

RUDR: A New Paradigm for Model Extraction and Fitting

Clark F. Olson *

Jet Propulsion Laboratory, California Institute of Technology

4800 Oak Grove Drive, MS 107-102, Pasadena, CA 91109

<http://robotics.jpl.nasa.gov/people/olson>

Abstract

This paper describes a new algorithmic paradigm for solving problems where a model is extracted from or fit to data. This paradigm has numerous applications, particularly in computer vision. The new paradigm is called RUDR (pronounced “ruder”) for Recognition Using Decomposition and Randomization. The main components of the paradigm are the decomposition of the problem into many smaller subproblems, the use of randomization to limit the number of subproblems that must be examined to maintain high accuracy, and the use of pose space analysis techniques to solve each subproblem. We show that, in general, this paradigm has advantages over previous methods. The application of these techniques to object recognition, extraction of geometric primitives, robust regression, and motion segmentation is discussed.

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 paradigm is to generate a hypothetical solution using the minimal amount of information and then test the quality of the solution. This is repeated for many hypothetical solutions and the best solution(s) are kept if they meet some criterion. Examples of this technique include RANSAC [7] and the alignment method [12]. A competing paradigm based on the Hough transform also generates hypothetical 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

*The work described in this paper was started at Cornell University, sponsored by ARPA under ARO contract D AAI104-93-C-0052 and by the National Science Foundation under grant IRI-9057928, and was completed at the Jet Propulsion Laboratory, California Institute of Technology, sponsored by the Wright Laboratory at Tyndall Air Force Base, Panama City, Florida, through an agreement with the National Aeronautics and Space Administration. Reference herein to any specific commercial product, process, or service by trade name, trademark, manufacturer, or otherwise, does not constitute or imply its endorsement by the United States Government, or the Jet Propulsion Laboratory, California Institute of Technology.

through a 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*. Examples include variants of the Hough transform (see [13, 16]) and pose clustering (e.g. [25]).

In this paper, a new paradigm that generalizes and extends previous work on model fitting [20,22] is presented. This paradigm draws ideas and advantages from both the generate-and-test paradigm and the Hough-based paradigm. While the underlying matching technique used is to examine pose space for sets of consistent matches like a Hough-based method, the use of problem decomposition techniques allows this class of algorithms to be viewed also as generate-and-test algorithms, where the initial matches consist of data that is insufficient to constrain the model position to a finite set of possibilities (even for errorless data) and these initial matches are tested using pose space analysis techniques in the subspace of the pose space that is consistent with the initial matches.

The basic steps of the new paradigm are as follows. First, the problem is decomposed into many small subproblems. (A method for performing this decomposition, in general, is given here.) Second, randomization is used to select a subset of subproblems to be examined while maintaining a low rate of failure. Third, the subproblems are solved using some parameter space analysis technique. We call this new paradigm RUDR (pronounced “rudder”), for Recognition Using Decomposition and Randomization.

In general, these techniques test fewer hypotheses than previous generate-and-test methods, with a test phase that is no more complex. In addition, the decomposition of the problem allows each of the subproblems to examine a much smaller parameter space than the original problem and this often allows the error inherent in localization procedures to be propagated accurately and efficiently in the matching process.

This algorithmic paradigm has a tremendous 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 an example application of these techniques, where parallel lines are detected in edge images, in detail. Several practical issues are examined with respect to this application. Previous application of the RUDR paradigm to object recognition [20] and curve detection [22], as well as methods by which RUDR can be applied to robust, regression and motion segmentation, are summarized.

There has been significant, previous work combining Hough transform techniques and randomization [2, 14, 15, 26]. Of particular interest is the work of Leavers [15], who, in addition, considered subproblems where a single point was used to place a constraint on the allowable transformations. However, this does not achieve the full decomposition of the problem and does not handle localization error robustly. RUDR yields considerable improvement over this method.

2 A 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, $\{\delta_1, \dots, \delta_d\}$, that have been extracted, for example, from an image. For simplicity, we assume that all of the data features are of a single type, but this restriction can be easily removed.

- **\mathcal{T}** : The possible positions or transformations of the model. This pose space is a (possibly unbounded) parameter space in which the model must lie. We denote individual transformations in this space by τ .

- **$A(\mathcal{M}, \mathcal{D}, \mathcal{T}, \tau, D)$** : An acceptance criterion that determines whether a transformation, τ , satisfactorily brings the model into agreement with a set of data features, D . We allow this criterion to be a function of the 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. The stipulation that *these* should be maximal sets of data features means that a set of data features should not be reported if it is fully contained within another set that is reported.

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

A particularly useful acceptance criterion is lined on bounding the fitting error between the model and the data. Let $C(\mathcal{M}, \delta, \tau)$ be a function of the model, a particular data feature, and a model position that determines whether the model at the specified position fits the data feature (e.g. up to a bounded error). We define $C(\mathcal{M}, \delta, \tau) = 1$ if the criterion is satisfied, and $C(\mathcal{M}, \delta, \tau) = 0$, otherwise. A set of data features, $D = \{\delta_1, \dots, \delta_x\}$, is said to be brought, into alignment 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, $D = \{\delta_1, \dots, \delta_x\}$, 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.

This bounded error criterion can incorporate not only the location of features, but in

addition, other local information such as curvature, color, texture, and contrast. Weighting of the features can be easily added. It cannot incorporate global information, such as mean-square-error or least-median-of-squares. However, RUDR is not restricted to using this bounded error criterion. Indeed, RUDR has been applied to least-median-of-squares regression with excellent results.

We note that this bounded error criterion, as stated, does not allow the exclusion of multiple data features matching a single model feature in discrete models. However, this exclusion is easy to achieve in practice, if desired. This also yields a technique that maximizes the number of model features that are matched by data features, while excluding data features from matching multiple model features.

3 Decomposition into subproblems

Let us call a set of matches between data features and the model a *matching*. The generate-and-test strategy 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 . It should be noted that we consider, in particular, two types of model. One type of model is 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*.

3.1 RUDR approach

In the generate-and-test paradigm, the model positions generated using the minimal matchings are tested by comparing the model position to the data to determine if it results in a good fit of the model to the data. 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 many other pose space analysis techniques can be used (e.g. [3, 6, 11, 13, 17]).

