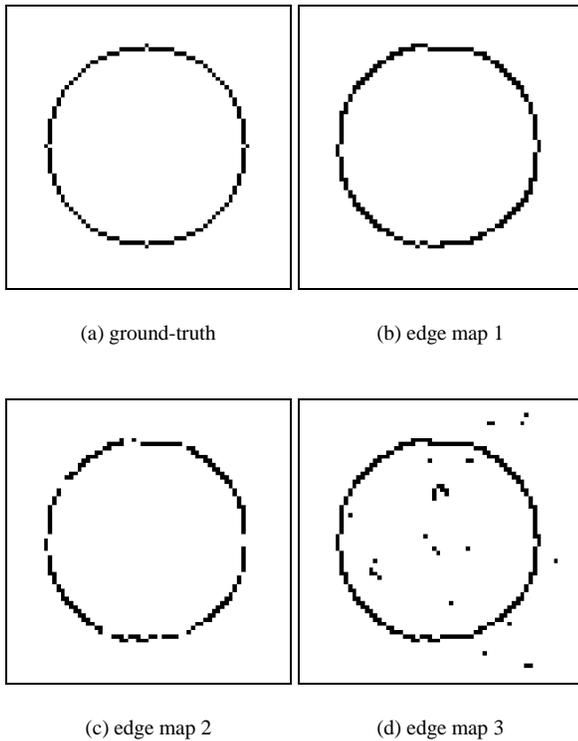


Figure 2. Edge maps on which the performance measures are calculated.

and the root-mean-squared (RMS) localization error for the detected ground-truth edge pixels from these two methods are given in Table 1.

It should be noted that the purpose of the experiment here is to demonstrate the appropriateness of using the assignment model 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 was made to tune the detector to 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, these thicker parts of the declared edge (where edge pixels have more than two 8-neighbors) contains false alarms. The proposed method precisely picks out some declared pixels as false alarms,

while the distance transform based method overly 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 assignment based 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 tend to point out some false alarms on the thicker parts of the declared edges. Both of these are reflected in the evaluation result by the assignment based method. The distance transform based method is overly lenient again.

In edge map 3, the assignment based method treats not only the stray edge pixels as false alarms, but also some on the thicker parts of the edges around the ground-truth edges. 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 of being overly reluctant to declare false alarms around ground-truth edge locations, and overly reluctant to declare misdetections around any declared edge pixels. It gives inappropriately high performance measures. The assignment based method is more precise, and gives performance measures conforming to the intuition and the subjective evaluation.

5. Usefulness in other applications

The proposed methodology is useful in many applications where a maximal one-to-one correspondence is to be established between two sets, where 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 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 so that the main idea of the proposed methodology can be described more clearly without much potential confusion from other unrelated application-specific issues.

The proposed methodology is readily applicable to performance evaluation in applications where the classification as well as the localization performance are of interest. Examples include vehicle detection in aerial images where the misdetections and false alarm vehicles need to be counted, and automatic target recognition (ATR) applications where the detection rate, false alarm rate and the recognition rate for the correctly detected targets need to be computed to

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

edge map	assignment			distance transform		
	# MD	# FA	RMS loc error	# MD	# FA	RMS loc err
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

characterize the system performance.

For applications such as ATR which involve both detection and recognition, a two-stage assignment based 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 potentially 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 total number of pairs produced by both 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 misdetections 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 the system.

6. Summary

We have presented a methodology for most reasonably associating the ground-truth edge pixels with the edge pixels declared by edge detection algorithms. The central idea is to identify the association problem as a maximal assignment problem. The solution to this problem establishes a one-to-one correspondence between detected ground-truth edge pixels and the correct declarations in the edge detector output. The determination of misdetections ground-truth pixels and false alarms in the declaration as well as the computation of the localization error for the detected ground-truth edge pixels are made according to the established correspondence. We showed examples where the appropriateness of this technique was verified. The usefulness of this methodology in the performance evaluation of other detection and recognition applications was discussed.

Acknowledgment and software

Gang Liu would like to thank Lixin Gong and Selim Aksoy for many helpful discussions related to this topic. The reviewers' comments are also very helpful in improving the quality of this paper.

The C source code package for the software used in the experiment is available on the WWW at <http://isl.ee.washington.edu/~gliu>.

References

- [1] S. Borra and S. Sarkar. A framework for performance characterization of intermediate-level grouping modules. *IEEE Trans. on PAMI*, 19(11):1306–12, Nov 1997.
- [2] J. F. Canny. A computational approach to edge detection. *IEEE Trans. on PAMI*, 8(6):679–98, Nov 1986.
- [3] R. M. Haralick. Digital step edges from zero crossing of second directional derivatives. *IEEE Trans. on PAMI*, 6(1):58–68, Jan 1984.
- [4] R. M. Haralick. Dialogue: Performance characterization in computer vision. *CVGIP: Image Understanding*, 60(2):245–9, Sep 1994.
- [5] R. M. Haralick. Propagating covariance in computer vision. *IJPRAI*, 10(5):561–572, Aug 1996.
- [6] M. D. Health *et al.* A robust visual method for assessing the relative performance of edge-detection algorithms. *IEEE Trans. on PAMI*, 19(12):1338–59, Dec 1997.
- [7] L. Kitchen and A. Rosenfeld. Edge evaluation using local edge coherence. *IEEE Trans. on SMC*, 11(9):597–605, Sep 1981.
- [8] D. E. Knuth. *The Stanford GraphBase: a platform for combinatorial computing*. Addison-Wesley, 1993.
- [9] V. S. Nalwa and T. O. Binford. On detecting edges. *IEEE Trans. on PAMI*, 8(6):699–714, Nov 1986.
- [10] V. Ramesh. *Performance characterization of image understanding algorithms*. PhD thesis, University of Washington, 1995.
- [11] R. T. Rockafellar. *Network flows and monotropic optimization*. Wiley, 1984.
- [12] S. M. Smith and J. M. Brady. SUSAN — a new approach to low level image processing. *IJCV*, 23(1):45–78, 1997.