International Journal of Computer Vision 45(1), 39–54, 2001

A General Method for Geometric Feature Matching and Model Extraction

CLARK F. OLSON
University of Washington, Bothell, Department of Computing and Software Systems, 18115 Campus Way NE, Box 358534, Bothell, WA 98011-8246, USA
cfolson@u.washington.edu

Received March 3, 2000; Accepted June 15, 2001

Abstract. Popular algorithms for feature matching and model extraction fall into two broad categories: generate-and-test and Hough transform variations. However, both methods suffer from problems in practical implementations. Generate-and-test methods are sensitive to noise in the data. They often fail when the generated model fit is poor due to error in the data used to generate the model position. Hough transform variations are less sensitive to noise, but implementations for complex problems suffer from large time and space requirements and from the detection of false positives. This paper describes a general method for solving problems where a model is extracted from, or fit to, data that draws benefits from both generate-and-test methods and those based on the Hough transform, yielding a method superior to both. An important component of the method is the subdivision of the problem into many subproblems. This allows efficient generate-and-test techniques to be used, including the use of randomization to limit the number of subproblems that must be examined. Each subproblem is solved using pose space analysis techniques similar to the Hough transform, which lowers the sensitivity of the method to noise. This strategy is easy to implement and results in practical algorithms that are efficient and robust. We describe case studies of the application of this method to object recognition, geometric primitive extraction, robust regression, and motion segmentation.

Keywords: divide-and-conquer, feature matching, generate-and-test, geometric primitive extraction, Hough transform, model extraction, motion segmentation, object recognition, randomized algorithm, robust regression

1. Introduction

The generate-and-test paradigm is a popular strategy for solving model matching problems such as recognition, detection, and fitting. The basic idea of this method is to generate (or predict) many hypothetical model positions using the minimal amount of data necessary to identify unique solutions. A sequence of such positions is tested, and the positions that meet some criterion are retained. Examples of this technique include RANSAC (Fischler and Bolles, 1981) and the alignment method (Huttenlocher and Ullman, 1990).

The primary drawback to the generate-and-test paradigm is its sensitivity to noise. Let us call the data points (or other features, in general) that are used in predicting the model position for some test the *distinguished features*, since they play a more important role in whether the test is successful. The other features are *undistinguished features*. Errors in the locations of the distinguished features cause the predicted model position(s) to be in error. As the error grows, the testing step becomes more likely to fail.

To deal with this problem, methods have been developed to propagate errors in the locations of the distinguished features (Alter and Jacobs, 1998; Grimson et al., 1994). Under the assumption of a bounded error region for each of the distinguished features, these methods can place bounds on the locations in which various model features can be located in an image. If we count the number of image features that can be aligned with the model (with the constraint that the distinguished features must always be in alignment up

to the error bounds) these techniques can guarantee that we never undercount the number of alignable features. The techniques will thus never report that the model is not present according to some counting criterion when, in fact, the model does meet the criterion.

On the other hand, this method is likely to overcount the number of alignable features, even if the bounds on the location of each individual feature are tight. The reason for this is that, while the method checks whether there are model positions that brings each of the undistinguished image features into alignment with the model (along with all of the distinguished features) up to the error bounds, it does not check whether there is a single position that brings all of the counted undistinguished features into alignment up to the error bounds.

A competing technique for feature matching and model extraction is based on the Hough transform. This method also generates hypothetical model positions solutions using minimal information, but rather than testing each solution separately, the testing is performed by analyzing the locations of the solutions in the space of possible model positions (or *poses*). This is often, but not always, accomplished through a histogramming or clustering procedure. The large clusters in the pose space indicate good model fits. We call techniques that examine the pose space for sets of consistent matches among all hypothetical matches *Hough-based methods*, since they derive from the Hough transform (Illingworth and Kittler, 1988; Leavers, 1993). While these techniques are less sensitive to noise in the features, they are prone to large computational and memory requirements, as well as the detection of false positive instances (Grimson and Huttenlocher, 1990) if the pose space analysis is not precise.

In this paper, we describe a technique that combines the generate-and-test and Hough-based methods in a way that draws ideas and advantages from each, yielding a method that improves upon both. Like the generate-and-test method, (partial) solutions based on distinguished features are generated for further examination. However, each such solution is under-constrained and Hough-based methods are used to determine and evaluate the remainder of the solution. This allows both randomization to be used to reduce the computational complexity of the method and error propagation techniques to be used in order to better extract the model(s). We call this technique RUDR (pronounced "rudder"), for Recognition Using Decomposition and Randomization.

We will show that the problem can be treated as many subproblems, each of which is much simpler than the original problem. We then discuss various methods by which the subproblems can be solved. The application of randomization to reduce the number of subproblems that must be examined is next described. These techniques yield efficiency gains over conventional generate-and-test and Hough-based methods. In addition, the subdivision of the problem allows us to examine a much smaller parameter space in each of the subproblems than in the original problem and this allows the error inherent in localization procedures to be propagated accurately and efficiently in the matching process.

This method has a large number of applications. It can be applied to essentially any problem where a model is fit to cluttered data (that is, with outliers or multiple models present). We describe case studies of the application of this method to object recognition, curve detection, robust regression, and motion segmentation.

The methodology described here is a generalization of previous work on feature matching and model extraction (Olson, 1997a, 1997b, 1999). Similar ideas have been used by other researchers. A simple variation of this method has been applied to curve detection by Murakami et al. (1986) and Leavers (1992). In both of these papers, the problem decomposition was achieved through the use of a single distinguished feature in the image for each of the subproblems. We argue that the optimal performance is achieved when the number of distinguished features is one less than the number necessary to fully define the model position in the errorless case. This has two beneficial effects. First, it reduces the amount of the pose space that must be considered in each problem (and the combinatorial explosion in the sets of undistinguished features that are examined). Second, it allows a more effective use of randomization in reducing the computational complexity of the method. A closely related decomposition and randomization method has been described by Cass (1997) in the context of pose equivalence analysis. He uses a "base match" to develop an approximation algorithm for feature matching with uncertain data.

2. Related Research

In this section, we review research on generate-and-test and Hough-based algorithms for model matching.

2.1. Generate-and-Test Methods

The basic idea in generate-and-test methods is to sequentially generate hypothetical model positions in the data and test the positions. The hypothetical model positions are generated by conjecturing matches between the model and a set of features in the data and then determining the model position(s) such that the model is aligned with the data features. These positions are tested by comparing the position of the model against the remainder of the data. One common measure of the quality of the position is the number of data features that agree well with the model at this position.

Fischler and Bolles (1981) described the RANSAC (for RANdom SAmple Consensus) technique. They suggested that, when solving for the model position, it is best to use the minimum number of data features necessary to yield a finite set of solutions (assuming no error). This reduces the likelihood that one of the data features does not belong to the model, since there may be outliers or multiple models present. They also used a randomization technique, where sets of data features were sampled randomly until the probability of at least one sample being correct is sufficiently large, assuming that the model is actually present in the image. In addition to applying these techniques to pose determination from point features (Fischler and Bolles, 1981), these techniques were applied to curve and surface detection (Bolles and Fischler, 1981).

One problem that generate-and-test methods can have is that error in the data features causes error in the model position estimate that can result in failure to detect the model correctly. An alternative to simply testing the model position corresponding to some hypothesized data set is to iteratively refine the model position as additional data features are conjectured to belong to the model (Ayache and Faugeras, 1986; Faugeras and Hebert, 1986; Lowe, 1987). This can significantly improve the overall match, particularly when the initial data set is noisy, although it will not help if the initial matching contains an outlier. Another technique, which can be used if the localization error can be modeled, is to carefully propagate the effects of localization error in the testing step (Alter and Jacobs, 1998; Grimson et al., 1994). While this technique will not miss instances of a model that satisfy some error criterion, it will find instances that do not satisfy the criterion.

