A General Method for Feature Matching and Model Extraction

Clark F. Olson

Jet Propulsion Laboratory, California Institute of Technology
4800 Oak Grove Drive, Mail Stop 125-209, Pasadena, CA 91109-8099
http://robotics.jpl.nasa.gov/people/olson/homepage.html

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 selected features. Hough transform variations are somewhat less sensitive the noise, but implementations for complex problems suffer from large time and space requirements and 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. However, the subproblems are 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 apply this method to object recognition, geometric primitive extraction, robust regression, and motion segmentation.

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 information 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 [6] and the alignment method [10].

The primary drawback to generate-and-test paradigm is sensitivity to noise. Let us call the features 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*. Error in

the distinguished features causes the predicted position 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 [1, 8]. Under the assumption of a bounded error region for each of the distinguished image features, these methods can place bounds on the locations to which the undistinguished model features can be located in an image. When we count the number of undistinguished model features that can be aligned with image features (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 this method checks whether there is a model position that brings each of the undistinguished model features into alignment with image features (along with all of the distinguished features) up to the error bounds, it does not check whether there is a 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 [11, 15]. 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 [7], if the pose space analysis is not careful.

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 relevant models. We call this technique RUDR (pronounced "rudder"), for Recognition Using Decomposition and Randomization.

First, it is shown that the problem can be treated as many subproblems, each of which is much simpler than the original problem. We next discuss various methods by which the subproblems can be solved. The application of random-

ization to reduce the number of subproblems that must be examined is then 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 (i.e. with outliers or multiple models present). We discuss the application of this method to object recognition, curve detection, robust regression, and motion segmentation.

The work described here is a generalization of previous work on feature matching and model extraction [19, 20, 22]. Similar ideas have been used by other researchers. A simple variation of this method has been applied to curve detection by Murakami et al. [18] and Leavers [14]. In both of these cases, 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 [4] in the context of pose equivalence analysis. He uses a *base match* to develop an approximation algorithm for feature matching under uncertainty.

2 General Problem Formalization

The class of problems that we attack 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 or there may be multiple models present in the data. These problems can, in general, be formalized as follows.

Given:

• $\mathcal{M}$: 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 geometric primitive extraction and robust regression.

• $\mathcal{D}$: The data to match. This data consists of a set of features or measurements, $\{\delta_1, ..., \delta_d\}$, that have been extracted, for example, from an image.

• $\mathcal{T}$: The possible positions or transformations of the model. We use τ to denote individual transformations in this space.

• $A(\mathcal{M}, \mathcal{D}, \mathcal{T}, \tau, D)$: A binary-valued acceptance 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.

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 (e.g. 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, $D = \{\delta_1, ..., \delta_x\}$ 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, $D = \{\delta_1, ..., \delta_x\}$, 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 cannot 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 [19].

Example As a running example, we will consider the detection of 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 $\left|\sqrt{(x - x_c)^2 + (y - y_c)^2} - r\right| < \epsilon$. 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 half of their perimeter present in the image.

3 Approach

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 the *minimal matchings* and we 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 the 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, but other pose space analysis techniques can be used (e.g. [3, 4]).

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 method (see Section 4), 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 to subdivide 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.

Let's consider the effect of this 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 of 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.

The proof of this proposition, which follows directly from combinatorics, is sketched in [20]. This result 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.

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 improve matters, since there are more subproblems to examine. However, since we use randomization to limit the number of subproblems that are examined, we can achieve a lower computational complexity by having more simple subproblems than fewer difficult ones. On the other hand, when we reach $g = k$, the method becomes equivalent to a 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 $g = k - 1$.

Now, 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 [20], 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 no longer 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 [19].

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

4 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 of a given cardinality can be brought approximately into alignment with the model at a particular position. The simplest method is one that uses a multi-dimensional histogramming step in order to locate large clusters in the pose space. This method can be implemented efficiently in both time and space [20]. However, errors in the data cause the clusters to spread in a manner that can be difficult to handle using this technique. For complex problems, it can become problematic to detect the clusters without also detecting a significant number of false positives [7]. Alternatively, recently developed pose equivalence analysis techniques developed by Breuel [3] and Cass [4] can be applied that allow localization error to be propagated accurately. Breuel's experiments indicate that his techniques can operate in linear expected time in the number of matchings, so we can, in general, perform this step efficiently.

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 b constraints on the transformation, then a subproblem where the distinguished matchings have cardinality g considers only a $(p - gb)$-dimensional manifold of the transformation space in the errorless case. This allows us to parameterize the sub-manifold (using $p-gb$ 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 implicitly) 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 a 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 [22].

