Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 150-156, 1998.

A Probabilistic Formulation for Hausdorff Matching

Clark F. Olson
Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove Drive, Mail Stop 107-102, Pasadena, CA 91109-8001
http://robotics.jpl.nasa.gov/people/olson/homepage.html

Abstract

Matching images based on a Hausdorff measure has become popular for computer vision applications. However, no probabilistic model has been used in these applications. This limits the formal treatment of several issues, such as feature uncertainties and prior knowledge. In this paper, we develop a probabilistic formulation of image matching in terms of maximum likelihood estimation that generalizes a version of Hausdorff matching. This formulation yields several benefits with respect to previous Hausdorff matching formulations. In addition, we show that the optimal model position in a discretized pose space can be located efficiently in this formation and we apply these techniques to a mobile robot self-localization problem.

1 Introduction

The use of variants of the Hausdorff distance has recently become popular for image matching applications (see, for example, [6, 9, 11, 16, 18, 19]). While these methods have been largely successful, they have lacked a probabilistic formulation of the matching process and this has made it difficult to incorporate probabilistic information, such as feature uncertainties and the prior probability of model positions, into these applications. This work addresses these issues by introducing a probabilistic formulation of image matching that generalizes a version of Hausdorff matching.

After a brief review of Hausdorff matching techniques, we describe a probabilistic formulation of image matching based on the principal of maximum likelihood estimation. In this formulation, we seek local maxima of the likelihood function over the possible model positions. While this formulation implicitly assumes that the model appears exactly once in the image, it can be applied even when the model does not appear in the image or appears multiple times. We must simply set the criterion determining which model positions are reported as likely hypotheses appropriately. When a particular probability density function (PDF) is introduced for the distance from each model feature to the closest matching image feature, this formulation yields a conventional Hausdorff matching method. Alternate PDF's yield new and interesting image matching measures.

This probabilistic formulation of image matching yields several benefits. It allows the incorporation of prior knowledge, such as the prior probability of model positions, into the matching process. It also allows formal treatment of feature uncertainties in the search for likely model positions. Perhaps most importantly, we can consider arbitrary probability distributions for the matching error between model and image features. This allows us to eliminate the sharp distinction between matched and unmatched model features that is inherent to previous Hausdorff matching methods.

We discuss techniques for efficiently searching the pose space in this formulation and give experimental evidence that indicates that we achieve improved accuracy in the recognition and localization of objects in images. Finally, we apply these techniques to a mobile robot self-localization application that performs matching between terrain occupancy maps to determine the robot's position.

2 Hausdorff matching

This section reviews a variation of the Hausdorff distance used to perform image matching, as well as the application of this measure to matching binary images and an efficient search strategy for finding the relative image positions where the measure meets some criterion.

2.1 Hausdorff measure

For two sets of points A and B, the directed Hausdorff distance from A to B is:

$$h(A, B) = \max_{a \in A} \min_{b \in B} ||a - b|| , \tag{1}$$

where $||\cdot||$ is any norm. This yields the maximum distance from a point in set A to its nearest point in set B.

Notice, however, that a single outlier in A can change this distance by an arbitrary amount. For image matching, where A is usually a set of model points and B is a set of image points, we wish to allow outliers (which correspond to occluded or undetected model features in this case). It is thus common to use the partial distance [5]:

$$h_K(A, B) = \underset{a \in A}{K^{\text{th}}} \min_{b \in B} ||a - b|| \qquad (2)$$

This yields the Hausdorff distance among the K points in A that best match points in B (and thus allows $|A|-K$ outliers in the set A). This measure is asymmetric, since it does not consider how well each of the points in B is fit by A. Matching can thus be performed against a large image that contains the model image as a subset.

A variation on the partial Hausdorff distance is to determine the maximum number of points in the model such that the distance is below a given error threshold:

$$h_K(A, B) \leq \delta \qquad (3)$$

Let $K_\delta(A, B)$ denote the maximum K for which (3) is true. The ratio $F_\delta(A, B) = \frac{K_\delta(A,B)}{|A|}$ is called the *Hausdorff fraction*, since it is the fraction of the points in A that match a point in B up to the error δ. This formulation is easy to work with, since $K_\delta(A, B)$ is simple to compute, and we examine this variation of Hausdorff matching in this paper.

Note that pre-setting the maximum allowable error δ and determining the model positions such that F_δ is above some threshold T yields equivalent results to setting the model fraction to T and determining the model positions with partial Hausdorff distance no greater than δ. This formulation of the Hausdorff measure does not change the solutions that are found.