2.2. Hough-Based Methods

Parameter space analysis techniques can be traced back to the patent of the Hough transform (Hough, 1962). The Hough transform was initially used to track particle curves in bubble-chamber imagery. Subsequent work by Rosenfeld (1969) and Duda and Hart (1972) played a substantial role in popularizing the Hough transform and it has since become an established technique for the detection of curves and surfaces, as well as many other applications. The basic idea of the Hough transform is that each data feature can be mapped into a manifold in the parameter space of possible curves (or model positions, in general). Typical implementations consider a quantized parameter space and count the number of data features that map to a manifold intersecting each cell in the quantized space. Cells with high counts correspond to curves in the image. Surveys of the Hough transform and applications have been published by Illingworth and Kittler (1988) and Leavers (1993).

Ballard (1981) demonstrated that the Hough transform can be generalized to detect arbitrary shapes (composed of discrete points) in images. This method determines a mapping between image features and an accumulator for a parameter space describing the possible positions of the object. When each image feature has been mapped into the parameter space, instances of the shape yield local maxima in the accumulator. Ballard used pixel orientation information to speed up the algorithm and improve accuracy. It was also suggested that pairs of features could be mapped into the parameter space to reduce the effort in mapping features into the parameter space, but this was considered infeasible for most cases.

More recent work has incorporated the idea of mapping sets of features into the parameter space (Bergen and Shvaytser, 1991; Leavers, 1992; Xu et al., 1990). The primary benefit that is gained from this is that only those feature sets that are large enough to map to a single point (or a finite set of points) in the parameter space are examined. A drawback is that there are many such sets of features. An additional technique that has proven useful for improving the efficiency of the Hough transform is randomization (Bergen and Shvaytser, 1991; Kiryati et al., 1991; Leavers, 1992; Xu et al., 1990).

Generalized Hough transform techniques have been applied to many object recognition problems (and these techniques are sometimes called pose clustering in this context) (Linnainmaa et al., 1988; Silberberg

et al., 1984; Stockman, 1987; Thompson and Mundy, 1987). Most of these applications consider feature sets that map to single points in the space of possible model transformations and then perform some clustering method to detect objects (often multi-dimensional histogramming).

3. General Problem Formalization

The class of problems that we solve using RUDR are those that require a model to be fit to a set of observed data features, where a significant portion of the observed data may be outliers (in fact, the model may contain outliers as well) or there may be multiple models present in the data. These problems can, in general, be formalized as follows.

Given

- $\mathcal{D}$—the data to match. The data is a set of a set of features or measurements, $\{\delta_1, \ldots, \delta_d\}$, that have been extracted, for example, from an image. For simplicity, we assume that all of the data features are of a single geometric class, such as points or segments, but this restriction can be removed.
- $\mathcal{M}$—the model to be fit. This model may be a set of distinct features as is typical in object recognition, or it may be a parameterized manifold such as a curve or surface, as in curve detection and robust regression. The only constraint on the model is that we must be able to determine hypothetical model positions by matching the model with sets of data features.
- $\mathcal{T}$—the set of possible positions or transformations of the model. We use τ to denote individual transformations in this space.
- $A(\mathcal{M}, \mathcal{D}, \mathcal{T}, \tau, D)$—the acceptance criterion. A binary-valued criterion that specifies whether a transformation τ satisfactorily brings the model into agreement with a set of data features, $D \in \mathcal{D}$. We allow this criterion to be a function of the full set of data features and the set of transformations to allow the criterion to select the single best subset of data features according to some criterion or to take into account global matching information.

Determine and Report

- All maximal sets of data features, $D \in \mathcal{D}$, for which there is a transformation, $\tau \in \mathcal{T}$, such that the acceptance criterion, $A(\mathcal{M}, \mathcal{D}, \mathcal{T}, \tau, D)$, is satisfied. (Only the maximal sets are reported so that the subsets of each maximal set need not be reported.)

This formalization is very general. Many problems can be formalized in this manner, including object recognition, geometric primitive extraction, motion segmentation, and robust regression.

A useful acceptance criterion is based on bounding the fitting error between the model and the data. Let $C(\mathcal{M}, \delta, \tau)$ be a function that determines whether the specified position of the model fits the data feature δ (for example, up to a bounded error). We let $C(\mathcal{M}, \delta, \tau) = 1$, if the criterion is satisfied, and $C(\mathcal{M}, \delta, \tau) = 0$, otherwise. The model is said to be brought into alignment with a set of data features, $D = \{\delta_1, \ldots, \delta_x\}$ up to the error criterion, if all of the individual features are brought into alignment:

$$\prod_{i=1}^{x} C(\mathcal{M}, \delta_i, \tau) = 1 \tag{1}$$

The bounded-error acceptance criterion specifies that a set of data features, $D = \{\delta_1, \ldots, \delta_x\}$, should be reported if the cardinality of the set meets some threshold $(x \geq c)$, there is a position of the model that satisfies (1), and the set is not a subset of some larger set that is reported.

While this criterion does not incorporate global information, such as mean-square-error or least-median-of-squares, RUDR is not restricted to using this bounded-error criterion. This method has been applied to least-median-of-squares regression with excellent results (Olson, 1997a).

Example. As a running example, we will consider detecting circles in two-dimensional image data. For this case, our model $\mathcal{M}$ is simply the parameterization of a circle, $(x - x_c)^2 + (y - y_c)^2 = r^2$, and our data $\mathcal{D}$ is a set of image points. The space of possible transformations is the space of circles, $\mathcal{T} = [x_c, y_c, r]^T$. We use a bounded-error acceptance criterion such that a point is considered to be on the circle if $|\sqrt{(x - x_c)^2 + (y - y_c)^2} - r| < \epsilon$. We will report the circles that have $\sum_{i=1}^{d} C_\epsilon(\mathcal{M}, \delta_i, \tau) > \pi r$. In other words, we search for the circles that have approximately half of their perimeter present in the image, although any other threshold could be used.

4. RUDR Approach

The main components of the approach that we take are as follows. First, we subdivide the problem into many

smaller subproblems, as in generate-and-test methods. However, our subproblems do not fully constrain the pose of the model. Each subproblem is solved using a Hough transform-based method to search a partially constrained pose space in which with model may lie. Randomization techniques are used to limit the number of subproblems that are examined with a low probability of failure.

Let us call the hypothetical correspondence between a set of data features and the model a *matching*. The generate-and-test paradigm and many Hough-based strategies solve for hypothetical model positions using matchings of the minimum cardinality to constrain the model position up to a finite ambiguity (assuming errorless features). We call the matchings that contain this minimal amount of information *minimal matchings* and denote their cardinality k. We consider two types of models. One type of model consists of a set of discrete features similar to the data features. The other is a parameterized model such as a curve or surface. When the model is a set of discrete features, the minimal matchings specify the model features that match each of the data features in the minimal matching and we call these *explicit matchings*. Otherwise, the data features are matched implicitly to positions in parameterized model and we, thus, call these *implicit matchings*.

In the generate-and-test paradigm, the model positions generated using the minimal matchings are tested by determining how well the undistinguished features are fit according to the predicted model position. In Hough-based methods, it is typical to determine the positions of the model that align each of the minimal matchings and detect clusters of these positions in the parameter space that describes the set of possible model positions,[1] but other pose space analysis techniques can be used (Breuel, 1992; Cass, 1997).