Example For circle detection, 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.) 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 containing 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 [22]. We determine if a circle is present in each trial by finely discretizing d and performing a simple Hough transform variation, 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.

5 Randomization and Complexity

A deterministic implementation of these ideas examines 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 by 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 approximately bounded by $\left(\frac{b}{d}\right)^{k-1}$. 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:

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

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

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

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 $\left(\frac{d}{f_e m}\right)^{k-1} \ln \frac{1}{\gamma}$. For each trial, we must consider matching the set of image features against each possibly matching set of model features, so the total number of distinguished matchings that are considered is approximately $\left(\frac{d}{f_e}\right)^{k-1} (k-1)! \ln \frac{1}{\gamma}$. Each explicit distinguished matching requires $O(md)$ time to process, so the overall time required is $O(md^k)$.

For implicit matches, we may 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 model or image features. Since each trial can be solved in $O(d)$ time, the overall time required is $O(d)$.

Note that the complexity can be reduced further by performing subsampling among the 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 [2]. We have not found this further complexity reduction to be necessary in our experiments. However, it may be useful when the number of image features is very large.

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 $d > 53$.

6 Comparison With Other 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 (e. g. [16]). RUDR yields a superior computational complexity requirement for this case. When randomization is applied to generate-and-test techniques, the computation 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 [1, 8], 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. A more serious problem is that, while the generate-and-test techniques that propagate errors correctly 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) [3, 4]. 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 [20].

Randomization has been previously applied to Hough transform techniques [2, 13, 14, 24]. However, in previous methods, randomization has been used in a different manner than it is used here. While RUDR examines all of the data in each of the subproblems, previous uses of randomization in Hough-based methods subsample the overall data examined, causing both false positives and false negatives to occur as a result. While false negatives can occur due to the use of randomization in the RUDR approach, the probability of such an occurrence can be set arbitrarily low.

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 reap 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 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, [12, 17]). In cases where models can be identified by purely data-driven methods, such techniques are likely to be faster than the techniques described here. However, work has shown the even imperfect feature grouping methods can improve both the complexity and the rate of false positives in the RUDR method [21].

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 is 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 benefits will still apply.

7 Applications of RUDR

RUDR has been applied to several problems. We review the important aspects of these applications here and discuss additional areas where RUDR can be applied.

7.1 Extraction of Geometric Primitives

The Hough transform is a well known technique for geometric primitive extraction [11, 15]. 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 [22].

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 distinguished 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 errors into segments of this curve for the case of lines and circles [22]. $O(d)$ time and space is required for curve detection with these techniques, where d is the number of data points extracted from the image.

Figure 1 shows the results of using RUDR to detect circles in a binary image of an engineering drawing. The results are very good, with the exception of circles found with a low threshold that are not perceptually salient. However, these circles meet the acceptance criterion specified, so this is not a failure of the algorithm.

The image in Figure 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 the same probability of examining a trial where the distinguished features belong to some circle, but will still miss circles due to the error in the features.

7.2 Robust Regression

RUDR can be applied to the problem of finding the least-median-of-squares (LMS) regression line. The most commonly considered problem is to fit a line to points in the plane. We apply RUDR to this problem by considering a series of distinguished points in the data. A single distinguished point is examined in each trial (since only two 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.

