Statistical Object Recognition

by

William Mercer Wells III

B.S., University of California, Santa Cruz (1976)
M.S., Stanford University (1984)

Submitted to the Department of Electrical Engineering and Computer Science
in partial fulfillment of the requirements for the degree of

Doctor of Philosophy

at the

MASSACHUSETTS INSTITUTE OF TECHNOLOGY

February 1993

Signature of Author ..
Department of Electrical Engineering ~~and Computer~~ Science
November 24, 1992

Certified by..... ....
W. Eric L. Grimson
Associate Professor of Electrical Engineering and Computer Science
Thesis Supervisor

Accepted by ....
Campbell L. Searle
Chairman, Departmental Committee on Graduate Students

Statistical Object Recognition

by

William Mercer Wells III

Submitted to the Department of Electrical Engineering and Computer Science
on November 24, 1992, in partial fulfillment of the
requirements for the degree of
Doctor of Philosophy

Abstract

To be practical, recognition systems must deal with uncertainty. Positions of image features in scenes vary. Features sometimes fail to appear because of unfavorable illumination. In this work, methods of statistical inference are combined with empirical models of uncertainty in order to evaluate and refine hypotheses about the occurrence of a known object in a scene.

Probabilistic models are used to characterize image features and their correspondences. A statistical approach is taken for the acquisition of object models from observations in images: *Mean Edge Images* are used to capture object features that are reasonably stable with respect to variations in illumination.

The Alignment approach to recognition, that has been described by Huttenlocher and Ullman, is used. The mechanisms that are employed to generate initial hypotheses are distinct from those that are used to verify (and refine) them. In this work, posterior probability and Maximum Likelihood are the criteria for evaluating and refining hypotheses. The recognition strategy advocated in this work may be summarized as *Align Refine Verify*, whereby local search in pose space is utilized to refine hypotheses from the alignment stage before verification is carried out.

Two formulations of model-based object recognition are described. MAP Model Matching evaluates joint hypotheses of match and pose, while Posterior Marginal Pose Estimation evaluates the pose only. Local search in pose space is carried out with the Expectation–Maximization (EM) algorithm.

Recognition experiments are described where the EM algorithm is used to refine and evaluate pose hypotheses in 2D and 3D. Initial hypotheses for the 2D experiments were generated by a simple indexing method: Angle Pair Indexing. The Linear Combination of Views method of Ullman and Basri is employed as the projection model in the 3D experiments.

Thesis Supervisor: W. Eric L. Grimson
Title: Associate Professor of Electrical Engineering and Computer Science

Acknowledgments

I feel fortunate to have had Professor W. Eric L. Grimson as my thesis advisor, mentor and friend. His deep knowledge was a tremendous asset. His sharp instincts and thoughtful guidance helped me to stay focussed on the problem of object recognition, while his support and generous advising style provided me the freedom to find and solve my own problems.

I appreciate the help offered by my reading committee, Professors Tomas Lozano-Pérez, Jeffrey Shapiro and Shimon Ullman. Their insights and criticism improved my work. In particular, I thank Professor Shapiro for offering his time and statistical expertise. His contribution to my research went well beyond his close reading of my thesis. Among his valuable suggestions was use of the EM algorithm. In thanking the above committee, however, I claim any errors or inconsistencies as my own.

I feel lucky for my years at MIT. I have enjoyed Professor Grimson's research group. I learned much about recognition from: Tao Alter, Todd Cass, David Clemens, David Jacobs, Karen Sarachick, Tanveer Syeda and Steve White as well as Amnon Shashua. In moving on, I shall miss working with such a critical mass of talent, and beyond this, I know I shall miss them as friends.

My early days at the AI Lab were spent in Professor Rodney Brook's robotics group. There, I learned a lot working on Squirt the robot from him and Anita Flynn. I appreciate their continued friendship. I found much to admire in them as colleagues. I also thank Flynn for her assistance with some experimental work in this thesis.

In the AI Lab as a whole, I enjoyed my contacts with Paul Viola, Professors Tom Knight and Berthold Horn, and others too numerous to mention. Sundar Narasimhan, Jose Robles and Pat O'Donnell provided invaluable assistance with the Puma robot. Grimson's administrative assistant, Jeanne Speckman, was terrific. I thank Professor Patrick Winston, director of the AI Lab, for providing the unique environment that the AI Lab is. My stay has been a happy one.

I spent three summers as a student at MIT Lincoln Laboratory in Group 53.

Group leader Al Gschwendtner provided support and a good environment for pursuing some of the research found in this thesis. There, I enjoyed collaborating with Murali Menon on image restoration, and learned some things about the EM algorithm from Tom Green. Steve Rak helped prepare the range images used in Chapter 7.

Prior to MIT, I worked with Stan Rosenschein at SRI International and Teleos Research. The earliest incarnation of this research originated during those years. Rosenschein led a mobile robot group comprised of Leslie Kaelbling, Stanley Reifel, myself and more loosely of Stuart Shieber and Fernando Pereira. Working with them, I learned how enjoyable research can be.

Professor Thomas O. Binford at Stanford University introduced me to computer vision. There, I found stimulating contacts in David Lowe, David Marimont, Professor Brian Wandell and Christopher Goad. After Stanford, my first opportunity to work on computerized object recognition was with Goad at Silma Incorporated.

I owe much to my parents who have always been there to support and encourage me. My time at the AI Lab would not have been possible without them.

And finally, this work depended daily upon the love and support of my wife, Colleen Gillard, and daughters, Georgia and Whitney, who will soon be seeing more of me.

This research was supported in part by the Advanced Research Projects Agency of the Department of Defense under Army contract number DACA76-85-C-0010 and under Office of Naval Research contracts N00014-85-K-0124 and N00014-91-0J-4038.

To Colleen, Georgia and Whitney

Contents

Chapter 1

Introduction

Visual object recognition is the focus of the research reported in this thesis. Recognition must deal with uncertainty to be practical. Positions of image features belonging to objects in scenes vary. Features sometimes fail to appear because of unfavorable illumination. In this work, methods of statistical inference are combined with empirical models of uncertainty in order to evaluate hypotheses about the occurrence of a known object in a scene. Other problems, such as the generation of initial hypotheses and the acquisition of object model features are also addressed.

1.1 The Problem

Representative recognition problems and their solutions are illustrated in Figures 1-1 and 1-2. The problem is to detect and locate the car in digitized video images, using previously available detailed information about the car. In these figures, object model features are superimposed over the video images at the position and orientation where the car was found. Figure 1-1 shows the results of 2D recognition, while Figure 1-2 illustrates the results of 3D recognition. These images are from experiments that are described in Chapter 10. Practical solutions to problems like these will improve the flexibility of robotic systems.