The approach that we take in the RUDR paradigm 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 paradigm, but unlike previous Hough-based methods, the problem is subdivided into many small subproblems, each of which examines a subset of the minimal matchings. This decomposition is achieved by considering sets of distinguished matches between data features and the model. We call these sets of matches *distinguished matchings* and the data features that are matched in such a matching are called a *distinguished set*. Each subproblem considers only those model positions that are consistent with the distinguished matching.

The cardinality of the distinguished matchings must be smaller than the cardinality of minimal matchings for the RUDR paradigm to be useful. We will thus have a set of minimal matchings that includes each distinguished matching. The restriction of each subproblem to those positions consistent with the distinguished matching is achieved by considering only those minimal matchings that include the distinguished matching in the subproblem.

This decomposition of the problem allows these techniques to be viewed as a new 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.

We should note that if some special structure is required in the minimal matching to suitably constrain the model position (such as colinearity), we must take care to select suitable distinguished matchings. However, such special structure is not required for any of the applications examined here.

3.2 Equivalence of formulations

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:

$$\sum_{\delta \in \mathcal{D}} C(\mathcal{M}, \delta, \tau) \geq x$$

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:

$$\sum_{\{\delta_1, \dots, \delta_k\} \subset \mathcal{D}} \left(\prod_{i=1}^k C(\mathcal{M}, \delta_i, \tau) \right) \geq \binom{x}{k}$$

3. For any distinguished set, $\mathcal{G} = \{J_0, \dots, J_g\}$, that is brought into alignment with the model up to the error criterion by τ , there are $\binom{x-g}{k-g}$ minimal matchings of distinct sets of data features containing the distinguished set that are brought into alignment, up to the error criterion by τ .

$$\sum_{\{\delta_{g+1}, \dots, \delta_k\} \subset \mathcal{D} - \mathcal{G}} \left(\prod_{i=1}^k C(\mathcal{M}, \delta_i, \tau) \right) \geq \binom{x-g}{k-g}$$

Proof :

The proof of this proposition follows directly from combinatorics. We prove (a) Statement 1 implies Statement 2, (b) Statement 2 implies Statement 3, and (c) Statement 3 implies Statement 1. Thus, the statements are 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 $L(\mathcal{M}, \delta_i, \tau) = 1$. (If there were $y < x$ such features then we could only form $\binom{y}{k}$ minimal matchings satisfying Equation (1).) Choose any subset, \mathcal{G} , of these matches of cardinality g . Form the $\binom{x-g}{k-g}$ subsets of 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 . \square

The implication of this result, is that, if we are interested in finding model positions that bring x data features into alignment with the model, it is equivalent to finding $\binom{x}{k}$ sets of k data features that, are brought into alignment with the model, and it also is equivalent to finding a sufficient number of such sets that share some distinguished set that are brought into alignment. Thus, as long we examine at least one distinguished set that belongs to each of the matchings that should be reported, the strategy of decomposing the problem into subproblems yields equivalent results to examining the original problem.

Note that this theorem is stated in terms of distinguished sets, but for explicit matches, the subproblems examine distinguished matchings, where the match for each data feature is given. For each distinguished set that is considered in this case, the matching algorithm examines all possible matchings for the distinguished set to ensure that we examine a correct distinguished matching if the distinguished set belongs to the model.

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 not efficient. When we use an approximation algorithm, Proposition 1 is no longer precisely correct. Just as the use of an approximation algorithm introduces the possibility that we do not find the best (or all) solutions, it introduces the possibility that examining subsets of minimal matchings, as described above, does not yield the same results as examining the full set. However, empirical evidence suggests that the examination of these subproblems yields superior results when an approximation algorithm is used [20, 23].

We can also use these techniques with acceptance criteria other than the bounded error

criterion. With other criteria, the theorem is also, usually, only approximately correct, but if an approximation algorithm is used to detect good matchings, it often yields good results. For example, an application of these ideas to least-median-of-squares regression has yielded an approximation algorithm that was provably accurate with high probability, while previous approximation algorithms do not have this property [18].

3.3 Error sensitivity

Hough-based methods have been criticized for their error sensitivity [8], but, from this analysis, it is clear that, if exact matching methods are used to solve the RUDR subproblems (or even the original problem), then no other method can achieve superior performance with respect to the bounded error criterion, since we report exactly those matchings that satisfy the acceptance criterion.

In fact, Hough-based methods are, in general, superior to generate-and-test methods with respect to localization error. Initially, generate-and-test methods implicitly assumed that there was no localization error in the minimal matching used to determine the hypothetical mode position [7, 12]. In this case, localization error causes correct matches to be missed. More recently methods have been developed to propagate localization error in the testing step [1], but these methods do not ensure global consistency of the matches. These techniques result in a significant number of matches occurring due to random accumulation of possible matches [9]. When an accurate pose analysis technique is used, a Hough-based method can propagate localization error without resulting in false positives¹.

3.4 optimal matching 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. This is because the larger the distinguished matching is, the more constraint we have on the position of the model, and thus the simpler the subproblems are to solve. On the other hand, as the cardinality of the distinguished matching is increased, the number of subproblems that must be examined to maintain a low rate of failure also increases.

Our analysis below in Section 5 shows that the decrease in complexity caused by increasing the cardinality of the distinguished matching is greater than the increase caused by a greater number of subproblems. Note though, that no matter how large the cardinality of the distinguished matching is, we must always test each matching containing the distinguished matching and at least one additional feature matching to determine the quality of the distinguished matching. Since it is not particularly useful to examine matchings that are larger than the minimal matchings, the optimal cardinality of the distinguished matching is

¹Here we define a false positive as a match that does not meet the acceptance criterion, but that is reported, rather than a match that is not an instance of the model. Thus no match that meets the acceptance criterion is considered to be a false positive.

$k - 1$. We thus consider distinguished matchings with cardinality $g = k - 1$ for the balance of this paper.