It is worth noting that maximizing the Hausdorff fraction subject to a constant error threshold is essentially the same as performing object recognition using a bounded error criterion [1, 2], where the best model positions are those that match the most model features up to some bounded error. Much of this discussion applies equally well to this work. However, we concentrate on matching using a search strategy associated with the Hausdorff distance [8, 12, 15].

2.2 Application to binary images

We are concerned with the application of these techniques to binary images (e.g. image edge maps). Each pixel in such an image takes a value of 0 or 1. We say that the pixels with a value of 1 are *occupied* and those with a value of 0 are *unoccupied*.

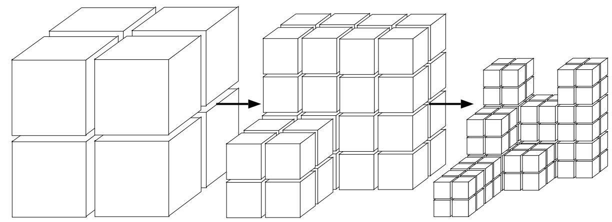


Figure 1: A search strategy is used that recursively divides and prunes cells of the search space.

Let M be a model image or template and I be an image that may contain an instance of the model. Both M and I can be considered to be discrete sets of points corresponding to the locations of the occupied pixels in the image or template. Let t be a particular position of the model with respect to the image. This model position can be thought of a function that maps the model points into the image; $t(M)$ is thus the set of model points after mapping them according to t.

Now, consider the dilation of the image by a structuring element S_δ that consists of all of the pixels within δ of the origin with respect to some norm. The dilated image, $I_\delta = I \oplus S_\delta$ (where $\oplus$ denotes the Minkowski sum or morphological dilation operator), has an occupied pixel at each location that is within δ of an occupied pixel in the original image. Let $I_\delta(m)$ denote the value of I_δ (i.e. 0 or 1) at the position of some model pixel, m. We can write the Hausdorff fraction (as a function of the model position) as follows:

$$F_\delta(t(M), I) = \frac{1}{|M|} \sum_{m \in M} I_\delta(t(m)) \qquad (4)$$

2.3 Efficient search strategy

An efficient search strategy for locating model positions that satisfy some criterion with respect to the Hausdorff fraction can be formulated using a multi-resolution search that examines a hierarchical cell decomposition of the space of possible model positions [8, 12, 15]. This method divides the space of model positions into rectilinear cells and determines which cells may contain a position satisfying the criterion using some test. The cells that pass the test are divided into subcells, which are examined recursively. The rest are pruned. See Figure 1.

The key to this method of searching the parameter space is a quick method to conservatively test whether a cell can contain a position satisfying the criterion. This test is allowed to fail to rule out a cell that does not contain any positions satisfying the criterion, but it should never rule out a cell that does contain such a position, since this may result in missing a valid model position. It is typical in this method to consider only the model positions in some underlying discretization of the pose space. When a cell of this space is reached that contains a single position in the discretization, this position is tested explicitly.

In order to develop an efficient testing mechanism for determining whether a cell can be pruned, it is useful to consider the distance transform of the image. For a binary image $I(x, y)$, the distance transform $\mathcal{D}_I(x, y)$ measures the distance from each pixel in the image to the closest occupied pixel [14].

To test a cell C of possible model positions, the discrete pose c closest to the center of the cell is first determined. The maximum distance between the location to which a model pixel is mapped into the image by c and by any other pose in the cell is then computed. We call this distance the *image-mapped radius* of the cell and denote it Δ_C:

$$\Delta_C = \max_{p \in C} \max_{m \in M} ||p(m) - c(m)||$$

Now, if we seek positions at which $K_\delta(t(M), I)$ is no less than T, then, to test the cell, we count the number of model points for which the distance transform at the appropriate location is no larger than $\delta + \Delta_C$. If this number is less than T, then we can prune the cell, since it cannot contain a model position that matches T pixels in the model to pixels in the image up to the error δ.

When a cell cannot be pruned, it is divided into multiple subcells, and the procedure is applied recursively to each of the subcells. This process continues until all of the cells in the pose space have been exhausted.