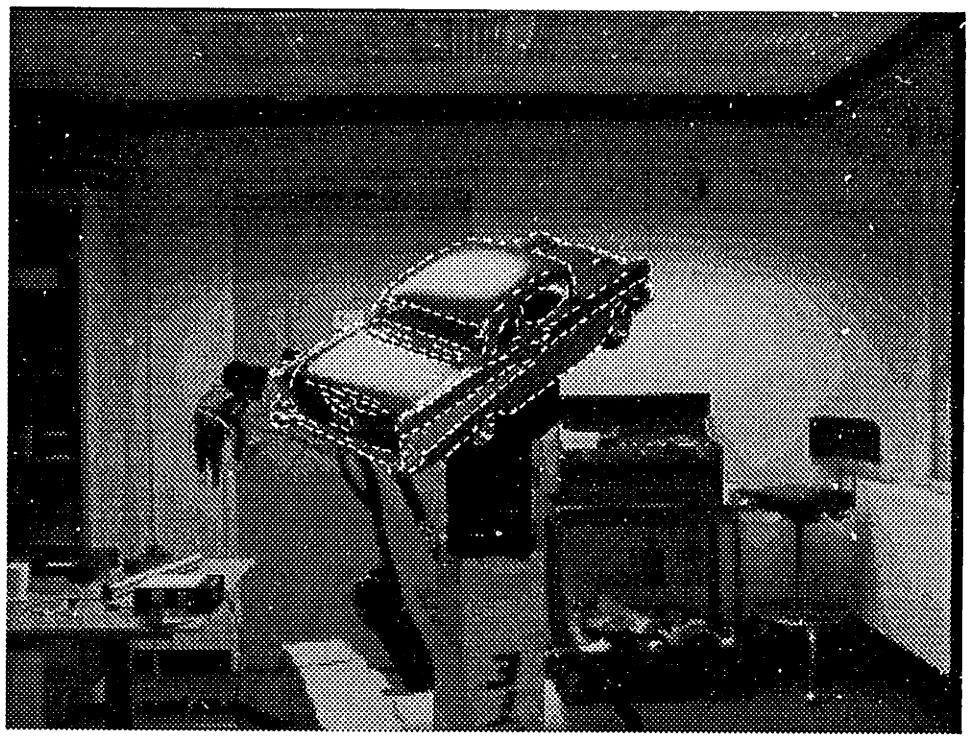


Figure 1-1: Representative Recognition Problem and Solution (2D)

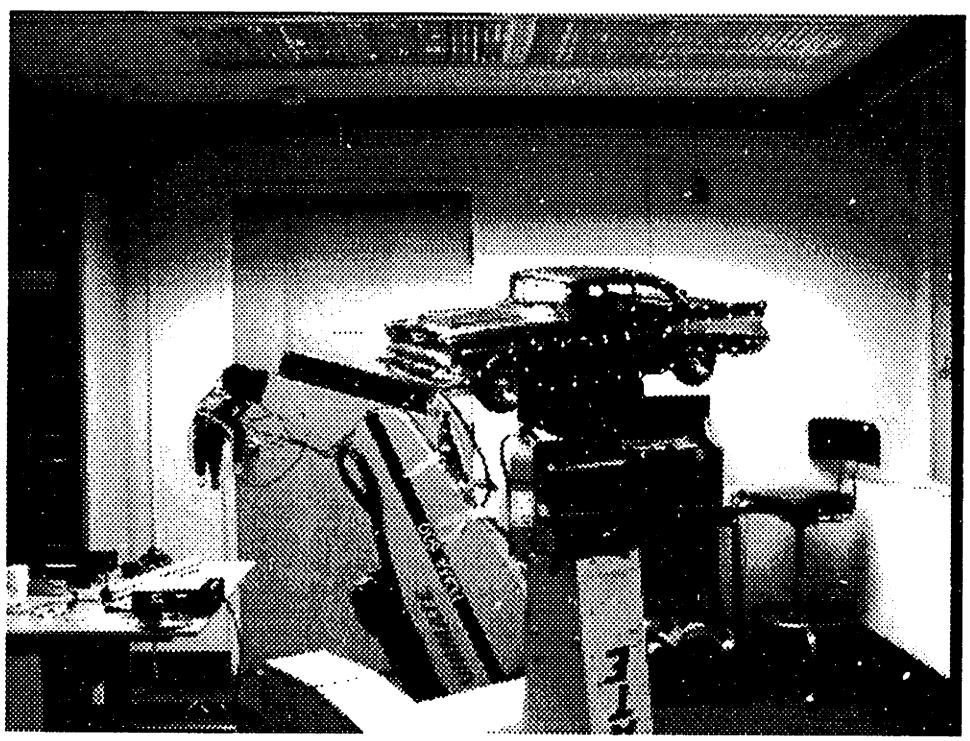


Figure 1-2: Representative Recognition Problem and Solution (3D)

In this work, the recognition problem is restricted to finding occurrences of a single object in scenes that may contain other unknown objects. Despite the simplification and years of research, the problem remains largely unsolved. Robust systems that can recognize smooth objects having six degrees of freedom of position, under varying conditions of illumination, occlusion, and background, are not commercially available. Much effort has been expended on this problem as is evident in the comprehensive reviews of research in computer-based object recognition by Besl and Jain [5], who cited 203 references, and Chin and Dyer [18], who cited 155 references. The goal of this thesis is to characterize, as well as to describe how to find, robust solutions to visual object recognition problems.

1.2 The Approach

In this work, statistical methods are used to evaluate and refine hypotheses in object recognition. Angle Pair Indexing, a means of generating hypotheses, is introduced. These mechanisms are used in an extension of the Alignment method that includes a pose refinement step. Each of these components are amplified below.

1.2.1 Statistical Approach

In this research, visual object recognition is approached via the principles of Maximum Likelihood (ML) and Maximum A-Posteriori probability (MAP). These principles, along with specific probabilistic models of aspects of object recognition, are used to derive objective functions for evaluating and refining recognition hypotheses. The ML and MAP criteria have a long history of successful application in formulating decisions and in making estimates from observed data. They have attractive properties of optimality and are often useful when measurement errors are significant.