4 Solving the subproblems

Now, we must use some method to solve each of the subproblems that are examined. We can, in fact, use any method from the literature that determines how many matchings of a given cardinality can be brought into alignment with the model at a particular position. This includes the standard histogramming or clustering methods used in the 1-ough transform and pose clustering and also recently developed pose equivalence analysis techniques that allow localization error to be propagated accurately [3, 6]. Note that the histogramming techniques require linear time in the number of matchings examined and that Breuel's experiments indicate that his techniques can operate in linear expected time in the number of matchings [3], so we can, in general, perform this step efficiently.

Each subproblem in RUDR can be solved efficiently, not only because fewer matchings are considered than in the full problem, but also because a small portion of the parameter space is examined. Each subproblem must consider only the positions of the model that bring the distinguished matching into alignment up to the error criterion.

If it is assumed that there is no error in the features in the distinguished matching, then each subproblem needs to consider 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 examines only a $(p - gb)$ -dimensional sub-manifold of the transformation space in the errorless case².

Now, since each subproblem is concerned with only a sub-manifold of the transformation space, we can 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 good matchings by histogramming or determining which segments on the curve are consistent with many minimal matchings [22].

Note that this formulation of the subproblems as considering a sub-manifold of the transformation space allows the use of methods that determine how many individual features are brought into alignment with the model up to the error criterion, since we have removed the portions of the transformation space that do not agree with the distinguished matching being brought into alignment. Some examples of such methods are the Fast Hough Transform [17] and the multi-resolution parameter space search of Huttenlocher and Rucklidge [11].

When we consider the localization error in the features of the distinguished matching, the subproblems must (at least implicitly) consider a larger space than the manifold described above. The subproblems are still much easier to solve, since the transformations usually do

²There are exceptions to this rule, but they do not cause serious problems [24], and we do not consider them here.

not stray far from the manifold that they must lie on in the errorless case. 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].

5 Randomization and complexity

A deterministic implementation of these ideas that examines each possible distinguished matching with the appropriate cardinality requires $O(n^k)$ running time [24], 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 sets that are part of each of the large consistent matchings that we are seeking. We thus find each maximal matching many times, once for each distinguished set that is contained in the maximal matching. We can take advantage of this redundancy through the use of a randomization technique to limit the number of subproblems that we must examine while maintaining a low probability of failure. The usual method by which this is accomplished is to assume that some minimum number or some minimum fraction of the data features belong to the model that we are considering. If the number or fraction of data features that belong to the model is not large enough, then we allow the algorithm to fail. This is reasonable since the model is very difficult to find if the number of data features that belong to it is small and the model is not perceptually significant if the fraction of data features that belong to it is small.

5.1 Implicit matchings

Let's first consider the case of implicit matchings (i.e. where the model is a parameterized manifold). We use the assumption that some fraction of the data features belong to the model and determine the number of random distinguished sets that must be examined to achieve a fixed probability of examining at least one distinguished set that completely belongs to the model. Let ϵ be the minimum fraction of data features that must belong to the model to maintain a low rate of failure. For a distinguished set with cardinality $g < k$, the probability that all of the data features belongs to the model is (since we sample the data features without replacement) at least:

$$p_1 \geq \prod_{i=1}^g \frac{\epsilon d - i + 1}{d}$$

As the number of data features, d , increases, this probability approaches ϵ^g , and is thus asymptotically independent of d . The probability that t trials fail to select a correct distinguished set can be bounded by $p_t \leq (1 - p_1)^t$. We can now select an arbitrarily small probability of failure, γ , and determine the number of trials necessary to ensure that the

probability of failure is no larger than γ . This yields:

$$t \geq \frac{\ln \gamma}{\ln(1 - p_1)}$$

If we approximate p_1 by ϵ^g and further use the approximation: $\ln(1 + \alpha) \approx \alpha$ (for small α), then the number of trials we must examine is approximately $\epsilon^{-g} \ln \frac{1}{\gamma}$. Note that this is independent of d . The number of trials that must be examined is $O(1)$ for constant γ, ϵ and g . Each trial requires $O(d^{k-g})$ time (for $g < k$), since there are $\binom{d}{k-g}$ minimal matchings that include the distinguished matching, and each trial can be performed in linear time in the number of minimal matchings that are examined. We thus want g to be as large as possible, but we must have $g < k$, so we use $g = k - 1$ as mentioned previously in Section 3.4, and we achieve a running time of $O(d)$. A similar analysis for generate-and-test methods also yields a running time that is $O(d)$. However, RUDR still requires fewer trials (by a factor of approximately $\frac{1}{\epsilon}$).

5.2 Explicit matchings

For the case of explicit matchings, we use the assumption that some minimum fraction, f , of the model features appear in the data (i.e. that at least $\lfloor fm \rfloor$ data features belong to the mode). If the fraction of model features appearing in the image is below f , we allow the algorithm to fail. The probability that a sample of $g < k$ data features comes entirely from the model is at least:

$$p_1 = \prod_{i=1}^g \frac{fm - i + 1}{d}$$

If we require the probability of not examining a single sample of data features entirely from the model to be no more than γ , we again have $t \geq \frac{\ln \gamma}{\ln(1 - p_1)}$. Approximating p_1 by $(\frac{fm}{d})^g$ and using the approximation $\ln(1 + \alpha) \approx \alpha$ yields:

$$t \geq \left(\frac{d}{fm} \right)^g \ln \frac{1}{\gamma}$$

Note that for each of the sets of data features that are sampled, we must consider matching the data features against each possible set of model features of the appropriate cardinality. For each trial, we thus consider $O(m^g)$ sets of model features and overall we consider $O(d^g)$ distinguished matchings since the number of trials is $O((\frac{d}{m})^g)$ for constant γ and f . Now, for each distinguished matching, we examine the $O((md)^{k-g})$ minimal matchings that include the distinguished matching and since the pose analysis step can be performed in linear time, the total time that is required is $O(m^{k-g} d^k)$ for $g < k$. Once again, we want g to be as large as possible, which is $g = k - 1$, and this yields our $O(7 \text{ talk})$ algorithm.

A similar analysis for generate-and-test methods yields a computational complexity of $O(md^{k+1})$, if each of the $O(md)$ additional feature matches is tested for each distinguished matching. For many problems, this can be improved if we are only concerned with the

number of model features that are matched by an image feature. However, this does not allow the exclusion of the case where multiple model features match a single image feature, which RUDR can achieve without increasing the complexity.