3 Probabilistic formulation

We now describe a probabilistic formulation of image matching based on the principal of maximum likelihood estimation that retains the flavor of Hausdorff matching. To formalize the problem, let us say that we have a set of model features, $M = \{\mu_1, ..., \mu_m\}$ and a set of image features, $I = \{\nu_1, ..., \nu_n\}$. Let $t \in T$ be a random variable describing the position of the model in the image. This makes an implicit assumption that exactly one instance of the model appears in the image. However, we shall see that cases where the model does not appear, or that the model appears in multiple instances, can be easily handled in this formulation.

To formulate the problem in terms of maximum likelihood estimation of the model position, we must have some set of measurements that are a function of the position of the model. We use the distance from each model pixel (at the position specified by t) to the closest occupied pixel in the image as our set of measurements. Call these distances $D_1, ..., D_m$. Each of these distances can be found by looking up the position of the model pixel in the distance transform of the image. Recent work on determining the probability of a false positive for Hausdorff matching [3, 12] has achieved accurate results by treating the model features independently. We thus treat the distances, $D_1, ..., D_m$, as being independent.

The likelihood function for t can now be formulated as the product of the probabilities of these distances:

$$L(t) = \prod_{i=1}^{m} p(D_i; t), \tag{5}$$

where $p(D_i; t)$ is the probability density function (PDF) of D_i evaluated at the model position t. Taking the logarithm of (5) yields a measure that preserves the ordering of the model positions:

$$\ln L(t) = \sum_{i=1}^{m} \ln p(D_i; t) \tag{6}$$

Any PDF, $p(D_i; t)$, can be used to yield a matching criterion in this formulation. Some useful PDFs are examined in the next section. Here we note that a measure equivalent to $K_\delta(t(M), I)$ can be obtained by using:

$$\ln p(D_i; t) = \begin{cases} k_1 + k_2, & \text{if } D_i \leq \delta \\ k_1, & \text{otherwise} \end{cases} \tag{7}$$

This probability density function is two-valued as in the conventional Hausdorff matching formulation. If there is support for the model feature in the image at this position (i.e. an image feature lies with δ of it), then some constant probability is assigned to $p(D_i; t)$ (a uniform distribution in the distance to nearest feature). Otherwise, some smaller constant probability is assigned to $p(D_i; t)$ (also uniform, but less likely). The precise values of k_1 and k_2 are unimportant in this equation (they do not change the ranking of model positions) as long as $k_2 > 0$. In practice, we use $k_1 = 0$ and $k_2 = 1$.

Now, let us address the implicit assumption in this formulation that the model appears exactly once in the image. If we are seeking models that may appear more than once in an image, or not at all, we must only set some threshold on (6), as is usually done in Hausdorff matching formulations. The model positions that surpass the threshold correspond to the likely positions of the model in the image.

4 Using the probabilistic formulation

This section explores some of the advantages that can be achieved through the use of the new matching formulation.

4.1 Prior probabilities of model positions

In some applications, we have prior knowledge of the likelihood of various model positions being correct. For example, in tracking applications (e.g. [7]) we may use the previous position of the object being tracked and its velocity to predict the next position of the object.

In the case where the prior probability of each model position is not uniform, let $p(t)$ be the prior probability of position t. We now have:

$$\ln L(t) = \ln p(t) + \sum_{i=1}^{m} \ln p(D_i; t) \tag{8}$$

It is relatively easy to incorporate this information into the efficient search strategy. For each cell, we must only determine the maximum prior probability of any position in the cell and add it to the score from the model features when determining whether to prune the cell.

The use of this prior information yields the additional benefit that we have bounds on the space that we need to search. We need not examine any position for which $\ln p(t)$ is small enough that the sum with the best possible score for each of the model pixels could not surpass the matching threshold.

4.2 Alternative PDFs

This probabilistic formulation of image matching allows the use of an arbitrary probability distribution function for the distance from each model feature to the nearest image feature, $p(D_i; t)$. It is common to model the feature localization error with a normal distribution. However, using a normal distribution in this framework yields a least-squares formulation, which results in poor robustness to outliers. A solution to this problem is to add a constant term to the normal distribution, yielding:

$$p(D_i; t) = k_1 + \frac{1}{\sigma\sqrt{2\pi}} e^{-{D_i}^2/2\sigma^2} \tag{9}$$

The constant term provides a lower bound on $\ln p(D_i; t)$, preventing it from becoming arbitrarily small for large D_i and thus allows unmatched model points with lower cost.

If we allow the probability density function to vary with the image location of the model feature (not just the distance to the nearest image feature), we can allow varying levels of uncertainty in the image features. For example, we may use a feature detector that yields uncertainty estimates for the position or the existence of the features. A feature that is less likely to exist, or for which the position estimate is inaccurate, may be weighted less in the matching process. For features with inaccurate position estimates, we may allow model features that are more distant to match them (i.e. a wider, flatter probability density function).