In other areas of computer vision, statistics has proven useful as a theoretical framework. The work of Yuille, Geiger and Bülthoff on stereo [78] is one example,

while in image restoration the work of Geman and Geman [28], Marroquin [54], and Marroquin, Mitter and Poggio [55] are others. The statistical approach that is used in this thesis converts the recognition problem into a well defined (although not necessarily easy) optimization problem. This has the advantage of providing an explicit characterization of the problem, while separating it from the description of the algorithms used to solve it. Ad hoc objective functions have been profitably used in some areas of computer vision. Such an approach is used by Barnard in stereo matching [2], Blake and Zisserman [7] in image restoration and Beveridge, Weiss and Riseman [6] in line segment based model matching. With this approach, plausible forms for components of the objective function are often combined using trade-off parameters. Such trade-off parameters are determined empirically. An advantage of deriving objective functions from statistical theories is that assumptions become explicit – the forms of the objective function components are clearly related to specific probabilistic models. If these models fit the domain then there is some assurance that the resulting criteria will perform well. A second advantage is that the trade-off parameters in the objective function can be derived from measurable statistics of the domain.

1.2.2 Feature-Based Recognition

This work uses a feature-based approach to object recognition. Features are abstractions like points or curves that summarize some structure of the patterns in an image. There are several reasons for using feature based approaches to object recognition.

- Features can concisely represent objects and images. Features derived from brightness edges can summarize the important events of an image in a way that is reasonably stable with respect to scene illumination.

- In the alignment approach to recognition (to be described shortly), hypotheses are verified by projecting the object model into the image, then comparing the prediction against the image. By using compact, feature-based representations

of the object, projection costs may be kept low.

- Features also facilitate hypothesis generation. Indexing methods are attractive mechanisms for hypothesis generation. Such methods use tables indexed by properties of small groups of image features to quickly locate corresponding model features.

Object Features from Observation

A major issue that must be faced in model-based object recognition concerns the origin of the object model itself. The object features that are used in this work are derived from actual image observations. This method of feature acquisition automatically favors those features that are likely to be detected in images. The potentially difficult problem of predicting image features from abstract geometric models is bypassed. This prediction problem is manageable in some constrained domains (with polyhedral objects, for instance) but it is often difficult, especially with smooth objects, low resolution images and lighting variations.

For robustness, simple local image features are used in this work. Features of this sort are easily detected in contrast to extended features like line segments. Extended features have been used in some systems for hypothesis generation because their additional structure provides more constraint than that offered by simple local features. Extended features, nonetheless, have drawbacks in being difficult to detect due to occlusions and localized failures of image contrast. Because of this, systems that rely on distinguished features can lose robustness.

1.2.3 Alignment

Hypothesize-and-test, or *alignment* methods have proven effective in visual object recognition. Huttenlocher and Ullman [43] used search over minimal sets of corresponding features to establish candidate hypotheses. In their work these hypotheses,

or *alignments*, are tested by projecting the object model into the image using the pose (position and orientation) implied by the hypothesis, and then by performing a detailed comparison with the image. The basic strategy of the alignment method is to use separate mechanisms for generating and testing hypotheses.

Recently, indexing methods have become available for efficiently generating hypotheses in recognition. These methods avoid a significant amount of search by using pre-computed tables for looking up the object features that might correspond to a group of image features. The geometric hashing method of Lamdan and Wolfson [49] uses invariant properties of small groups of features under affine transformations as the look-up key. Clemens and Jacobs [19] [20], and Jacobs [45] described indexing methods that gain efficiency by using a feature grouping process to select small sets of image features that are likely to belong to one object in the scene.

In this work, a simple form of 2D indexing, *Angle Pair Indexing*, is used to generate initial hypotheses. It uses an invariant property of pairs of image features under translation, rotation and scale. This is described in Chapter 9.

The Hough transform [40] [44] is another commonly used method for generating hypotheses in object recognition. In the Hough method, feature-based clustering is performed in *pose space*, the space of the transformations describing the possible motion of the object. This method was used by Grimson and Lozano-Pérez [36] to localize the search in recognition.

These fast methods of hypothesis generation provide ongoing reasons for using the alignment approach. They are often most effective when used in conjunction with verification. Verification is important because indexing methods can be susceptible to table collisions, while Hough methods sometimes generate false positives due to their aggregation of inconsistent evidence in pose space bins. This last point has been argued by Grimson and Huttenlocher [35].

The usual alignment strategy may be summarized as *align verify*. Alignment and verification place differing pressures on the choice of features for recognition. Mech-

anisms used for generating hypotheses typically have computational complexity that is polynomial in the number of features involved. Because of this, there is significant advantage to using low resolution features – there are fewer of them. Unfortunately, pose estimates based on coarse features tend to be less accurate than those based on high resolution features.

Likewise, verification is usually more reliable with high resolution features. This approach yields more detailed comparisons. These differing pressures may be accommodated by employing *coarse-fine* approaches. The coarse-fine strategy was utilized successfully in stereo by Grimson [33]. In the coarse-fine strategy, hypotheses derived from low-resolution features limit the search for hypotheses derived from high-resolution features. There are some potential difficulties that arise when applying coarse-fine methods in conjunction with 3D object models. These may be avoided by using view-based alternatives to 3D object modeling. These issues are discussed more fully in Chapter 4.

Align Refine Verify

The recognition strategy advocated in this work may be summarized as *align refine verify*. This approach has been used by Lipson [50] in refining alignments. The key observation is that local search in pose space may be used to refine the hypothesis from the alignment stage before verification is carried out. In hypothesize and test methods, the pose estimates of the initial hypotheses tend to be somewhat inaccurate, since they are based on minimal sets of corresponding features. Better pose estimates (hence, better verifications) are likely to result from using all supporting image feature data, rather than a small subset. Chapter 8 describes a method that refines the pose estimate while simultaneously identifying and incorporating the constraints of all supporting image features.

1.3 Guide to Thesis

Briefly, the presentation of the material in this thesis is essentially bottom-up. The early chapters are concerned with building the components of the formulation, while the main contributions, the statistical formulations of object recognition, are described in Chapters 6 and 7. After that, related algorithms are described, followed by experiments and conclusions.

In more detail, Chapter 2 describes the probabilistic models of the correspondences, or mapping between image features and features belonging to either the object or to the background. These models use the principle of maximum-entropy where little information is available before the image is observed. In Chapter 3, probabilistic models are developed that characterize the feature detection process. Empirical evidence is described to support the choice of model.

Chapter 4 discusses a way of obtaining average object edge features from a sequence of observations of the object in images. Deterministic models of the projection of features into the image are discussed in Chapter 5. The projection methods used in this work are linear in the parameters of the transformations. Methods for 2D and 3D are discussed, including the Linear Combination of Views method of Ullman and Basri [71].