The approach that we take draws upon both generate-and-test techniques and Hough-based techniques. The underlying matching method may be any one of several pose space analysis techniques in the Hough-based approach (see Section 5), but unlike previous Hough-based methods, the problem is subdivided into many smaller problems, in which only a subset of the minimal matchings is examined. When randomization is applied to selecting which subproblems to solve, a low computational complexity can be achieved with a low probability of failure.

The key to this method is the subdivision of the problem into many small subproblems, in which a *distinguished matching* of some cardinality $g < k$ between data features and the model is considered. Only those minimal matchings that contain the distinguished matching are examined in each subproblem and this constrains the portion of the pose space that the subproblem considers. We could consider each possible distinguished matching of the appropriate cardinality as a subproblem, but we shall see that this is not necessary in practice.

In the remainder of this section, we show that the subdivision of the problem into many smaller subproblems does not inherently change the solutions that are detected by the algorithm. We then discuss the optimal cardinality for the distinguished matching that constrains each subproblem and the application of this methodology using approximation algorithms. Following sections describe algorithms that can be used to solve the individual subproblems and the use of randomization to limit the complexity of the algorithms.

4.1. Problem Equivalence

Let's consider the effect of the decomposition of the problem on the matchings that are detected by a system using a bounded-error criterion, $C(\mathcal{M}, d, t)$, as described above. For now, we assume that we have some method of determining precisely those sets of data features that should be reported according to the bounded-error acceptance criterion. The implications of performing matching only approximately and the use of an acceptance criterion other than the bounded-error criterion are discussed subsequently.

Proposition 1. *For any transformation, $\tau \in \mathcal{T}$, the following statements are equivalent*:

1. *Transformation τ brings at least x data features into alignment with the model up to the error criterion.*
2. *Transformation τ brings at least $\binom{x}{k}$ sets of data features with cardinality k into alignment with the model up to the error criterion.*
3. *For any distinguished matching with cardinality g that is brought into alignment with the model up to the error criterion by τ, there are $\binom{x-g}{k-g}$ minimal matchings that contain the distinguished matching that are brought into alignment up to the error criterion by τ.*

Proof: The proof follows from combinatorics. We sketch the proof that (a) Statement 1 implies

Statement 2, (b) Statement 2 implies Statement 3, and (c) Statement 3 implies Statement 1. The statements are thus equivalent.

(a) From Statement 1, there are at least x data features with $C(\mathcal{M}, \delta_i, \tau) = 1$. We can thus form at least $\binom{x}{k}$ distinct sets of these data features with cardinality k. Each such set has $\prod_{i=1}^{k} C(\mathcal{M}, \delta_i, \tau) = 1$. These matchings thus contribute at least $\binom{x}{k}$ to the sum.

(b) To form the $\binom{x}{k}$ sets of data features that are brought into alignment with the model, we must have x individual data features satisfying $C(\mathcal{M}, \delta_i, \tau) = 1$. (If there were $y < x$ such features then we could only form $\binom{y}{k}$ minimal matchings satisfying Eq. (1).) Choose any subset $\mathcal{G}$ of these matches with cardinality g. Form the $\binom{x-g}{k-g}$ subsets with cardinality $k-g$ that do not include any feature in $\mathcal{G}$. Each of these subsets when combined with $\mathcal{G}$ forms a minimal matching that is brought into alignment up to the error criterion since each of the individual features satisfies $C(\mathcal{M}, \delta_i, \tau) = 1$.

(c) From Statement 3, the g data features in the distinguished matching are brought into alignment up to the error criterion by τ. In addition there must exist $x - g$ additional data features that are brought into alignment up to the error criterion by τ to form the $\binom{x-g}{k-g}$ subsets of cardinality $k-g$ that are brought into alignment up to the error criterion by τ. Thus, in total, there must be $g + x - g = x$ data features that are brought into alignment up to the error criterion by τ. □

Statement 3 indicates that as long as we examine one distinguished matching that belongs to each of the matchings that should be reported, the strategy of subdividing the problem into subproblems yields equivalent results to examining the original problem as long as the threshold on the number of matches is set appropriately.

This decomposition of the problem allows our method to be viewed as a class of generate-and-test methods, where distinguished matchings (rather than minimal matchings) are generated and the testing step is performed using a pose space analysis method (such as clustering or pose space equivalence analysis) rather than comparing a particular model position against the data. Methods for solving the subproblems are discussed further in Section 5.

4.2. Optimal Cardinality

While distinguished matchings of any cardinality could be considered, we must balance the complexity of the subproblems with the number of subproblems that are examined. Increasing the cardinality of the distinguished matching is beneficial up to a point. As the size of the distinguished matching is increased, the number of minimal matchings that is examined in each subproblem is decreased and we have more constraint on the position of the model. The subproblems are thus simpler to solve.

By itself, this does not necessarily improve matters, since there are more subproblems to examine. However, since we use randomization to limit the number of subproblems that are examined, a lower computational complexity is achieved by having more simple subproblems than fewer difficult ones. Section 6 shows why this is true. On the other hand, when we reach $g = k$, the method becomes no better than a standard generate-and-test technique and we lose both the benefits gained through the Hough-based analysis of the pose space and the property that the subproblems become simpler with larger distinguished matchings. We, thus, use distinguished matchings with cardinality $g = k - 1$. For each subproblem, this means that we examine a set of minimal matchings that share $k - 1$ data features and that vary in only one feature.

4.3. Approximation Algorithms

For practical reasons, we may not wish to use an algorithm that reports exactly those matchings that satisfy the error criterion, since such algorithms are often time consuming. In this case, we cannot guarantee that examining a distinguished matching that belongs to a solution that should be reported will result in detecting that solution. However, empirical evidence suggests that the examination of these subproblems yields superior results when an approximation algorithm is used (Olson, 1997b), owing to failures that occur in the examination of full problem.

We can also use these techniques with acceptance criteria other than the bounded-error criterion. With other criteria, the proposition is not always true, but if an approximation algorithm is used to detect good matchings, examination of the subproblems often yields good results. For example, an application of these ideas to least-median-of-squares regression has yielded an approximation algorithm that is provably

accurate with high probability, while previous approximation algorithms do not have this property (Olson, 1997a).

Example. For our circle detection example, $k = 3$, since three points are sufficient to define a circle in the errorless case. The above analysis implies that, rather than examining individual image features, or all triples of features, we should examine trials (or subproblems) consisting of only the triples that share some distinguished pair of features in common. Multiple trials are examined to prevent missing a circle.

5. Solving the Subproblems

Now, we must use some method to solve each of the subproblems that are examined. We can use any method that determines the number of matchings with a given cardinality can be brought approximately into alignment with the model at a particular position. We discuss histogramming first, which is the simplest method, but is prone to errors. Pose constraint methods, which are precise methods for locating good pose subject to bounded error constraints, are discussed next. Finally, we describe a middle ground, where errors are propagated into a small subspace and then clustering is performed.

5.1. Histogramming

The simplest method for solving the subproblems is to use a multi-dimensional histogramming step in order to locate large clusters in the pose space. In this method, the parameter space is quantized and a counter is maintained for each cell in the quantized space. Each matching is mapped into a manifold in the parameter space and the counter associated with each parameter space cell that intersects that manifold is incremented. Clusters in the parameter space are found by locating cells with high counts. The primary reason this method is popular is that it requires linear time in the number of matchings that are considered, while most other methods are more complex. While this method is fast if the parameter space is not too large and yields good results for many applications, it does not propagate the effects of error accurately, it introduces quantization effects, and it is time consuming for large parameter spaces.