Note that the efficient search strategy discussed above does not work directly with an arbitrary PDF. We require some modification to the search strategy to perform matching with this formulation.

5 Relationship to MAP matching

The formulation of the image matching problem described here is similar to the MAP model matching (MMM) formulation of Wells [17]. In both formulations the best model positions are those that maximize a score from the prior distribution of model positions added to the sum of the scores for a set of features. The MMM formulation is based on a model with explicit feature correspondences. No such correspondences are necessary in our formulation, although implicit correspondences are given by the distance to the nearest image feature.

The primary difference between these methods is that the MMM formulation sums scores over the set of image features, while our formulation sums scores over the set of model features. Summing over the image features corresponds to a formulation where we try to best explain the image features. Multiple image features may contribute to the sum by matching the same model feature. Summing over the model features corresponds to a formulation where we try to best locate instances of the model. Multiple model features may contribute to the sum by matching the same image feature. We argue that summing over the model features is more appropriate for object recognition applications since many of the image features are irrelevant noise or background effects. In addition, the spacing of the model features can be controlled more easily than the spacing of image features. It is thus possible to ensure that no duplicate matches occur in some cases [4].

An additional difference is that Wells presents his work as a method to refine hypotheses that have been generated through indexing or some other hypothesis generation technique. He thus focuses on iterative optimization techniques for determining the best model positions, assuming that the starting point is not too far from the optimal position. We are more interested in global search techniques that can search a (discretized) pose space and guarantee that the best model position is found. It should be noted that these search techniques are complementary. We could generate the hypotheses using the global search techniques for refinement by the iterative optimization techniques. However, we have found that the global search techniques are usually of sufficient accuracy that refinement is not necessary.

6 Efficient algorithm

If we use the alternative probability density functions described in Section 4.2, we must modify the search strategy to efficiently find the model positions that satisfy the matching criterion.

Let us first note that a brute force method can be constructed by determining, for each pixel location in the image, the value of $\ln p(D_i;t)$, since $p(D_i;t)$ is independent of the particular model feature; only the position to which t maps μ_i into the image is important. We can thus compute a transform of the image, $P_I(X) = p(\mathcal{D}_I(X))$, where $X = [x\ y]^T$ is a pixel location in the image and $\mathcal{D}_I(X)$ is the distance transform of the image, according to Equation (9) or any other PDF. We call this the *probability transform* of the image. Each possible position of the model can be tested by examining this transform at the location that the position maps each model feature, summing the results, and determining if the sum meets the criterion.

Now, to search the space efficiently, we adapt the multi-resolution search strategy discussed previously, where we attempt to prune large cells of the transformation space. Recall that in this search strategy, we compute, for each cell that is examined, the discrete model position closest to the center of the cell and the image-mapped radius of the cell, called c and Δ_C, respectively. Then, each of the model features is tested to determine if there could be a position within the cell where the model feature is matched by an image feature up to the allowable error.

In the new formulation, we instead want to determine, for each model feature, the maximum value that $p(D_i;t)$ can achieve with respect to any model position in the cell. Call this value p_i^C. A bound can be placed on this value as follows:

$$p_i^C = \max_{t\in C} p(D_i;t) \leq p(\max(d_i^C, 0)) \tag{10}$$

$$d_i^C = \mathcal{D}_I(c(\mu_i)) - \Delta_C \tag{11}$$

Now, if we sum each p_i^C and the result does not satisfy the matching criterion, then we can prune the entire cell. The remainder of the search strategy remains the same.

If we allow the probability density function to vary with the image position to which the model feature is taken by t, this search becomes slightly more complicated. It is not necessarily sufficient to know the distance to the nearest image feature, since we may weight the image features differently. In this case, we parameterize the PDF by the image position, and it is equivalent to the probability transform of the image, $P_I(X)$. We can then bound the maximum likelihood that can be achieved for some model feature by any position in C as follows:

$$p_i^C = \max_{t\in C} P_I(t(\mu_i)) \leq \max_{Y\in\{c(\mu_i)\}\oplus S_{\Delta_C}} P_I(Y) \tag{12}$$