In Chapter 6 the above models are combined in a Bayesian framework to construct a criterion, *MAP Model Matching*, for evaluating hypotheses in object recognition. In this formulation, complete hypotheses consist of a description of the correspondences between image and object features, as well as the pose of the object. These hypotheses are evaluated by their posterior (after the image is observed) probability. A recognition experiment is described that uses the criteria to guide a heuristic search over correspondences. A connection between MAP Model Matching and a method of robust chamfer matching [47] is described.

Building on the above, a second criterion is described in Chapter 7: *Posterior Marginal Pose Estimation* (PMPE). Here, the solution being sought is simply the

pose of the object. The posterior probability of poses is obtained by taking the formal marginal, over all possible matches, of the posterior probability of the joint hypotheses of MAP Model Matching. This results in a smooth, non-linear objective function for evaluating poses. The smoothness of the objective function facilitates local search in pose space as a mechanism for refining hypotheses in recognition. Some experimental explorations of the objective function in pose space are described. These characterizations are carried out in two domains: video imagery and synthetic radar range imagery.

Chapter 8 describes use of the the *Expectation-Maximization* (EM) algorithm [21] for finding local maxima of the PMPE objective function. This algorithm alternates between the M step – a weighted least squares pose estimate, and the E step – re-calculation of the weights based on a saturating non-linear function of the residuals.

This algorithm is used to refine and evaluate poses in 2D and 3D recognition experiments that are described in Chapter 10. Initial hypotheses for the 2D experiments were generated by a simple indexing method, *Angle Pair Indexing*, that is described in Chapter 9 . The Linear Combination of Views method of Ullman and Basri [71] is employed as the projection model in the 3D experiments reported there.

Finally, some conclusions are drawn in Chapter 11. The notation used throughout is summarized in Appendix A.

Chapter 2

Modeling Feature Correspondence

This chapter is concerned with probabilistic models of feature correspondences. These models will serve as priors in the statistical theories of object recognition that are described in Chapters 6 and 7, and are important components of those formulations. They are used to assess the probability that features correspond before the image data is compared to the object model. They capture the expectation that some features in an image are anticipated to be due to the object

Three different models of feature correspondence are described, one of which is used in the recognition experiments described in Chapters 6, 7, and 10.

2.1 Features and Correspondences

This research focuses on feature-based object recognition. The object being sought and the image being analyzed consist of discrete features.

Let the image that is to be analyzed be represented by a set of v-dimensional point features

$$Y = \{Y_1, Y_2, \ldots, Y_n\} \ , \ Y_i \in R^v \ .$$

Image features are discussed in more detail in Chapters 3 and 5.

The object to be recognized is also described by a set of features,

$$M = \{M_1, M_2, \ldots, M_m\} \ .$$

The features will usually be represented by real matrices. Additional details on object features appears in Chapters 4 and 5.

In this work, the interpretation of the features in an image is represented by the variable Γ, which describes the mapping from image features to object features or the scene background. This is also referred to as the *correspondences.*

$$\Gamma = \{\Gamma_1, \Gamma_2, \ldots, \Gamma_n\} \ , \ \Gamma_i \in M \cup \{\perp\} \ .$$

In an interpretation, each image feature, Y_i, will be assigned either to some object feature M_j, or to the background, which is denoted by the symbol $\perp$. This symbol plays a role similar to that of the null character in the interpretation trees of Grimson and Lozano-Pérez [36]. An interpretation is illustrated in Figure 2-1. Γ is a collection of variables that is indexed in parallel with the image features. Each variable Γ_i represents the assignment of the corresponding image feature Y_i. It may take on as value any of the object features M_j, or the background, $\perp$. Thus, the meaning of the expression $\Gamma_5 = M_6$ is that image feature five is assigned to object feature six, likewise $\Gamma_7 = \perp$ means that image feature seven has been assigned to the background. In an interpretation each image feature is assigned, while some object features may not be. Additionally, several image features may be assigned to the same object feature. This representation allows image interpretations that are implausible – other mechanisms are used to encourage metrical consistency.

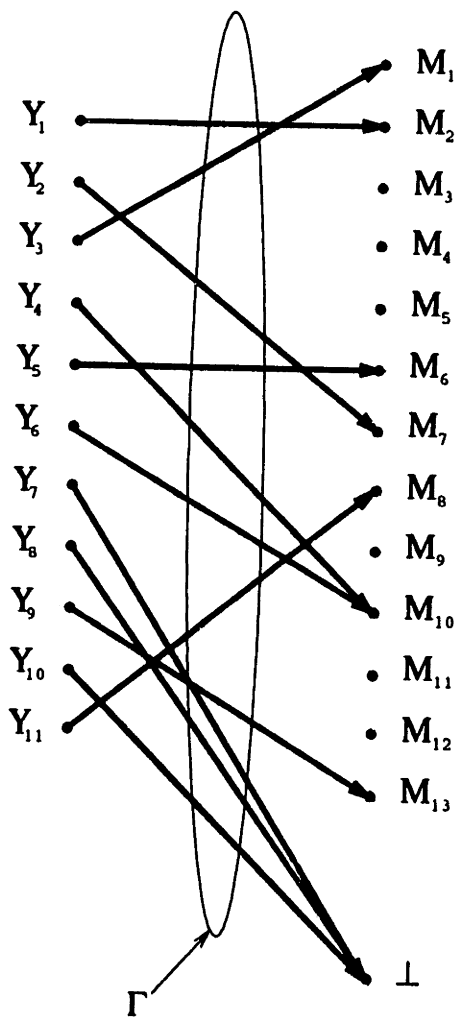


Figure 2-1: Image Features, Object Features, and Correspondences

2.2 An Independent Correspondence Model

In this section a simple probabilistic model of correspondences is described. The intent is to capture some information bearing on correspondences before the image is compared to the object. This model has been designed to be a reasonable compromise between simplicity and accuracy.

In this model, the correspondence status of differing image features are assumed to be independent, so that

$$p(\Gamma) = \prod_i p(\Gamma_i) \ . \tag{2.1}$$

Here, $p(\Gamma)$ is a probability mass function on the discrete variable Γ. There is evidence against using statistical independence here, for example, occlusion is locally correlated. Independence is used as an engineering approximation that simplifies the resulting formulations of recognition. It may be justified by the good performance of the recognition experiments described in Chapters 6, 7, and 10. Few recognition systems have used non-independent models of correspondence. Breuel outlined one approach in his thesis [9]. A relaxation of this assumption is discussed in the following section.

The component probability function is designed to characterize the amount of clutter in the image, but to be otherwise as non-committal as possible:

$$p(\Gamma_i) = \begin{cases} B & \text{if } \Gamma_i = \perp \\ \frac{1-B}{m} & \text{otherwise} \end{cases} \ . \tag{2.2}$$

The joint model $p(\Gamma)$ is the maximum entropy probability function that is consistent with the constraint that the probability of an image feature belonging to the background is B. B may be estimated by taking simple statistics on images from the domain. $B = .9$ would mean that 90 % of image features are expected to be due to the background.