To reduce the problem of a large parameter space, hierarchical decompositions of the parameter space have often been used. This is usually performed through a coarse-to-fine search of the parameter space or by decomposing the parameter space along the orthogonal axes. Such decompositions of the parameter space allow the initial steps to be performed using only a few cells in the transformation space. Those that cannot lead to a large cluster are eliminated and the rest are examined in finer detail.

The problem of error propagation is more difficult to handle with this technique. For complex problems, it can become problematic to detect the clusters without also detecting a significant number of false positives (Grimson and Huttenlocher, 1990). One possibility is to discretize the pose space finely and then increment the counter for each cell that is consistent with each minimal matching (Shapiro, 1978). However, this requires much computation.

5.2. Pose Constraint Methods

Alternatively, pose equivalence analysis techniques that have been developed by Breuel (1992) and Cass (1997) can be applied that allow localization error to be propagated accurately. The basic idea of these methods is that each match between a model feature and a data feature yields some constraints on the model position that must be satisfied in order for the features to be aligned up to the error boundary. These constraints divide the pose space into equivalence classes, within which the same sets of features are brought into correspondence. A search of the pose space is performed to locate the positions that satisfy the maximum number of constraints.

Cass (1997) searches the pose space by examining the arrangement of the constraints using computational geometry techniques. A significant speedup is achieved through the use of an approximation algorithm. Breuel (1992) uses a method that adaptively divides the pose space, pruning cells that are not consistent with a sufficient number of the constraints. Unpruned cells are divided recursively until they are small enough to accept as valid model positions. Breuel's experiments suggest that his techniques can operate in approximately linear expected time in the number of matchings, so this step can be performed efficiently in many cases.

Both of these techniques can be used in conjunction with our approach to subdividing the problem. In the method of Cass, much less of the pose space is examined in each trial, since only a few of the pose equivalence classes are consistent with each distinguished

matching. In Breuel's method, k-tuples are used to form closed constraint regions in the pose space, so our approach would examine only the k-tuples that contain the distinguished matching in each trial.

5.3. Subspace Error Propagation

In our method, only a small portion of the parameter space is examined in each subproblem. If it is assumed that there is no error in the data features in the distinguished matching, then each subproblem considers only a sub-manifold of the parameter space. In general, if there are p transformation parameters and each feature match yields c constraints on the transformation, then a subproblem where the distinguished matchings have cardinality g considers only a $(p - gc)$-dimensional manifold of the transformation space in the errorless case.[2] This allows us to parameterize the sub-manifold (using $p - gc$ parameters) and perform analysis in this lower dimensional space. A particularly useful case is when the resulting manifold has only one dimension (i.e. it is a curve). In this case, the subproblem can be solved very simply by parameterizing the curve and finding positions on the curve that are consistent with many minimal matchings.

When localization error in the data features is considered, the subproblems must (at least implicity) consider a larger space than the manifold described above. The subproblems are still much easier to solve. A technique that is useful in this case is to project the set of transformations that are consistent with each minimal matching up to the error criterion onto the manifold that results in the errorless case and then perform clustering only in the parameterization of this manifold as discussed above (Olson, 1999).

This method slightly overestimates the total number of consistent matches, since matches that are consistent in the projections may not be in the full pose space. However, significant errors are unlikely, because the regions of the pose space consistent with each minimal matching do not deviate far from the manifold corresponding the errorless case, except in extreme circumstances (cases that are nearly degenerate).

Example. For circle detection, we saw previously that we should use two distinguished points. The circle positions that share a pair of points lie on a curve in the pose space. (The center of the circle is always on the perpendicular bisector of the two distinguished points.) For triples of points that contain the two distinguished points, we parameterize the positions using the signed distance d from the center of the circle to the midpoint between the distinguished points (positive if above, negative if below). This yields a unique descriptor for every circle passing through the distinguished points. For each triple that is considered, we can project the pose space consistent with the triple onto the parameterization by considering which centers are possible given some error bounds on the point locations (Olson, 1999). We determine if a circle is present in each trial by finely discretizing d and performing a simple Hough transform variation in this one-dimensional space, where the counter for each bin is incremented for each triple that is consistent with the span represented by the counter. Peaks in the accumulator are accepted if they surpass some predetermined threshold. This process is repeated for several pairs of distinguished points to ensure a low probability of failure, as described in the next section.

6. Randomization and Complexity

A deterministic implementation of these ideas should examine each possible distinguished matching with the appropriate cardinality. This requires $O(n^k)$ time, where n is the number of possible matches between a data feature and the model. When explicit matchings are considered, $n = md$, where m is the number of model features and d is the number of data features. When implicit matchings are considered, $n = d$. Such a deterministic implementation performs much redundant work. There are many distinguished matchings that are part of each of the large consistent matchings that we are seeking. We thus find each matching that meets the acceptance criterion many times (once for each distinguished matching that is contained in the maximal matching). We can take advantage of this redundancy through the use of a common randomization technique to limit the number of subproblems that we must consider while maintaining a low probability of failure.

Assume that some minimum number of the image features belong to the model. Denote this number b. Since our usual acceptance criterion is based on counting the number of image features that belong to the model, we can allow the procedure to fail when too few image features belong to the model. Otherwise, the probability that some set of image features with cardinality $g = k - 1$ completely belongs to the model is bounded by $\binom{b}{k-1}/\binom{d}{k-1} \approx (\frac{b}{d})^{k-1}$. If we take t trials that select sets of $k - 1$ image features randomly,

then the probability that none of them will completely belong to the model is:

$$p_t \approx \left(1 - \left(\frac{b}{d}\right)^{k-1}\right)^t. \quad (2)$$

Setting this probability below some arbitrarily small threshold ($p_t < \gamma$) yields:

$$t \approx \frac{\ln \gamma}{\ln\left(1 - \left(\frac{b}{d}\right)^{k-1}\right)} \approx \left(\frac{d}{b}\right)^{k-1} \ln \frac{1}{\gamma}. \quad (3)$$

Now, for explicit matches, we assume that some minimum fraction f_e of the model features appear in the image. In this case, the number of trials necessary is approximately $(\frac{d}{f_e m})^{k-1} \ln \frac{1}{\gamma}$. For each trial, we must consider matching the set of image features against each possible matching set of model features, so the total number of distinguished matchings that are considered is approximately $(\frac{d}{f_e})^{k-1}(k-1)! \ln \frac{1}{\gamma}$. Each explicit distinguished matching requires $O(md)$ time to process, so the overall time required is $O(md^k)$. Note that if we use $g < k-1$ (that is, a smaller set of distinguished points than we have been discussing,) we will examine $O(m^{k-g}n^{k-g})$ minimal matchings for each distinguished matching and the overall complexity increases to $O(m^{k-g}d^k)$. For this reason, the use of fewer than $g = k-1$ distinguished points is suboptimal.

For implicit matches, we assume that each significant model in the image comprises some minimum fraction f_i of the image features. The number of trials necessary to achieve a probability of failure below γ is approximately $f_i^{1-k} \ln \frac{1}{\gamma}$, which is a constant independent of the number of image features. Since each trial can be performed in $O(d)$ time, the overall time required is $O(d)$. If a smaller distinguished matching was used, the trials would require $O(d^{k-g})$ time, since we consider all minimal matchings that contain the distinguished matching. Again we see that the use of a small distinguished matching yields an increased computational complexity.