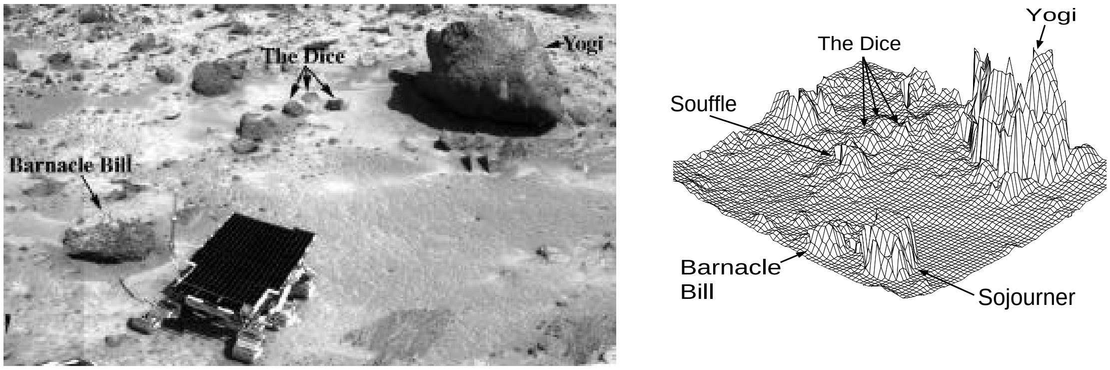
For some structuring elements, S_{Δ_C}, we can compute these bounds efficiently at each level of the cell hierarchy, if all of the cells at the level have the same dimensions.

7 Results

This section discusses the results of applying these techniques to both a synthetic problem, where we are concerned with matching two-dimensional image data, and a real application, where we localize a mobile robot by matching three-dimensional range maps.

7.1 Synthetic experiments

We first tested these techniques in controlled experiments where exact ground-truth was available, since the image feature data was generated synthetically.



(a) (b)

Figure 2: Terrain map generated from Mars Pathfinder imagery. (a) Annotated image of Sojourner and rocks on Mars. (b) Terrain map generated from stereo imagery.

We chose a simple problem domain (translation of isolated feature points) under demanding conditions to demonstrate the superiority of the probabilistic formulation. This experiment generated random model features (to subpixel accuracy). The model was translated randomly and placed in the image with considerable occlusion, clutter, and noise. We then performed a search for the model using both conventional Hausdorff matching techniques and the probabilistic formulation using a probability distribution similar to (9).

Over 10000 trials, the conventional Hausdorff matching method yielded 395 instances where an incorrect match had a higher score than the correct match, while the probabilistic formulation yielded 278 such failures on the same images. The probabilistic formulation thus yielded superior recognition of the feature patterns.

We also tested the localization accuracy of the techniques. Note that a lower bound on the average accuracy of matching of 0.25 pixels in each direction existed, since matching was performed only to pixel accuracy. In the successful trials, the probabilistic formulation yielded an average localization error of 0.46 pixels in each direction, while the conventional method yielded an average error of 0.58 pixels. The probabilistic formulation thus had superior performance in localizing objects as well.

7.2 Mobile robot localization

While the synthetic problem described above yields positive data with respect to the performance of the probabilistic formulation of Hausdorff matching, the real test, of course, is in real applications. We have previously implemented a mobile robot localization method using conventional Hausdorff matching methods [11]. Here we compare this system to a new implementation using the probabilistic formulation.

The motivation for studying this problem is to allow the next-generation Mars rover to have greater autonomy from the lander and from human operators. The basic method that is used is to generate a range map of the terrain near the robot through stereo vision [10]. This range map is transformed into a three-dimensional occupancy map describing the terrain (see Figure 2) and it is then compared against a previously generated occupancy map of the terrain to determine the relative position between the maps. For example, it can be compared to a map generated from previous robot positions, or to a map generated prior to the robot activity by some other means [13]. While the matching techniques described here have been discussed in terms of two-dimensional edge maps, the generalization to three-dimensional surface maps is straightforward.

In an experiment over 13 camera positions, where the ground-truth was measured by hand, the previous implementation using the conventional Hausdorff matching method had an average error of 0.050 meters, while the new implementation yielded an average error of 0.042 meters. It is likely that human error in collecting the ground-truth is responsible for a significant amount of the remaining error. In similar experiments where the cameras were panned by 25 degrees, but were not translated, the average error was reduced from 0.011 meters to 0.004 meters. The probabilistic formulation of Hausdorff matching thus yielded significantly improved results in this problem domain.