Having B constant during recognition is an approximation. The number of fea-

tures due to the object will likely vary according to the size of the object in the scene. B could be estimated at recognition time by pre-processing mechanisms that evaluate image clutter, and factor in expectations about the size of the object. In practice, the approximation works well in controlled situations.

The independent correspondence model is used in the experiments reported in this research.

2.3 A Markov Correspondence Model

As indicated above, one inaccuracy of the independent correspondence model is that sample realizations of Γ drawn from the probability function of Equations 2.1 and 2.2 will tend to be overly fragmented in their modeling of occlusion. This section describes a compromise model that relaxes the independence assumption somewhat by allowing the correspondence status of an image feature (Γ_i) to depend on that of its neighbors. In the domain of this research, image features are fragments of image edge curves. These features have a natural neighbor relation, adjacency along the image edge curve, that may be used for constructing a 1D Markov Random Field (MRF) model of correspondences. MRF's are collections of random variables whose conditional dependence is restricted to limited size neighborhoods. MRF models are discussed by Geman and Geman [28]. The following describes an MRF model of correspondences intended to provide a more accurate model of occlusion.

$$p(\Gamma) = q(\Gamma_1)q(\Gamma_2)\ldots q(\Gamma_n)\, r_1(\Gamma_1,\Gamma_2)r_2(\Gamma_2,\Gamma_3)\ldots r_{n-1}(\Gamma_{n-1},\Gamma_n) \ , \tag{2.3}$$

where

$$q(\Gamma_i) = \begin{cases} e_1 & \text{if } \Gamma_i = \perp \\ e_2 & \text{otherwise} \end{cases} \tag{2.4}$$

and

$$r_i(a,b) = \begin{cases} \left\{ \begin{array}{ll} e_3 & \text{if } a = \perp \text{ and } b = \perp \\ e_4 & \text{if } a \neq \perp \text{ and } b \neq \perp \\ e_5 & \text{otherwise} \end{array} \right\} & \text{if features } i \text{ and } i+1 \text{ are neighbors} \\ 1 & \text{otherwise .} \end{cases} \tag{2.5}$$

The assignment of indices to image features should be done in such a way that neighboring features have adjacent indices. The functions $r_i(\cdot,\cdot)$ model the interaction of neighboring features. The parameters $e_1 \ldots e_5$ may be adjusted so that the probability function $p(\Gamma)$ is consistent with observed statistics on clutter and frequency of adjacent occlusions. Additionally, the parameters must be constrained so that Equation 2.3 actually describes a probability function. When these constraints are met, the model will be the maximum entropy probability function consistent with the constraints. Satisfying the constraints is a non-trivial selection problem that may be approached iteratively. Fortunately, this calculation doesn't need to be carried out at recognition time. Goldman [30] discusses methods of calculating these parameters.

The model outlined in Equations 2.3 – 2.5 is a generalization of the Ising spin model. Ising models are used in statistical physics to model ferromagnetism [73]. Samples drawn from Ising models exhibit spatial clumping whose scale depends on the parameters. In object recognition, this clumping behavior may provide a more accurate model of occlusion.

The standard Ising model is shown for reference in the following equations. It has been restricted to 1D, and has been adapted to the notation of this section.

$$\sigma_i \in \{-1, 1\}$$

$$p(\sigma_1\sigma_2 \ldots \sigma_n) = \frac{1}{Z} q(\sigma_1)q(\sigma_2)\cdots q(\sigma_n)\; r(\sigma_1,\sigma_2)r(\sigma_2,\sigma_3)\cdots r(\sigma_{n-1},\sigma_n)$$

$$q(a) = \begin{cases} \exp(\frac{\mu H}{kT}) & \text{if } a = 1 \\ \exp(-\frac{\mu H}{kT}) & \text{otherwise} \end{cases}$$

$$r(a,b) = \begin{cases} \exp(\frac{J}{kT}) & \text{if } a = b \\ \exp(-\frac{J}{kT}) & \text{otherwise} \end{cases} .$$

Here, Z is a normalization constant, μ is the moment of the magnetic dipoles, H is the strength of the applied magnetic field, k is Boltzmann's constant, T is temperature, and J is a neighbor interaction constant called the exchange energy.

The approach to modeling correspondences that is described in this section was outlined in Wells [74] [75]. Subsequently, Breuel [9] described a similar local interaction model of occlusion in conjunction with a simplified statistical model of recognition that used boolean features in a classification based scheme.

The Markov correspondence model is not used in the experiments reported in this research.

2.4 Incorporating Saliency

Another route to more accurate modeling of correspondences is to exploit bottom-up saliency processes to suggest which image features are most likely to correspond to the object. One such process in described by Ullman and Shashua [66].

For concreteness, assume that the saliency process provide a per-feature measure of saliency, S_i. To incorporate this information, we construct $p(\Gamma_i = \perp | S_i)$. This may be conveniently calculated via Bayes' rule as follows:

$$p(\Gamma_i = \perp | S_i) = \frac{p(S_i \mid \Gamma_i = \perp) p(\Gamma_i = \perp)}{p(S_i)} .$$

$p(S_i \mid \Gamma_i = \perp)$ and $p(S_i)$ are probability densities that may be estimated from observed frequencies in training data. As in Section 2.2, we set $p(\Gamma_i = \perp) = B$.

A feature specific background probability may then be defined as follows:

$$B_i \equiv p(\Gamma_i = \perp | S_i) = \frac{p(S_i \mid \Gamma_i = \perp)}{p(S_i)} B \ .$$

In this case the complete probability function on Γ_i will be

$$p(\Gamma_i) = \begin{cases} B_i & \text{if } \Gamma_i = \perp \\ \frac{1-B_i}{m} & \text{otherwise} \end{cases} \ . \tag{2.6}$$

This model is not used in the experiments described in this research.

2.5 Conclusions

The simplest of the three models described, the independent correspondence model, has been used to good effect in the recognition experiments described in Chapters 6, 7, and 10. In some domains additional robustness in recognition might result from using either the Markov correspondence model, or by incorporating saliency information.

Chapter 3

Modeling Image Features

Probabilistic models of image features are the topic of this chapter. These are another important component of the statistical theories of object recognition that are described in Chapters 6 and 7.

The probability density function for the coordinates of image features, conditioned on correspondences and pose, is defined. The PDF has two important cases, depending on whether the image feature is assigned to the object, or to the background. Features matched to the object are modeled with normal densities, while uniform densities are used for background features. Empirical evidence is provided to support the use of normal densities for matched features. A form of stationarity is described.