Note that the complexity can be reduced further in some cases by performing subsampling among the minimal matchings considered in each trial. Indeed, $O(1)$ complexity is possible with some assumptions about the number of features present and the rate of errors allowable (Bergen and Shvaytser, 1991). We have not found this further complexity reduction to be necessary in our experiments. In fact, in most cases the number of samples necessary with this technique is large. However, this technique may be useful when the number of image features is very large.

Grouping techniques can also be used to improve the computational complexity of the algorithm. If we have some method that can determine feature sets that are more likely to belong to the same model, we can use these as sets of distinguished features in the image. Such methods can be used to reduce the likelihood of a false positive match in addition to reducing the computational complexity (Olson, 1998).

Example. Our circle detection case uses implicit matchings. If we assume that each circle that we wish to detect comprises at least $f_i = 5\%$ of the image data and require that the probability of failure is below $\gamma = 0.1\%$, then the number of trials necessary is 2764. Each trial considers the remaining $d-2$ image features. Note that techniques considering all triples will surpass the number of triples considered here when $d > 53$.

7. Comparison with Previous Techniques

This section gives a comparison of the RUDR approach with previous generate-and-test and Hough-based techniques.

Deterministic generate-and-test techniques require $O(n^{k+1})$ time to perform model extraction in general, since there are $O(n^k)$ minimal matchings and the testing stage can be implemented $O(n)$ time. This can often be reduced slightly through the use of efficient geometric searching techniques during the testing stage. RUDR yields a superior computational complexity requirement. When randomization is applied to generate-and-test techniques, the computational complexity becomes $O(md^{k+1})$ (or slightly better using efficient geometric search) for explicit matches and $O(d)$ for implicit matches. RUDR yields a superior computational complexity for the case of explicit matches and, while the generate-and-test approach matches the complexity for the case of implicit matches, RUDR examines less subproblems by a constant factor (approximately $\frac{1}{f_i}$) and is, thus, faster in practice.

In addition, previous generate-and-test techniques are inherently less precise in the propagation of localization error. The basic generate-and-test algorithm introduces false positives unless care is taken to propagate the errors correctly (Alter and Jacobs, 1998; Grimson et al., 1994), since error in the data features leads to error in the hypothetical model pose and this error causes some of the models to be missed as a result of a poor fit. On the other hand, when error propagation is used,

false positives are introduced. The error propagation techniques ensure that each of the undistinguished features can be separately brought into alignment (along with the distinguished set) up to some error bounds by a single model position, this position may be different for each such feature match. It does not guarantee that *all* of the features can be brought into alignment up to the error bounds by a single position and thus causes false positives to be found.

Hough-based methods are capable of propagating localization error such that neither false positives nor false negatives occur (in the sense that only matchings meeting the acceptance criterion are reported) (Breuel, 1992; Cass, 1997). However, previous Hough-based methods have had large time and space requirements. Deterministic Hough-based techniques that examine minimal matchings require $O(n^k)$ time and considerable memory (Olson, 1997b).

Randomization has been previously applied to Hough transform techniques (Bergen and Shvaytser, 1991; Kiryati et al., 1991; Leavers, 1992; Xu et al., 1990). However, in previous methods, randomization has been used in a different manner than it is used here. While our approach examines all of the data in each of the subproblems, previous uses of randomization in Hough-based methods have subsampled the overall data examined, causing both false positives and false negatives to occur as a result. While false negatives can also occur due to the use of randomization in our approach, the probability of such an occurrence can be set arbitrarily low, with a logarithmic dependency on the failure rate.

Our method draws the ability to propagate localization error accurately from Hough-based methods and combines it with the ability to subdivide the problem into many smaller subproblems and thus gain the full benefit of randomization techniques. The result is a model extraction algorithm with superior computational complexity to previous methods that is also robust with respect to false positives and false negatives.

All of the techniques considered so far have been model-based methods. The primary drawback to such techniques is a combinatorial complexity that is polynomial in the number of features, but exponential in the complexity of the pose space (as measured by k). This can be subverted in some cases by assuming that some fraction of the data features arises from the model (this shifts the base of the exponent to be the required fraction). An alternative that can be useful in reducing this problem is the use of grouping or perceptual organization methods that use data-driven techniques to determine features that are likely to belong to the same model (for example, Jacobs, 1996; Lowe, 1985). In cases where models can be identified by purely data-driven methods, such techniques may be faster than the techniques described here. However, we have shown that even imperfect feature grouping methods can improve both the complexity and lower the rate of false positives in the RUDR method (Olson, 1998).

There are some situations where RUDR can not be applied effectively. If a single data feature is sufficient to constrain the position of the model, the RUDR problem decomposition will not be useful. In addition, the techniques we describe will be of less value when there is a small number of features in the image. In this case, the randomization may not yield an improvement in the speed of the algorithm. However, the error propagation techniques will still be beneficial.

8. Case Studies

We have studied the application of these ideas to several problems. We review the important aspects of these applications here and discuss additional areas where RUDR can be applied. The problems described here show some of the generality of this method. However, there remain a wide variety of additional problems where these techniques have not yet been applied.

8.1. Extraction of Geometric Primitives

As previously discussed, the Hough transform is a well known technique for geometric primitive extraction (Illingworth and Kittler, 1988; Leavers, 1993). The application of RUDR to this method improves the efficiency of the technique, allows the localization error to be propagated accurately, and reduces the amount of memory that is required (Olson, 1999).

Consider the case of detecting curves from feature points in two-dimensional image data. If we wish to detect curves with p parameters, then we use distinguished matchings consisting of $p - 1$ feature points, since, in general, p points are required to solve for the curve parameters. Each distingusihed matching maps to a one-dimensional manifold (a curve) in the parameter space, if the points are errorless and in general position. Methods have been developed to map minimal matchings with bounded error into segments of the curve for the cases of lines and circles (Olson, 1999). $O(d)$ time and space is required for curve detection