5.3 Intelligent feature selection

If some method exists to select features intelligently, we can incorporate it into RUDR, by using this method to select distinguished matchings that are more likely to yield good results. An example is the use of perceptual grouping techniques to select distinguished matches that are likely to belong to the same object [21]. Such methods can be used both to reduce the complexity of algorithms in the RUDR paradigm and to reduce the likelihood that a large matching is detected that does not correspond to an instance of the model being sought.

6 Detailed application example

This section walks through the application of RUDR to an example problem, the detection of parallel lines in an image, and includes discussion of the algorithm design and implementation stages of the application and consideration of the practical issues involved in such an application.

The first step in applying the RUDR paradigm to a particular problem is to determine the model and the type of data features that will be used. For the detection of parallel lines, our model is simply a parameterized form of a pair of parallel lines. We use the (p, θ) parameterization for the lower line (i.e. $p = x \cos \theta + y \sin \theta$), along with a parameter, d , describing the perpendicular distance to the second line. We could use essentially any set of two-dimensional points as our data. We focus, however, on the edges detected in a digital image and we choose to use oriented edge points as our data, since a local orientation (such as the gradient) is available from most edge detectors. (We use a version of the Canny edge detector [5] to generate the data.)

Next, we must consider the number of data features necessary to constrain our model. In the noiseless case, two oriented points overconstrain the position of a pair of parallel lines (assuming one point lies on each line), while one point is clearly insufficient. Our minimal matchings thus consist of pairs of points. Since this overconstrains the model, we can eliminate many pairs of points prior to the pose analysis step, as no model position is consistent with them.

Our strategy is now to randomly sample single points from the edge map of an image to be the distinguished set (or simply the distinguished point, in this case). For each such sampled point, we consider pairing it with each other possible point and determine which pairs could feasibly belong to a pair of parallel lines. For the ones that could, we perform some pose space analysis to determine if there is a sufficient number to output, a pair of parallel lines in the image. Note that we must not only perform this pose space analysis to determine the potential location of the other line, we must also maintain a count of the

number of points parallel with the distinguished point, to ensure that both lines are present in the image.

Now, how should we perform our pose space analysis for each distinguished point? Here, we choose to use a bounded error criterion. If we know (or can estimate) the distribution of errors, the error bounds can be chosen by taking those that capture a certain percentile (say 95%) of all errors, or they can be chosen empirically. We have empirically chosen error bounds of 1.0 pixel in location and $\frac{\pi}{16}$ radians in orientation. We can now project the model position (if any) consistent with a particular pair of oriented points onto the manifold consistent the distinguished point in the errorless case. Since the distinguished point fully constrains one of the parallel lines (in the errorless case), this manifold is a curve in the three-dimensional parameter space that can be parameterized by the distance! from the line given by the distinguished point. Note that this distance may be negative, since we do not know which of the lines, if any, the distinguished point lies on.

To perform this projection for any pair of points that are examined, we consider the set of orientations that are consistent with both of the points up to the error boundaries. Each orientation in this set yields a distance between the pair of parallel lines. We obtain a range of such distances by examining the minimum and maximum consistent orientation. Note that if the set of consistent orientations contains the orientation of the segment connecting the points, the lower bound on this distance is the distance between the points. To this range we must also add the error possible in the position of the points. This final range is our projection of the pose space that is consistent with the pair of points onto the curve in the parameter space yielded by the distinguished point.

We use these projections to count the number of points that are consistent with parallel lines whose distances from the line given by the distinguished point are between discrete intervals. We chose the intervals to have a width of one pixel. An array of counters of size $2d_{max} + 1$ is allocated to store values ranging from d_{max} to $-d_{max}$, where d_{max} is the maximum distance between any pair of parallel lines (and can certainly be bounded by the distance between two opposing corners of the image). Now for each range that is yielded by the projections, as described above, we increment the counters that correspond to values covered by the projection. After the distinguished point has been matched with each additional point, we look for peaks in the counter array. Note that this yields a conservative technique that provides an upper bound on the number of pixels that can belong to a model up to the bounded error.

We also use the error bounds to determine if each additional point examined lies on the same parallel line as the distinguished point. If both points have orientations that are consistent with being perpendicular to the segment between the points (in which we must also consider the localization error), then the points are considered to be parallel up to the localization error and the counter maintaining this count is incremented.

Finally, we must have some criterion determining which parallel lines are output. In our experiments, we have simply output the pair of parallel lines for which we obtain the maximum geometric mean between the number of points lying on the same line as the distinguished point and the number of points lying on a parallel line. Note that we could,

without additional work, report all of the parallel lines that surpass some threshold with respect, to our quality measure.

In order to improve the efficiency of the method, we employ the concept of a radius of coherence for each point (see [4] for a similar usage). With this technique, we assume that only points' within some fixed radius of the distinguished point are likely to contribute to the model and we eliminate the remaining points from consideration in each subproblem. In our experiments, we have used a radius of coherence of 100 pixels.

Figure 1 shows the results of applying this method to four example images. The first three examples show greyscale versions of scanned color images of pieces of unexploded munitions. The last example is a thermal image of an inert example of the same type of munitions. In each case, the best set of parallel lines found indicates the position of the bomb. In these experiments, we have assumed that the best pair of parallel lines comprised at least 4% of the data and we allowed a probability of failure to detect the best pair of parallel lines of 1%. With these parameters, the number of trials necessary is:

$$t = \left\lceil \frac{\ln .01}{\ln(1 - .04)} \right\rceil = 113$$

From the analysis of section 5, we can see that the complexity of the techniques is $O(n)$, where n is the number of edge pixels. On these edge images, consisting of between 2394 and 4119 edge pixels, these techniques required between 1.17 and 1.76 seconds per image on a SPARCstation™20.

We note that as the percentage of outliers increases, or number of pixels belonging to the model decreases, the number of trials that is necessary in this application increases slowly (i.e. with the logarithm of the fraction of pixels belonging to the model). This assumes that we can set some lower bound on the fraction of pixels that belong to model. In practice this is not difficult, and this bound can be set either empirically or theoretically by examining the likelihood of finding a hypothesis that meets the acceptance criterion due to the random accumulation of data features.