Many recognition systems implicitly use uniform densities (rather than normal densities) to model matched image features (*bounded error* models). The empirical evidence of Section 3.2.1 indicates that the normal model may sometimes be better. Because of this, use of normal models may provide better performance in recognition.

3.1 A Uniform Model for Background Features

The image features, Y_i, are v dimensional vectors. When assigned to the background, they are assumed to be uniformly distributed,

$$p(Y_i \mid \Gamma, \beta) = \frac{1}{W_1 \cdots W_v} \qquad \text{if } \Gamma_i = \perp \quad . \tag{3.1}$$

(The PDF is defined to be zero outside the coordinate space of the image features, which has extent W_i along dimension i.) Γ describes the correspondences from image features to object features, and β describes the position and orientation, or *pose* of the object. For example, if the image features are 2D points in a 640 by 480 image, then $p(Y_i \mid \perp, \beta) = \frac{1}{640*480}$, within the image. For Y_i, this probability function depends only on the i'th component of Γ.

Providing a satisfying probability density function for background features is problematical. Equation 3.1 describes the maximum entropy PDF consistent with the constraint that the coordinates of image features are always expected to lie within the coordinate space of the image features. E.T. Jaynes [46] has argued that maximum entropy distributions are the most honest representation of a state of incomplete knowledge.

3.2 A Normal Model for Matched Features

Image features that are matched to object features are assumed to be normally distributed about their predicted position in the image,

$$p(Y_i \mid \Gamma, \beta) = \mathrm{G}_{\psi_{ij}}(Y_i - \mathcal{P}(M_j, \beta)) \qquad \text{if } \Gamma_i = M_j \quad . \tag{3.2}$$

Here Y_i, Γ, and β are defined as above.

$\mathrm{G}_{\psi_{ij}}$ is the v-dimensional Gaussian probability density function with covariance

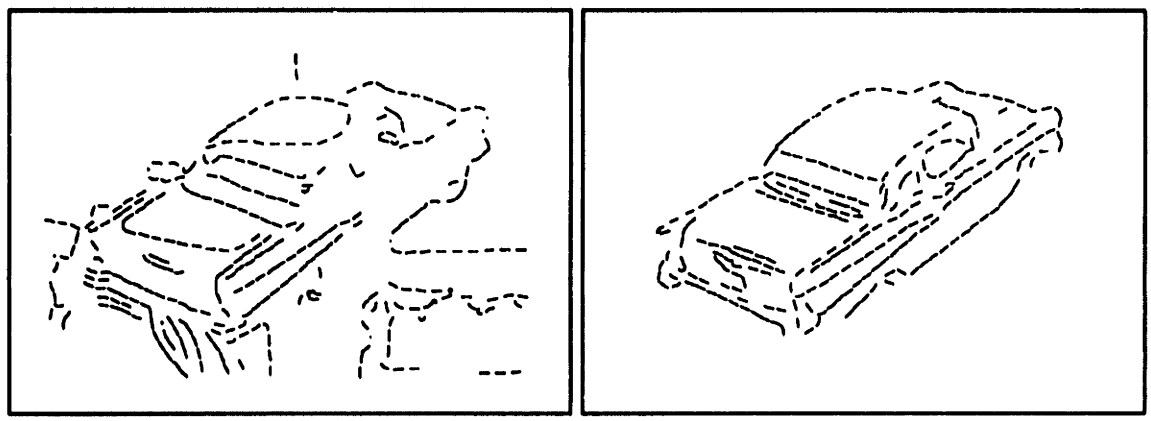


Figure 3-1: Fine Image Features and Fine Model Features

matrix ψ_{ij},

$$G_{\psi_{ij}}(x) = (2\pi)^{-\frac{k}{2}}|\psi_{ij}|^{-\frac{1}{2}} \exp\left(-\frac{1}{2}x^T\psi_{ij}^{-1}x\right) .$$

The covariance matrix ψ_{ij} is discussed more fully in Section 3.3.

When $\Gamma_i = M_j$, the predicted coordinates of image feature Y_i are given by $\mathcal{P}(M_j, \beta)$, the projection of object feature j into the image with object pose β. Projection and pose are discussed in more detail in Chapter 5.

3.2.1 Empirical Evidence for the Normal Model

This section describes some empirical evidence from the domain of video image edge features indicating that normal probability densities are good models of feature fluctuations, and that they can be better than uniform probability densities. The evidence is provided in the form of observed and fitted cumulative distributions and Kolmogorov-Smirnov tests. The model distributions were fitted to the data using the Maximum Likelihood method.

The data that is analyzed are the perpendicular and parallel deviations of fine and coarse edge features derived from video images. The fine and coarse features are shown in Figures 3-1 and 3-3 respectively.

The model features are from Mean Edge Images, these are described in Section 4.4. The edge operator used in obtaining the image features is ridges in the magnitude