This method has also been applied to imagery from the Mars Pathfinder mission [13]. While there is no ground-truth for this data, the autonomous localization results that have been computed are close to the results generated by a human operator.

8 Summary

The primary contribution of this paper is a new formulation of image matching in terms of maximum likelihood estimation. This formulation seeks local maxima in the likelihood function of position of the model with respect to the image. While this implicitly assumes that the model appears in the image, this formulation can be applied equally well when the model does not appear in the image if an appropriate threshold is used to determine which locations are output as likely model positions.

This formulation yields several advantages over previous Hausdorff matching methods. Feature uncertainties, in both the position and existence of the features, can be treated formally in the framework. Smoothly varying probability density functions can be used that eliminate the sharp boundary inherent in the conventional two-valued support function. In addition, it is simple to incorporate prior knowledge about the probability distribution of model positions in the matching process with this formulation.

We have described new techniques for performing matching efficiently with this formulation. Experiments on synthetic data indicate that the new techniques yield performance superior to the standard formulation with respect to both recognition and localization. Finally, we have applied this technique to the self-localization of a mobile robot in a natural environment using range maps from stereo vision. Improved results were also obtained in this domain.

Acknowledgments

The research described in this paper was carried out by the Jet Propulsion Laboratory, California Institute of Technology, under a contract with the National Aeronautics and Space Administration.

References

[1] H. S. Baird. *Model-Based Image Matching Using Location.* MIT Press, 1985.

[2] T. A. Cass. Polynomial-time geometric matching for object recognition. *International Journal of Computer Vision*, 21(1/2):37–61, January 1997.

[3] W. E. L. Grimson and D. P. Huttenlocher. Analyzing the probability of a false alarm for the Hausdorff distance under translation. In *Proceedings of the Workshop on Performance versus Methodology in Computer Vision*, pages 199–205, 1994.

[4] D. P. Huttenlocher and T. A. Cass. Measuring the quality of hypotheses in model-based recognition. In *Proceedings of the European Conference on Computer Vision*, pages 773–775, 1992.

[5] D. P. Huttenlocher, G. A. Klanderman, and W. J. Rucklidge. Comparing images using the Hausdorff distance. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 15(9):850–863, September 1993.

[6] D. P. Huttenlocher, M. E. Leventon, and W. J. Rucklidge. Visually-guided navigation by comparing two-dimensional edge images. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 842–847, 1994.

[7] D. P. Huttenlocher, J. J. Noh, and W. J. Rucklidge. Tracking non-rigid objects in complex scenes. In *Proceedings of the International Conference on Computer Vision*, pages 93–101, 1993.

[8] D. P. Huttenlocher and W. J. Rucklidge. A multi-resolution technique for comparing images using the Hausdorff distance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 705–706, 1993.

[9] K. Kedem and Y. Yarmovski. Curve based stereo matching using the minimum Hausdorff distance. In *Proceedings of the ACM Symposium on Computational Geometry*, pages 15–18, 1996.

[10] L. Matthies. Stereo vision for planetary rovers: Stochastic modeling to near real-time implementation. *International Journal of Computer Vision*, 8(1):71–91, July 1992.

[11] C. F. Olson. Mobile robot self-localization by iconic matching of range maps. In *Proceedings of the International Conference on Advanced Robotics*, pages 447–452, 1997.

[12] C. F. Olson and D. P. Huttenlocher. Automatic target recognition by matching oriented edge pixels. *IEEE Transactions on Image Processing*, 6(1):103–113, January 1997.

[13] C. F. Olson and L. H. Matthies. Maximum-likelihood rover localization by matching range maps. In *Proceedings of the International Conference on Robotics and Automation*, pages 272–277, 1998.

[14] A. Rosenfeld and J. Pfaltz. Sequential operations in digital picture processing. *Journal of the ACM*, 13:471–494, 1966.

[15] W. J. Rucklidge. *Efficient Visual Recognition Using the Hausdorff Distance.* Springer-Verlag, 1996.

[16] E. Saber and A. M. Tekalp. Image query-by-example using region-based shape matching. In *Image and Video Processing IV, Proc. SPIE 2666*, pages 200–211, 1996.

[17] W. M. Wells. Statistical approaches to feature-based object recognition. *International Journal of Computer Vision*, 21(1/2):63–98, 1997.

[18] X. Yi and O. I. Camps. Robust occluding contour detection using the Hausdorff distance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 962–968, 1997.

[19] R. Zabih, J. Miller, and K. Mai. Video browsing using edges and motion. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 439–446, 1996.