7 Applications of RUDR

RUDR has been applied to several additional 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 to extract curves and surfaces from data by mapping sets of data features into manifolds in the parameter space and then searching for peaks in the parameter space. The application of the RUDR to this problem improves the efficiency of the techniques, allows the localization error to be propagated accurately, and reduces the amount of memory that is required [22].

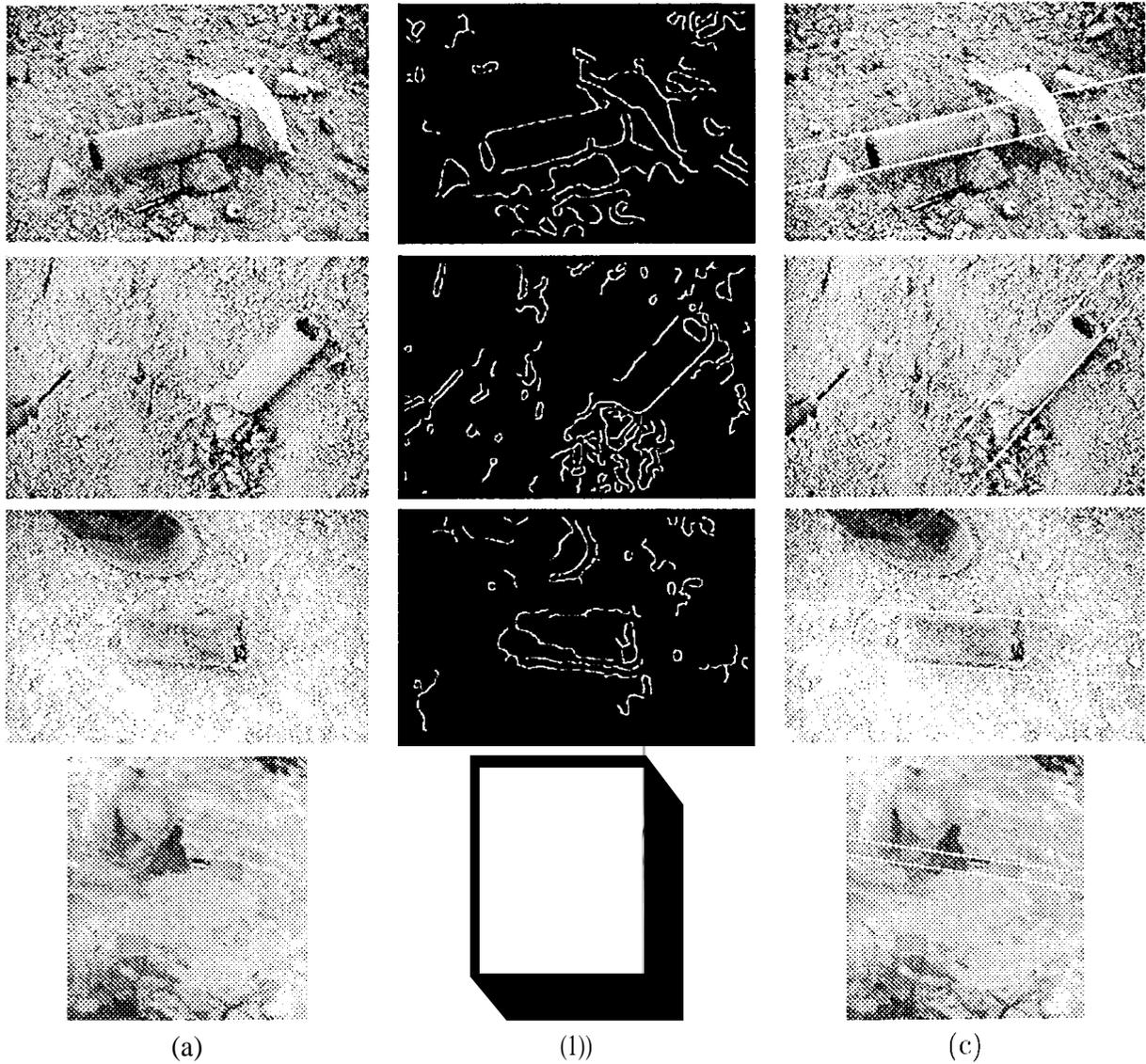


Figure 1: Parallel line detection examples. (a) Original image. (b) Edges detected in image. (c) Parallel lines detected overlaid on original image.

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]. If we then look for sections on the curve where many of these segments overlap, this yields a conservative algorithm that finds all cases where the curve fits the points up to the error criterion, but may also find cases where some of the points are not quite fit up to the error criterion. This method allows each subproblem (corresponding to a particular distinguished matching) to be solved efficiently and propagates the localization error without introducing many false positives. $O(d)$ time and space is required for curve detection with these techniques, where d is the number of points present in the image.

Figure 2 shows the results of using the RUDR paradigm to detect circles in a binary image of an engineering drawing. All of the large circles are found when the circles reported are required to comprise at least 4% of the image. When this fraction is reduced to 0.8%, not only are the pairs of dashed circles that are perceptually salient found, but also several circles that are not perceptually salient. These additional circles that are found satisfy the acceptance criterion, so this is not a failure of the algorithm. Such insalient circles are difficult to eliminate without the use of additional information. In this example, the implementation found only one circle at the locations where concentric dashed circles were very close together. The circle found consists of the top half of one of the circles and the bottom half of the other. This is due both to their proximity and imperfections in the circles.

The robustness of this technique for line detection has been compared against other methods in a large number of synthetic images. Four methods were 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 problem decomposition.
4. The standard Hough transform.

Figure 3 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 distracters were added 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. Interestingly, the RUDR paradigm fares poorly when localization error is not propagated carefully. It thus appears crucial when using the RUDR paradigm to propagate the localization into the parameter space. Further details can be found in [19].