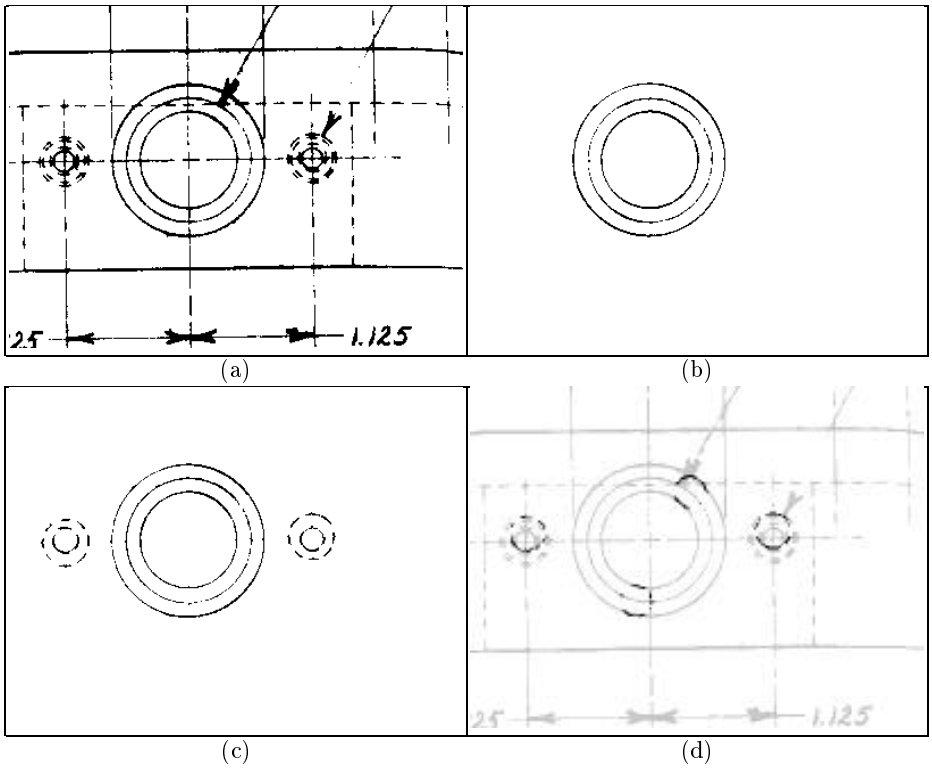


Fig. 1. Circle detection. (a) Engineering drawing. (b) Circles found comprising 4% of the image. (c) Perceptually salient circles found comprising 0.8% of the image. (d) Insalient circles found comprising 0.8% of the image.

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 [19]. Now, at least half of the data points must lie no farther from the optimal regression line than the optimal median residual (by definition). Each trial thus 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 [19]. 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.

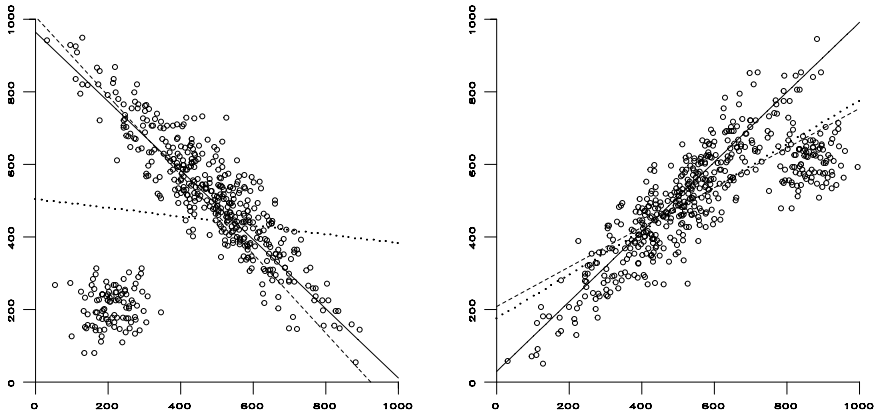


Fig. 2. Robust regression examples. The solid lines are the RUDR LMS estimate. The dashed lines are the PROGRESS LMS estimate. The dotted lines are the least-squares fit.

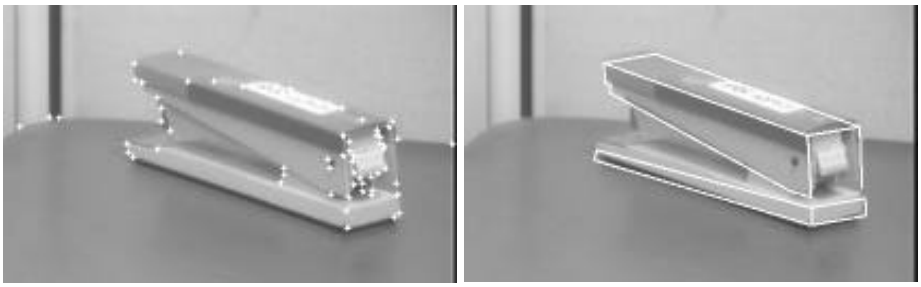
The complexity of our method is superior to the best known exact algorithms for this problem [5]. The PROGRESS algorithm [23] is a commonly used approximation algorithm for LMS regression that is based on the generate-and-test paradigm. It requires $O(n)$ time. However, unlike our algorithm, this algorithm yields no lower bounds (with high probability) on the quality of the solution detected.

Figure 2 shows two examples where RUDR, PROGRESS, and least-squares estimation were used to perform regression. In these examples, there were 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 cases, 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 has difficulty, since, even in 50 trials, it does not generate a solution very close to the optimal solution.

7.3 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 features, and k is the minimal number of feature matches necessary to constrain the position of the model up to a finite ambiguity in the case of errorless features in general position.