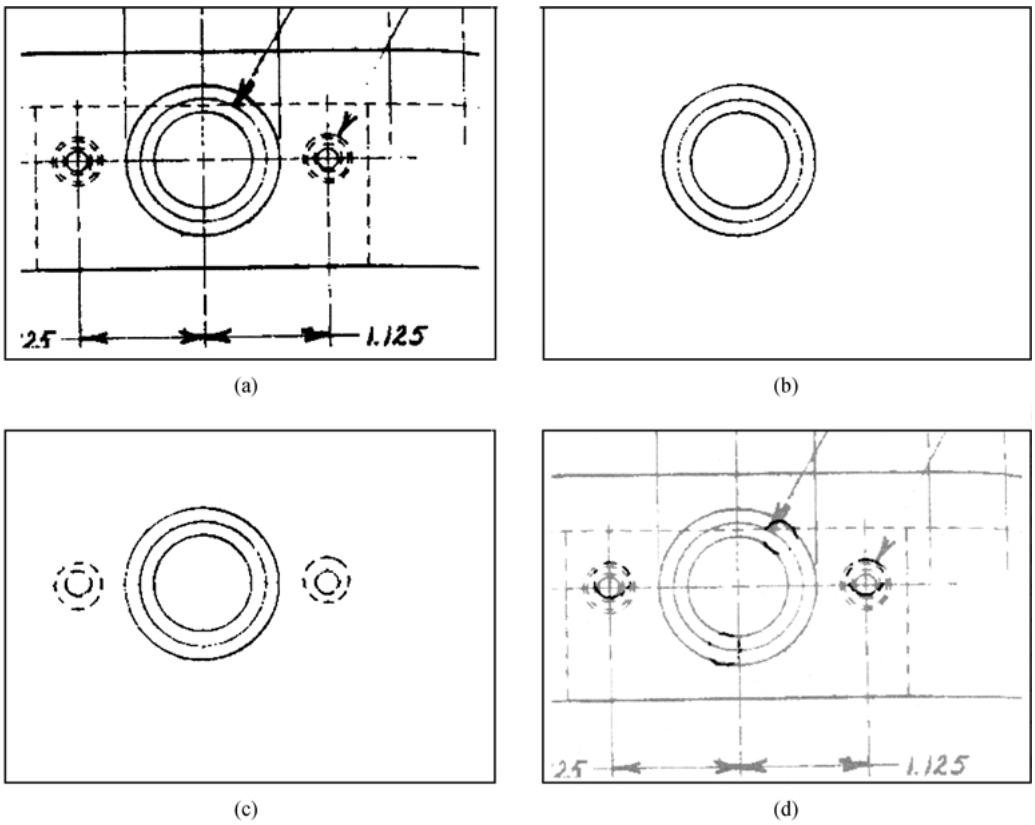

Figure 1. Circle detection. (a) Engineering drawing. (b) Circles found comprising 4% of the image. (c) Perceptually salient circles found comprising 0.8% of the image. (d) Insalient circles found comprising 0.8% of the image.

with these techniques, where d is the number of data points extracted from the image.

The case of circle detection has been discussed in some detail in the running example. Figure 1 shows the results of using RUDR to detect circles in a binary image of an engineering drawing. Note that, when a low threshold is used, small circles are found that are not perceptually salient. These circles meet the acceptance criterion specified, so this is not a failure of the algorithm.

The image in Fig. 1 contains 9299 edge pixels. In order to detect circles comprising 4% of the image, RUDR examines 4318 trials and considers 4.01×10^7 triples. Contrast this to the 8.04×10^{11} possible triples. A generate-and-test technique using the same type of randomization examines 1.08×10^5 trials (1.00×10^9 triples) to achieve the same probability of examining a correct trial, but will still miss circles due to the error in the features.

The robustness of this technique for line detection has been compared against other methods in a large number of synthetic images. The methodology and an example synthetic image have been previously published (Olson, 1999). This methodology has been extended to include generate-and-test methods here. Five methods have been compared:

1. The RUDR paradigm with propagated localization error.
2. The RUDR paradigm without propagated localization error.
3. A method mapping pairs of points into the parameter space, but without decomposition into subproblems.
4. The standard Hough transform.
5. Generate-and-test line detection. Four times as many trials were used as in the first and second methods.

Figure 2 shows the results. For each method, the probability of detecting the single correct line segment present in the image is plotted versus the probability of finding a false positive (curved distractors were added

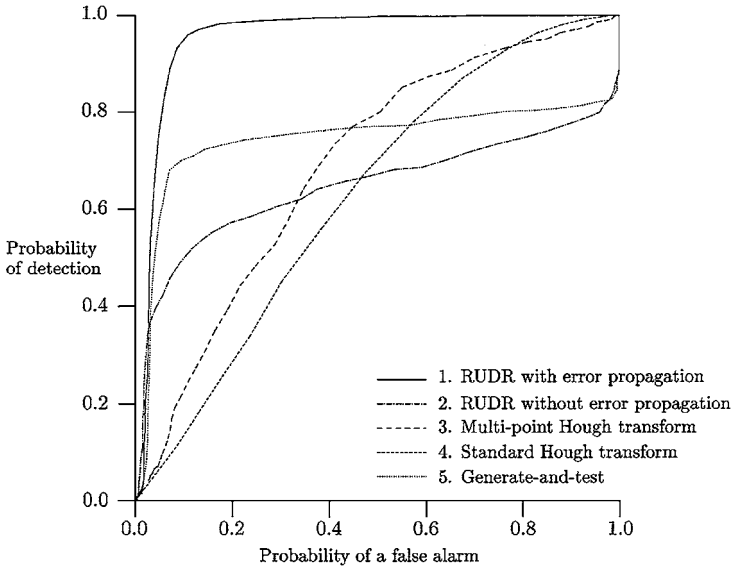

Figure 2. Receiver operating characteristic (ROC) curves for line detection generated using synthetic data.

to the images) for varying levels of the threshold used to determine which lines are detected.

The best performance is achieved by the RUDR paradigm with propagation of localization error into the parameter space. It is important to note that the RUDR paradigm fares much worse when localization error is not propagated carefully. This reason for this is that the method without propagation only succeeds when one of the trials that is examined uses a distinguished matching that is quite close to the model to be detected. However, when a trial with a significant amount of error is examined, the method becomes likely to fail (like generate-and-test methods), due to the poor model fit. This experiment demonstrates the importance of propagating the localization error into the parameter space when using techniques with a generate-and-test component.

8.2. Robust Regression

RUDR has been applied to the problem of finding the least-median-of-squares (LMS) regression line. In this case, a single distinguished point is examined in each trial (since two points are required to define a line). For each trial, we determine the line that is optimal with respect to the median residual, but with the constraint that the line must pass through the distinguished point.

It can be shown that the solution to this constrained problem has a median residual that is no more than the sum of the optimal median residual and the distance of the distinguished point from the optimal LMS regression line (Olson, 1997a). Now, at least half of the data points must lie no farther from the optimal regression line than the optimal median residual (by definition). Thus, each trial has a probability of at least 0.5 of obtaining a solution with a residual no worse than twice the optimal median residual. The use of randomization implies that we need to perform only a constant number of trials to achieve a good solution with high probability (approximately $-\log_2 \delta$ trials are necessary to achieve an error rate of δ).

Each subproblem (corresponding to a distinguished point) can be solved using a specialized method based on parametric search techniques (Olson, 1997a). This allows each subproblem to be solved exactly in $O(n \log^2 n)$ time or in $O(n \log n)$ time for a fixed precision solution using numerical techniques. These techniques have also been extended to problems in higher dimensional spaces.

The complexity of our method is superior to the best known exact algorithms for this problem (Edelsbrunner and Souvaine, 1990). The PROGRESS algorithm (Rousseeuw and Leroy, 1987) is a commonly used approximation algorithm for LMS regression that uses the generate-and-test paradigm. It requires $O(n)$ time. The basic method for line detection is to sample pairs of points from the data and examine the median residual of the line that passes through the points. The line

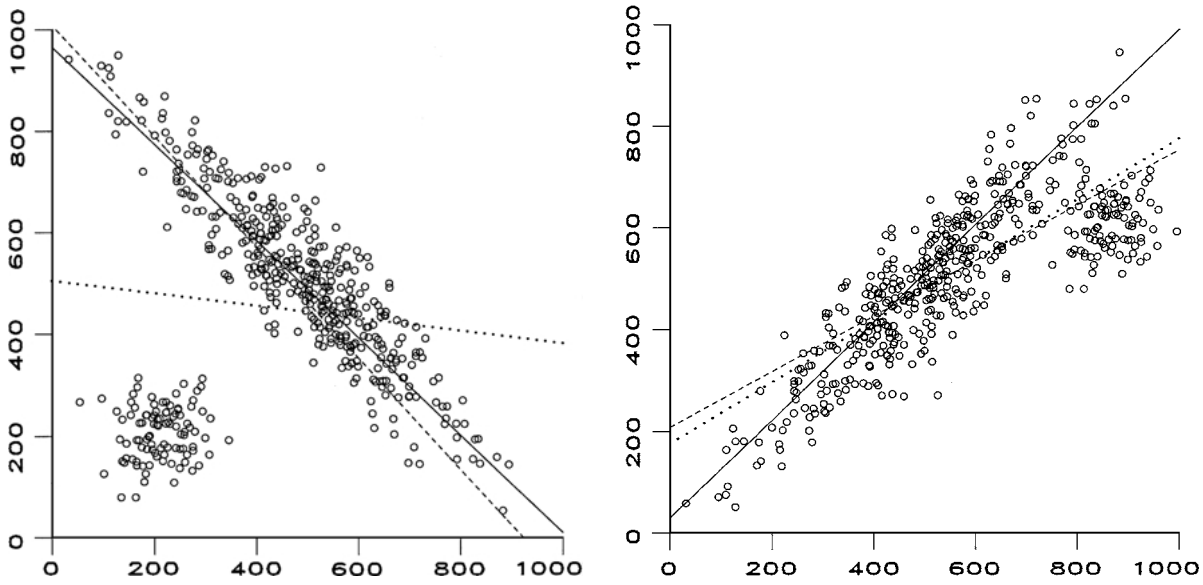

Figure 3. Robust regression examples. The solid lines are the RUDR LMS estimate. The dashed lines are the PROGRESS LMS estimate. The dotted lines are the least-squares fit.