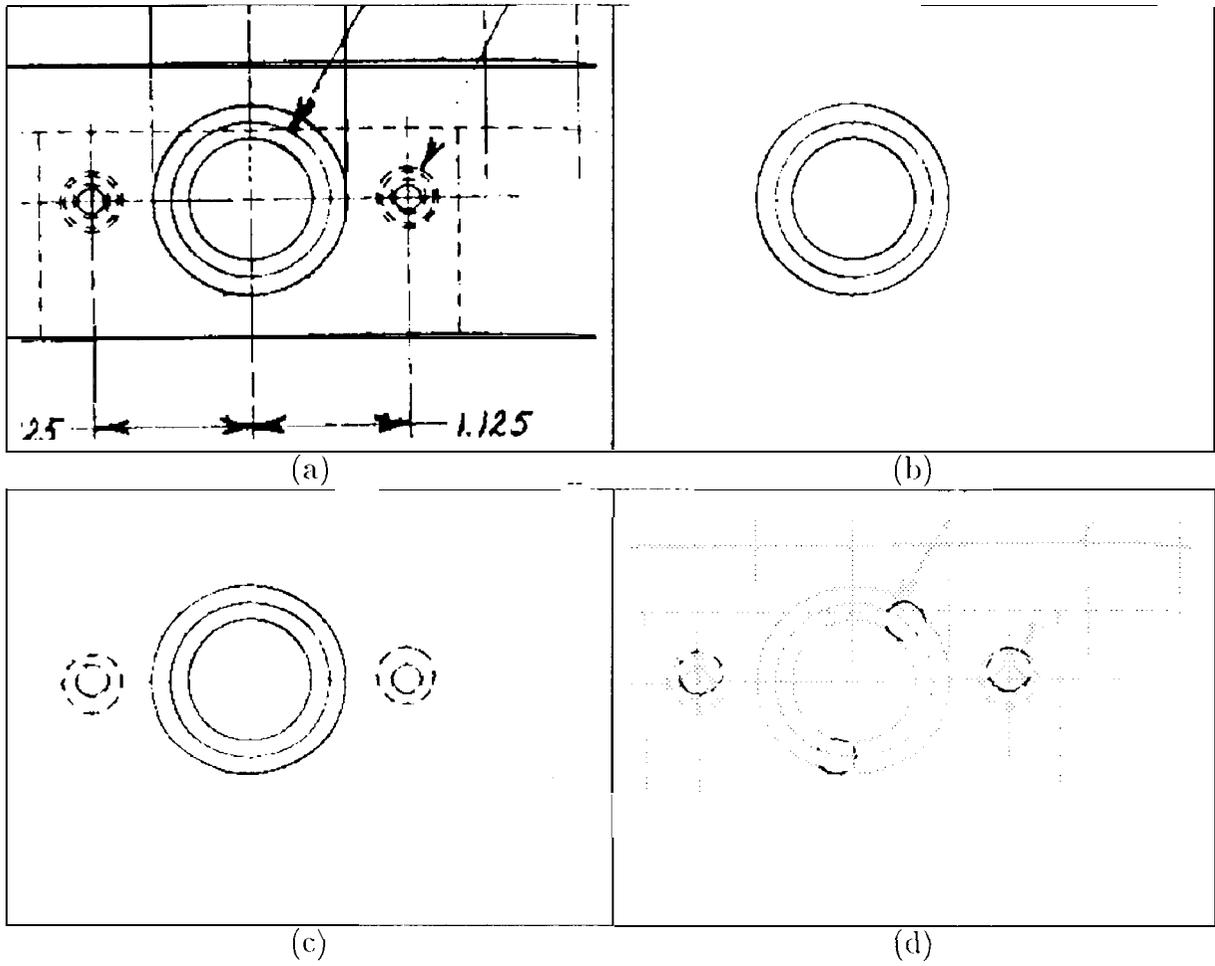


Figure 2: Circle detection using RUDR. (a) Engineering drawing. (b) Circles found that comprised 4% of the image. (c) Perceptually salient circles found that comprised 0.8% of the image. (d) Insalient circles found that comprised 0.8% of the image.

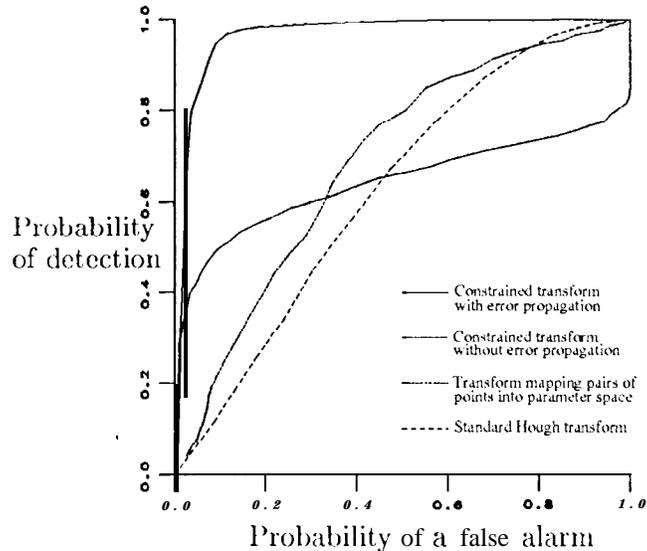


Figure 3: Receiver operating characteristic (ROC) curves for line detection generated using synthetic data.

7.2 Robust regression

Here we consider the application of RUDR to the problem of finding the least-median-of-squares (LMS) regression line. In LMS regression, the fit of the model that minimizes the median residual error is sought. We thus do not use a bounded error criterion for this case. We should report only the single best fit according the median residual criterion.

The most commonly considered problem is that of fitting a line to points in the plane. We apply RUDR to this problem by considering a series of distinguished points in the data. Each trial examines a single distinguished point (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.

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 [18]. 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.