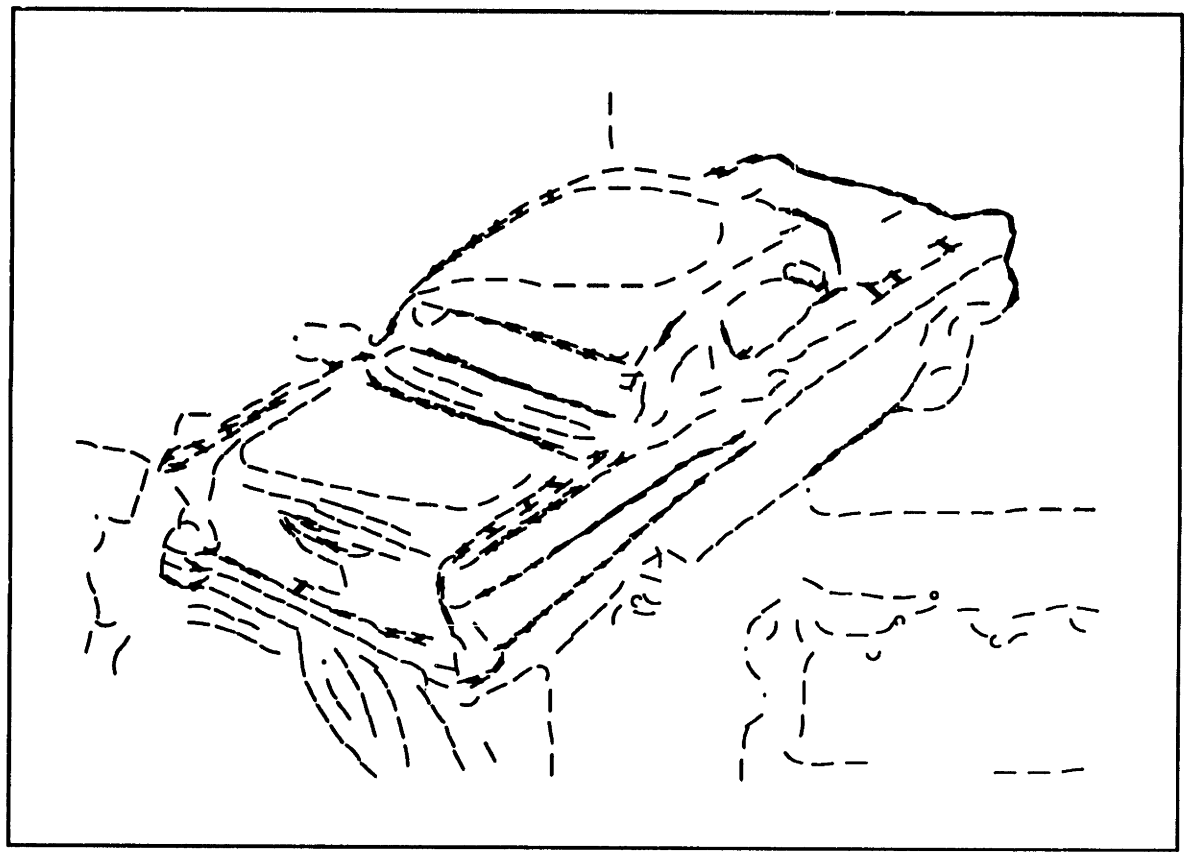


Figure 3-2: Fine Feature Correspondences

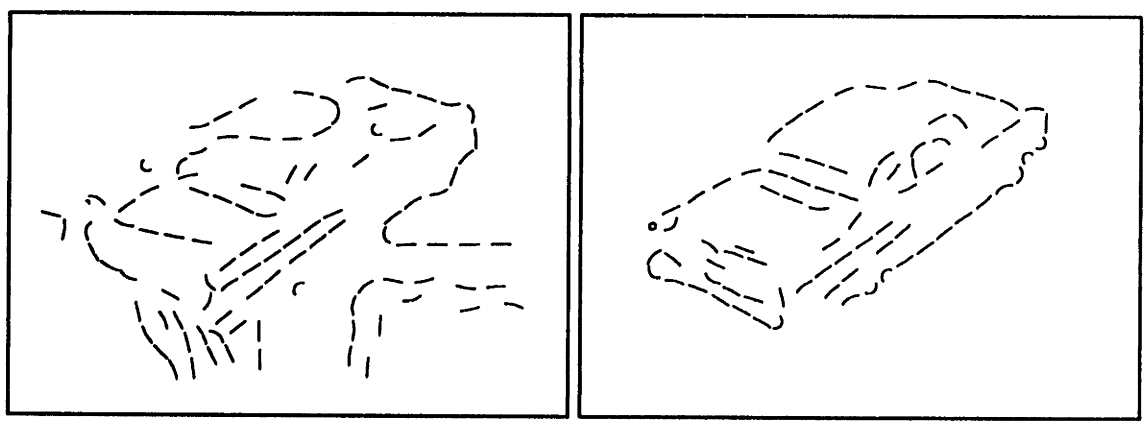


Figure 3-3: Coarse Image Features and Coarse Model Features

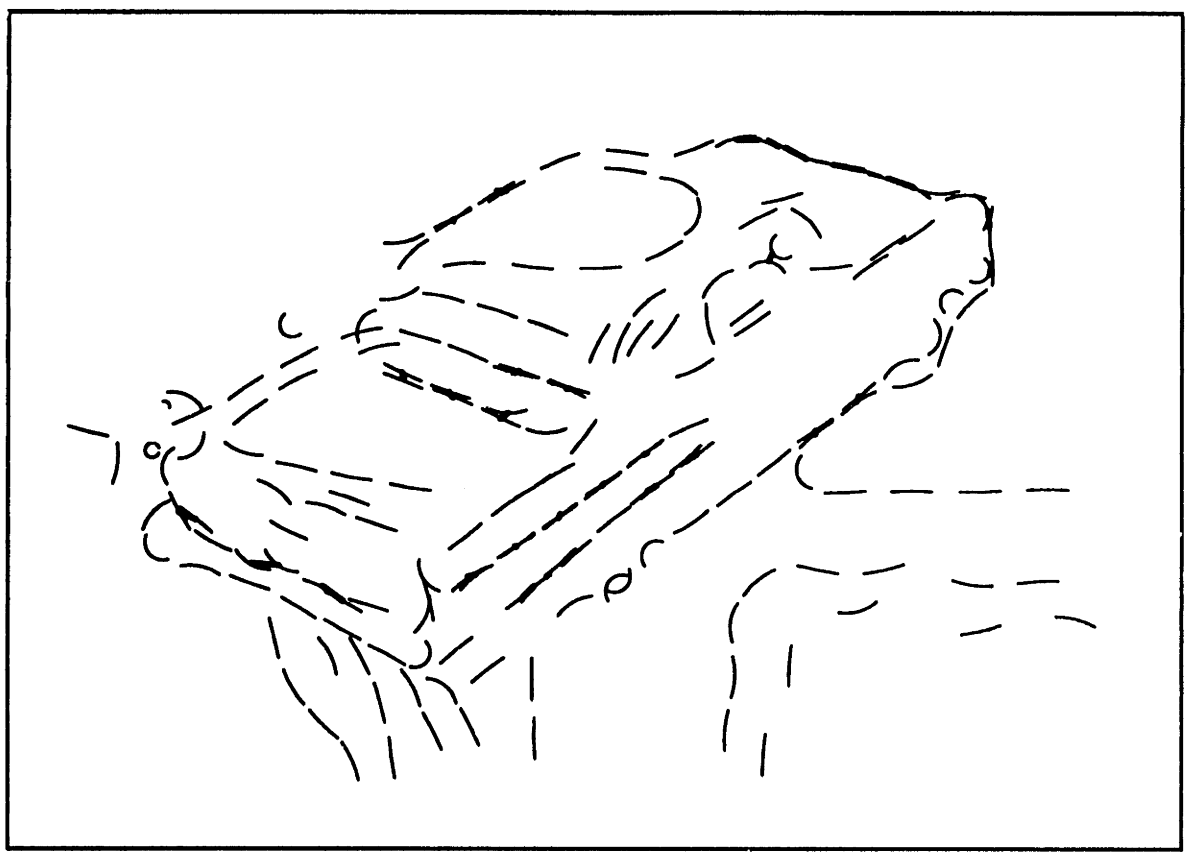


Figure 3-4: Coarse Feature Correspondences

of the image gradient, as discussed in Section 4.4. The smoothing standard deviation used in the edge detection was 2.0 and 4.0 pixels respectively, for the fine and coarse features. These features were also used in the experiments reported in Section 10.1, and the correspondences were used there as training data.