with the lowest median residual is kept. Unlike our algorithm, this algorithm yields no lower bounds (with high probability) on the quality of the solution detected.

Figure 3 shows two examples where RUDR, PROGRESS, and least-squares estimation were used to perform linear regression. In these examples, we used 400 inliers and 100 outliers, both from two-dimensional normal distributions. For these experiments, 10 trials of the RUDR algorithm were considered, and 50 trials of the PROGRESS algorithm. For both examples, RUDR produces the best fit to the inliers. The least-squares fit is known to be non-robust, so it is not surprising that it fairs poorly. The PROGRESS algorithm also results in lower quality fits, since, even in 50 trials, it failed generate a solution that was close enough to the optimal solution.

8.3. Object Recognition

The application of RUDR to object recognition yields an algorithm with $O(md^k)$ computational complexity, where m is the number of model features, d is the number of data featuers, and k is the minimum number of feature matches necessary to constrain the position of the model up to a finite ambiguity in the case of error-less features in general position.

We have examined the recognition of three-dimensional objects using two-dimensional image data, for which $k = 3$ (Olson, 1997b). In each subproblem, we compute the pose for each minimal matching containing the distinguished matching using the method of Huttenlocher and Ullman (1990). We then use a multi-dimensional histogramming technique that examines each axis of the pose space separately. After finding the clusters along some axis in the pose space, the clusters of sufficient size are then analyzed recursively in the remainder of the pose space. The poses for all sets of points sharing a distinguished matching with cardinality $k - 1$ lie in a two-dimensional subspace for this case. Despite this fact, we perform the histogramming in the full six-dimensional space, since this requires little extra time and space with our histogramming method. Feature error has been treated in an ad hoc manner in this implementation through the examination of overlapping bins in the pose space. Complex images may require a more thorough analysis of the errors.

We can also apply these techniques to images in which imperfect grouping techniques have determined sets of points that are likely to derive from the same object (Olson, 1998). This allows a reduction in both the computational complexity and the rate of false positives.

Figure 4 shows an example where this approach has been applied to the recognition of a three-dimensional object. Also shown in this figure are poses generated using the alignment method (Huttenlocher and Ullman,

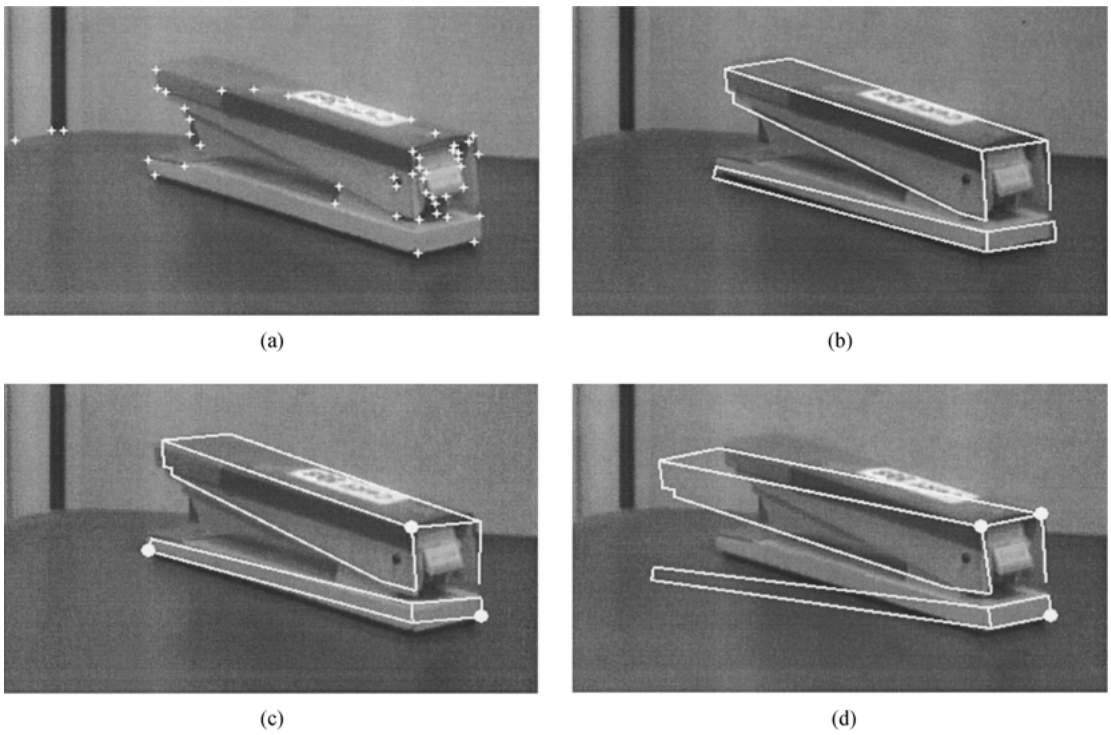

Figure 4. Three-dimensional object recognition. (a) Corners detected in the image. (b) Best hypothesis found. (c) Example pose generated using alignment. (d) Example pose generated using alignment.

1990). Figure 4(c) shows a case where the distinguished points are well distributed across of the object. Even in this case, the pose is significantly in error, with the upper right corner of the model off by over eight pixels due to corner detection error and perspective distortion. When the distinguished points are not well distributed, the error can be much worse, as is shown in Fig. 4(d). These examples illustrate the sensitivity to error that generate-and-test methods suffer from.

8.4. Motion Segmentation

In addition to the applications that we have previously studied, RUDR can be used to perform motion segmentation with any technique for determining structure and motion from corresponding data features in multiple images. In this problem, we are given sets of data features in multiple images. We assume that we know the feature correspondences between images (for example, from a tracking mechanism), but not which sets of features belong to rigid objects.

Say that we have an algorithm to determine structure and motion using k feature correspondences in i images and that there are d features for which we know the correspondences between the images. (See Huang and Netravali, 1994 for a review of such techniques.) We examine distinguished matchings of $k-1$ sets of feature correspondences between the images. Each subproblem is solved by determining the hypothetical structure and motion of each minimal matching (sets of k feature correspondences) containing the distinguished matching and then determining how many of the minimal matchings yield consistent structures for the distinguished matching and motions that are consistent with them belonging to a single object. This is repeated for enough distinguished matchings to find all of the rigidly moving objects consisting of some minimum fraction of all image features.

Our analysis for implicit matchings implies that we must examine approximately $\epsilon^{1-k} \ln \frac{1}{\gamma}$ trials to find objects whose fraction of the total number of data features is at least ϵ with a probability of failure for a particular object no larger than γ.