Each subproblem (corresponding to a distinguished point) can be solved using a specialized method based on parametric search techniques [18]. This allows each subproblem to be solved exactly in $O(n \log^2 n)$ time or using numerical techniques in $O(n \log n)$ time for a

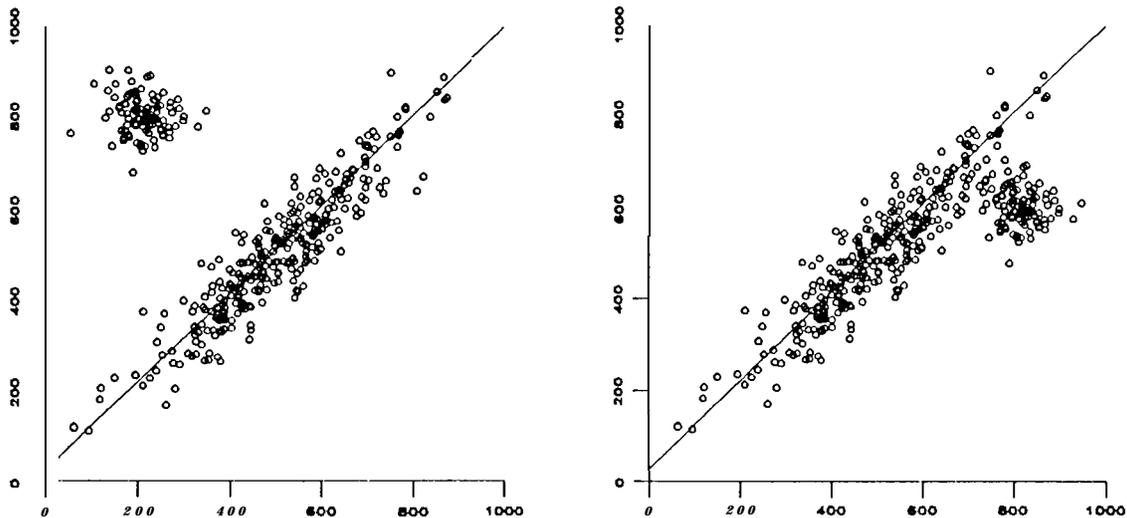


Figure 4: The RUDR paradigm can be used for robust, regression.

fixed precision solution. These complexities are lower than those for the best known exact algorithms for this problem and these algorithms yield bounds with high probability [13] the quality of the solution, unlike previous approximation algorithms. These techniques can also be extended to higher dimensions, with an increased computational complexity.

Figure 4 shows two examples where RUDR was used to perform approximate least-squares regression. In these examples, there were 300 inliers and 100 outliers, both from two-dimensional Gaussian distributions. A very good approximation to the 300 inliers is obtained in both cases. Our benchmarks show that this technique can process 100,000 data points in under a minute on a SPARCstation™20.

7.3 Object recognition

The application of the RUDR paradigm to object recognition has been explored in [20, 23]. The use of RUDR 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. This is the lowest complexity that has been achieved to perform general object recognition using the geometry of the data features, without additional information.

The method used in [20, 23] to solve each subproblem is a multi-dimensional histogramming procedure that allows clusters to be determined efficiently in the space of possible model positions using little memory. This results in an approximation algorithm, so the matchings that are found, are not necessarily those that meet the defined acceptance criterion. Breuel's

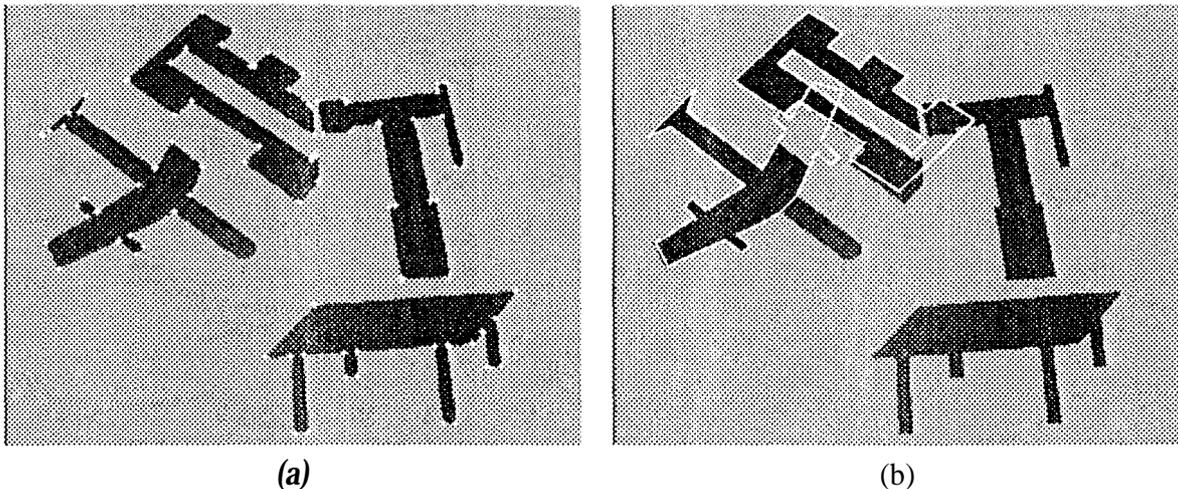


Figure 5: Recognition of occluded two-dimensional objects. (a) The corners detected in the image. (b) The best hypotheses found for the occluded objects with the edges drawn in.

adaptive subdivision of transformation space method [3] or Cass's transformation constraint analysis [6] can be used to solve the subproblems when geometrically precise results are necessary.

Figure 5 shows an example where the RUDR paradigm was used to recognize occluded two-dimensional figures by matching feature points. The partial occlusion of the objects and the distracting features from other objects did not prevent the method from recognizing the objects of interest. Figure 6 shows an example of the recognition of a three-dimensional object where self-occlusion is present.

7.4 Motion segmentation

RUDR can be used with any technique for determining structure and motion from corresponding data features in multiple images (see [10] for a review of such techniques) to perform motion segmentation. In this problem we are given sets of data features in multiple images. For now, we assume that we know the feature correspondences between images (e.g. from a tracking mechanism), but we do not know which sets of features belong to coherent objects.