For recognizing three-dimensional objects using two-dimensional image data, $k = 3$. In each subproblem, we compute the pose for each minimal matching containing the distinguished matching using the method of Huttenlocher and Ullman [10]. We then use a multi-dimensional histogramming technique that



(a) (b)

Fig. 3. Three-dimensional object recognition. (a) Corners detected in the image. (b) Best hypothesis found.

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 [20]. The poses for all sets of points sharing a distinguished matching of cardinality $k-1$ lie in a two-dimensional subspace for this case. Despite this, we perform the histogramming in the full six-dimensional space, since this requires little extra time and space with this 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 [21]. This allows a reduction in both the computational complexity and the rate of false positives. Figure 3 shows an example where this approach has been applied to the recognition of a three-dimensional object.

7.4 Motion Segmentation

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 (e.g. from a tracking mechanism), but not which sets of features belong to coherent 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 [9] 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 (k sets of 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.

8 Summary

This paper has described a technique that we have named RUDR for solving model extraction and fitting problems such as recognition and regression. 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 by 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 the University of California at Berkeley.

References

1. T. D. Alter and D. W. Jacobs. Uncertainty propagation in model-based recognition. *International Journal of Computer Vision*, 27(2):127–159, 1998.
2. J. R. Bergen and H. Shvaytser. A probabilistic algorithm for computing Hough transforms. *Journal of Algorithms*, 12:639–656, 1991.

3. T. M. Breuel. Fast recognition using adaptive subdivisions of transformation space. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 445–451, 1992.
4. T. A. Cass. Polynomial-time geometric matching for object recognition. *International Journal of Computer Vision*, 21(1/2):37–61, January 1997.
5. H. Edelsbrunner and D. L. Souvaine. Computing least median of squares regression lines and guided topological sweep. *Journal of the American Statistical Association*, 85(409):115–119, March 1990.
6. M. A. Fischler and R. C. Bolles. Random sample consensus: A paradigm for model fitting with applications to image analysis and automated cartography. *Communications of the ACM*, 24:381–396, June 1981.
7. W. E. L. Grimson and D. P. Huttenlocher. On the sensitivity of the Hough transform for object recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 12(3):255–274, March 1990.
8. W. E. L. Grimson, D. P. Huttenlocher, and D. W. Jacobs. A study of affine matching with bounded sensor error. *International Journal of Computer Vision*, 13(1):7–32, 1994.
9. T. S. Huang and A. N. Netravali. Motion and structure from feature correspondences: A review. *Proceedings of the IEEE*, 82(2):252–268, February 1994.
10. D. P. Huttenlocher and S. Ullman. Recognizing solid objects by alignment with an image. *International Journal of Computer Vision*, 5(2):195–212, 1990.
11. J. Illingworth and J. Kittler. A survey of the Hough transform. *Computer Vision, Graphics, and Image Processing*, 44:87–116, 1988.
12. D. W. Jacobs. Robust and efficient detection of salient convex groups. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 18(1):23–37, 1996.
13. N. Kiryati, Y. Eldar, and A. M. Bruckstein. A probabilistic Hough transform. *Pattern Recognition*, 24(4):303–316, 1991.
14. V. F. Leavers. 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, November 1992.
15. V. F. Leavers. Which Hough transform? *CVGIP: Image Understanding*, 58(2):250–264, September 1993.
16. R. J. Lipton and R. E. Tarjan. Applications of a planar separator theorem. *SIAM Journal on Computing*, 9(3):615–627, 1980.
17. D. G. Lowe. *Perceptual Organization and Visual Recognition*. Kluwer, 1985.
18. K. Murakami, H. Koshimizu, and K. Hasegawa. 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*, pages 831–833, 1986.
19. C. F. Olson. An approximation algorithm for least median of squares regression. *Information Processing Letters*, 63(5):237–241, September 1997.
20. C. F. Olson. Efficient pose clustering using a randomized algorithm. *International Journal of Computer Vision*, 23(2):131–147, June 1997.
21. C. F. Olson. Improving the generalized Hough transform through imperfect grouping. *Image and Vision Computing*, 16(9-10):627–634, July 1998.
22. C. F. Olson. Constrained Hough transforms for curve detection. *Computer Vision and Image Understanding*, 73(3):329–345, March 1999.
23. P. J. Rousseeuw and A. M. Leroy. *Robust Regression and Outlier Detection*. John Wiley and Sons, 1987.
24. L. Xu, E. Oja, and P. Kultanen. A new curve detection method: Randomized Hough transform (RHT). *Pattern Recognition Letters*, 11:331–338, May 1990.