9. Summary

This paper has described a technique for solving model extraction and fitting problems such as recognition and regression that we have named RUDR. This approach is very general and can be applied to a wide variety problems where a model is fit to a set of data features and it is tolerant to noisy data features, occlusion, and outliers.

The RUDR method draws advantages from both the generate-and-test paradigm and from parameter space methods based on the Hough transform. The key ideas are:

1. Break down the problem into many small subproblems in which only the model positions consistent with some distinguished matching of features are examined.
2. Use randomization techniques to limit the number of subproblems that need to be examined to guarantee a low probability of failure.
3. Use clustering or parameter space analysis techniques to determine the matchings that satisfy the criteria.

The use of this technique yields two primary advantages over previous methods. First, RUDR is computationally efficient and has a low memory requirement. Second, we can use methods by which the localization error in the data features is propagated precisely, so that false positives and false negatives do not occur.

Acknowledgments

The research described in this paper was carried out in part at the Jet Propulsion Laboratory, California Institute of Technology, and was sponsored in part by the Air Force Research Laboratory at Tyndall Air Force Base, Panama City, Florida, through an agreement with the National Aeronautics and Space Administration. Portions of this research were carried out while the author was with Cornell University and with the University of California at Berkeley. A condensed version of this work appeared in the Vision Algorithms: Theory and Practice Workshop (Olson, 2000).

Notes

1. Early Hough transform strategies mapped single features into manifolds in the parameter space, but further work in Hough transforms has improved on these techniques by mapping sets of data features into points in the parameter space (Bergen and Shvaytser, 1991; Leavers, 1992; Xu et al., 1990).
2. This is not always true. For example, consider the case where the data and the model consist of sets of three-dimensional points and the transformation space is the six-dimensional space of rigid motions. Each individual match between a data point and a model point yields three constraints on the position of the model. However, a pair of such matches yields only five constraints, since the rotation around the segment joining the points is unconstrained. In this case, the additional constraint lies in the distance between the points, which must be the same in both the model and the data.

References

Alter, T.D. and Jacobs, D.W. 1998. Uncertainty propagation in model-based recognition. *International Journal of Computer Vision*, 27(2):127–159.

Ayache, N. and Faugeras, O.D. 1986. HYPER: A new approach for the recognition and positioning of two-dimensional objects. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(1):44–54.

Ballard, D.H. 1981. Generalizing the Hough transform to detect arbitrary shapes. *Pattern Recognition*, 13(2):111–122.

Bergen, J.R. and Shvaytser, H. 1991. A probabilistic algorithm for computing Hough transforms. *Journal of Algorithms*, 12:639–656.

Bolles, R.C. and Fischler, M.S. 1981. A RANSAC-based approach to model fitting and its application to finding cylinders in range data. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 637–642.

Breuel, T.M. 1992. Fast recognition using adaptive subdivisions of transformation space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 445–451.

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

Duda, R.O. and Hart, P.E. 1972. Use of the Hough transformation to detect lines and curves in pictures. *Communications of the ACM*, 15:11–15.

Edelsbrunner, H. and Souvaine, D.L. 1990. Computing least median of squares regression lines and guided topological sweep. *Journal of the American Statistical Association*, 85(409):115–119.

Faugeras, O.D. and Hebert, M. 1986. The representation, recognition, and locating of 3-d objects. *International Journal of Robotics Research*, 5(3):27–52.

Fischler, M.A. and Bolles, R.C. 1981. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24:381–396.

Grimson, W.E.L. and Huttenlocher, D.P. 1990. On the sensitivity of the Hough transform for object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(3):255–274.

Grimson, W.E.L., Huttenlocher, D.P., and Jacobs, D.W. 1994. A study of affine matching with bounded sensor error. *International Journal of Computer Vision*, 13(1):7–32.

Hough, P.V.C. 1962. Method and means for recognizing complex patterns. U.S. Patent 3069654.

Huang, T.S. and Netravali, A.N. 1994. Motion and structure from feature correspondences: A review. In *Proceedings of the IEEE*, 82(2):252–268.

Huttenlocher, D.P. and Ullman, S. 1990. Recognizing solid objects by alignment with an image. *International Journal of Computer Vision*, 5(2):195–212.

Illingworth, J. and Kittler, J. 1988. A survey of the Hough transform. *Computer Vision, Graphics, and Image Processing*, 44:87–116.

Jacobs, D.W. 1996. Robust and efficient detection of salient convex groups. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(1):23–37.

Kiryati, N., Eldar, Y., and Bruckstein, A.M. 1991. A probabilistic Hough transform. *Pattern Recognition*, 24(4):303–316.

Leavers, V.F. 1992. The dynamic generalized Hough transform: Its relationship to the probabilistic Hough transforms and an application to the concurrent detection of circles and ellipses. *CVGIP: Image Understanding*, 56(3):381–398.

Leavers, V.F. 1993. Which Hough transform? *CVGIP: Image Understanding*, 58(2):250–264.

Linnainmaa, S., Harwood, D., and Davis, L.S. 1988. Pose determination of a three-dimensional object using triangle pairs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 10(5):634–647.

Lowe, D.G. 1985. *Perceptual Organization and Visual Recognition*. Kluwer Academic: Dordrecht.

Lowe, D.G. 1987. Three-dimensional object recognition from single two-dimensional images. *Artificial Intelligence*, Boston, MA, 31:355–395.

Murakami, K., Koshimizu, H., and Hasegawa, K. 1986. On the new Hough algorithms without two-dimensional array for parameter space to detect a set of straight lines. In *Proceedings of the IAPR International Conference on Pattern Recognition*, pp. 831–833.

Olson, C.F. 1997a. An approximation algorithm for least median of squares regression. *Information Processing Letters*, 63(5):237–241.

Olson, C.F. 1997b. Efficient pose clustering using a randomized algorithm. *International Journal of Computer Vision*, 23(2):131–147.

Olson, C.F. 1998. Improving the generalized Hough transform through imperfect grouping. *Image and Vision Computing*, 16(9/10):627–634.

Olson, C.F. 1999. Constrained Hough transforms for curve detection. *Computer Vision and Image Understanding*, 73(3):329–345.

Olson, C.F. 2000. A general method for feature matching and model extraction. *Vision Algorithms: Theory and Practice*, LNCS 1883, Springer-Verlag: Berlin, pp. 20–36.

Rosenfeld, A. 1969. *Picture Processing by Computer*. Academic Press: New York, NY.

Rousseeuw, P.J. and Leroy, A.M. 1987. *Robust Regression and Outlier Detection*. John Wiley and Sons: New York.

Shapiro, S.D. 1978. Generalization of the Hough transform for curve detection in noisy digital images. In *Proceedings of the International Joint Conference on Pattern Recognition*, pp. 710–714.

Silberberg, T.M., Davis, L.S., and Harwood, D.A. 1984. An iterative Hough procedure for three-dimensional object recognition. *Pattern Recognition*, 17(6):621–629.

Stockman, G. 1987. Object recognition and localization via pose clustering. *Computer Vision, Graphics, and Image Processing*, 40:361–387.

Thompson, D.W. and Mundy, J.L. 1987. Three-dimensional model matching from an unconstrained viewpoint. In *Proceedings of the IEEE Conference on Robotics and Automation*, Vol. 1, pp. 208–220.

Xu, L., Oja, E., and Kultanen, P. 1990. A new curve detection method: Randomized Hough transform (RHT). *Pattern Recognition Letters*, 11:331–338.