Let us 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. We examine distinguished matchings of size $k - 1$ in the RUDR paradigm (i.e. $k - 1$ 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 set, and motions that are consistent with them belonging to a single object. This is repeated

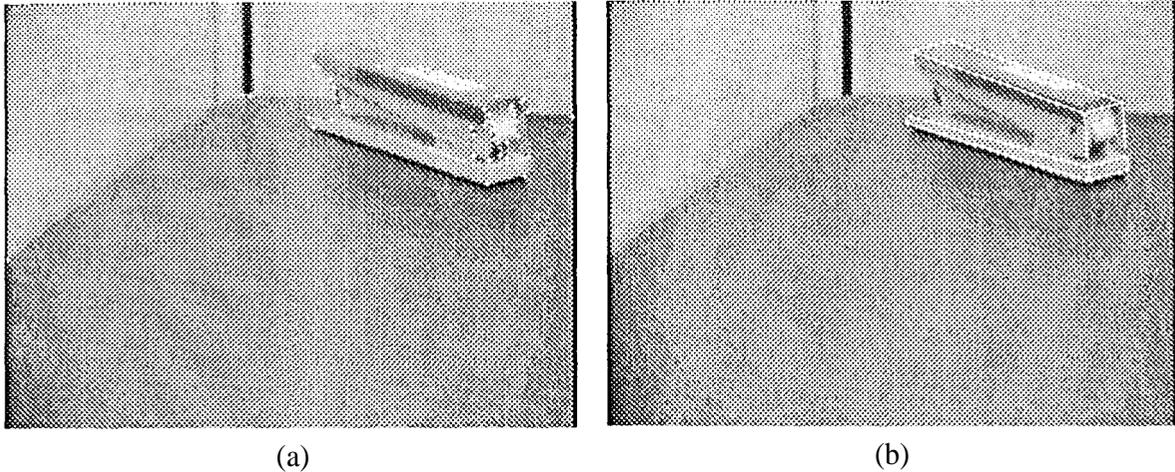


Figure 6: Recognition of a three-dimensional object. (a) Corners detected in the image. (b) Best hypothesis found (with edges drawn in).

for enough distinguished matchings to find all 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 feature is at least ϵ with a probability of failure for a particular object, no larger than γ . For fixed γ, ϵ , and k this is a constant number of trials and each trial can be performed in $O(d)$ time using histogramming.

This problem is much more difficult if we do not know the correspondences between images. In this case, we could select a distinguished set of points from one of the images and consider every possible set of matches in the other images. $O(d^{(k-1)(i-1)})$ distinguished matchings would be examined. For each of distinguished matching, we examine $O(d^i)$ minimal matchings. The total running time would thus be $O(d^{i(k-1)})$. In practice, some additional information or constraint should be used to reduce the number of matchings that must be examined when the matches between images are not known.

8 Summary

This paper has described a new algorithmic paradigm called RUDR for solving model extraction and fitting problems such as recognition and regression. This paradigm 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 paradigm 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 that examine only the model

positions consistent with some distinguished matching of features.

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 large sets of the minimal matchings that include the distinguished matching that can be brought into alignment up to some error criterion by a single model position.

This decomposition of the problem yields an equivalent formulation of the recognition problem when perfectly accurate techniques are used to solve the problems and it allows the subproblems to be solved efficiently with accurate error propagation. The additional use of randomization yields substantial gains in efficiency, offset, by a small probability that a matching that meets the acceptance criterion could be missed.

The use of this paradigm yields two primary advantages over previous generate-and-test or Hough-based methods. First, the efficiency of these techniques is superior to previous methods to solve this class of recognition and matching problems and the memory required by these techniques is low. Second, methods by which the localization error of data features can be propagated accurately without reporting matches that do not meet the acceptance criterion are possible through the use of these techniques. In addition, these techniques can be easily parallelized by mapping the subproblems that are considered onto the set of processors that are available.

References

- [1] P. D. Alter and D. W. Jacobs. Error propagation in full 3d-from-2d object recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 892-898, 1994.
- [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] A. Califano and R. M. Belle. The multiple window parameter transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(12):1157-1170, December 1992.
- [5] J. Canny. A computational approach to edge detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 8(6):679-697, November 1986.
- [6] T. A. Cass. Polynomial-time geometric matching for object recognition. *International Journal of Computer Vision*, 21(1/2):37-62, January 1997.

- [7] 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.
- [8] 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.
- [9] 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.
- [10] 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.
- [11] D. P. Huttenlocher and W. J. Rucklidge. A multi-resolution technique for comparing images using the Hausdorff distance. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 705-706, 1993.
- [12] D. P. Huttenlocher and S. Ullman. Recognizing solid objects by alignment with an image. *International Journal of Computer Vision*, 5(2):105- 212, 1990.
- [13] J. Illingworth and J. Kittler. The adaptive Hough transform. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 9(5):690-698, September 1987.
- [14] N. Kiryati, Y. Eldar, and A. M. Bruckstein. A probabilistic Hough transform. *Pattern Recognition*, 24(4):303- 316, 1991.
- [15] 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.
- [16] V. F. Leavers. Which Hough transform? *CVGIP: Image Understanding*, 58(2):250- 264, September 1993.
- [17] H. Li, M. A. Lavin, and R. J. LeMaster. Fast Hough transform: A hierarchical approach. *Computer Vision, Graphics, and Image Processing*, 36:139- 161, 1986.
- [18] C. F. Olson. An approximate algorithm for least median of squares regression. Submitted to *Information Processing Letters*, details available from the author.
- [19] C. F. Olson. Constrained Hough transforms for curve detection. Submitted to *Computer Vision and Image Understanding*, details available from the author.
- [20] C. F. Olson. Time and space efficient pose clustering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 251-258, 1994.
- [21] C. F. Olson. On the speed and accuracy of object recognition when using imperfect grouping. In *Proceedings of the International Symposium on Computer Vision*, pages 449-454, 1995.

- [22] C.F. 01s011. Decomposition of the Hough transform: Curve detection with efficient error propagation. In *Proceedings of the European Conference on Computer Vision*, volume 1, pages 263-272, 1996.
- [23] C. F. Olson. Efficient pose clustering using a randomized algorithm. *International Journal of Computer Vision*, to appear 1997.
- [24] C. F. Olson. RUDR: A new paradigm for model extraction and fitting. Technical report, Jet Propulsion Laboratory, California Institute of Technology, 1997.
- [25] G. Stock man. Object recognition and localization via pose clustering. *Computer Vision, Graphics, and Image Processing*, 40:361-387, 1987.
- [26] L. Xu, E. Oja, and P. Kultanen. A new curve detection method: Randomized Hough transform (RHIT). *Pattern Recognition Letters*, 11:331-338, May 1990.