For the analysis in this section, the feature data consists of the average of the x and y coordinates of the pixels from edge curve fragments – they are 2D point features. The features are displayed as circular arc fragments for clarity. The edge curves were broken arbitrarily into 10 and 20 pixel fragments for the fine and coarse features respectively.

Correspondences from image features to model features were established by a neutral subject using a mouse. These correspondences are indicated by heavy lines in Figures 3-2 and 3-4. Perpendicular and parallel deviations of the corresponding features were calculated with respect to the normals to edge curves at the image features.

Figure 3-5 shows the cumulative distributions of the perpendicular and parallel deviations of the fine features. The cumulative distributions of fitted normal densities are plotted as heavy dots over the observed distributions. The distributions were fitted to the data using the Maximum Likelihood method – the mean and variance of the normal density are set to the mean and variance of the data. These figures show good agreement between the observed distributions, and the fitted normal distributions. Similar observed and fitted distributions for the coarse deviations are shown in Figure 3-6, again with good agreement.

The observed cumulative distributions are shown again in Figures 3-7 and 3-8, this time with the cumulative distributions of fitted uniform densities over-plotted in heavy dots. As before, the uniform densities were fitted to the data using the Maximum Likelihood method – in this case the uniform densities are adjusted to just include the extreme data. These figures show relatively poor agreement between the observed and fitted distributions, in comparison to normal densities.

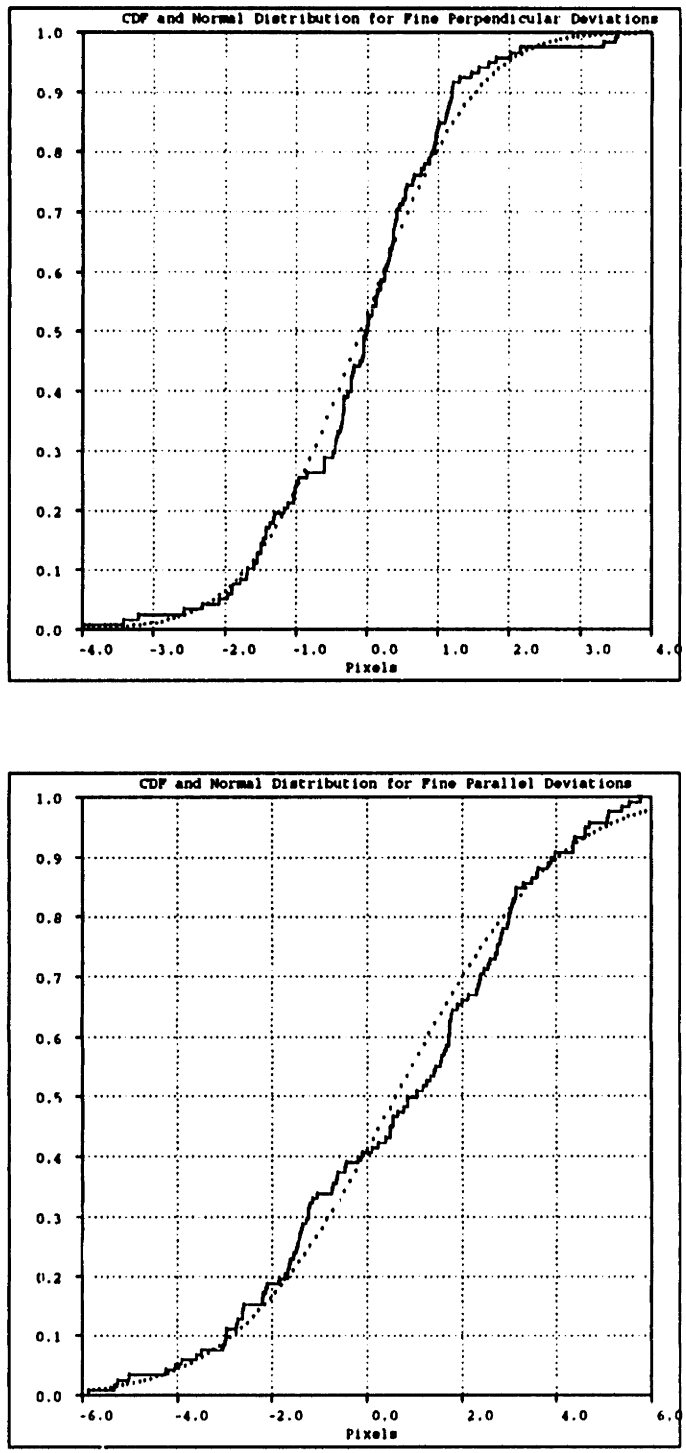
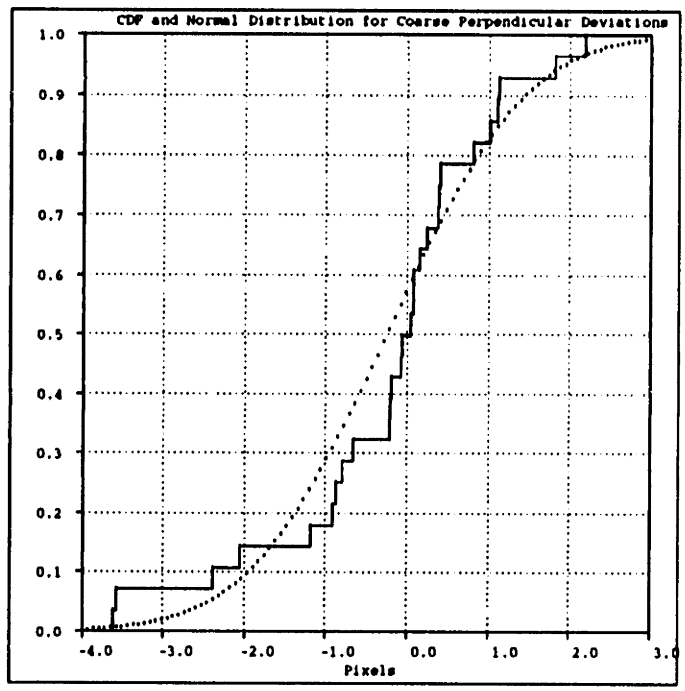


Figure 3-5: Observed Cumulative Distributions and Fitted Normal Cumulative Distributions for Fine Features



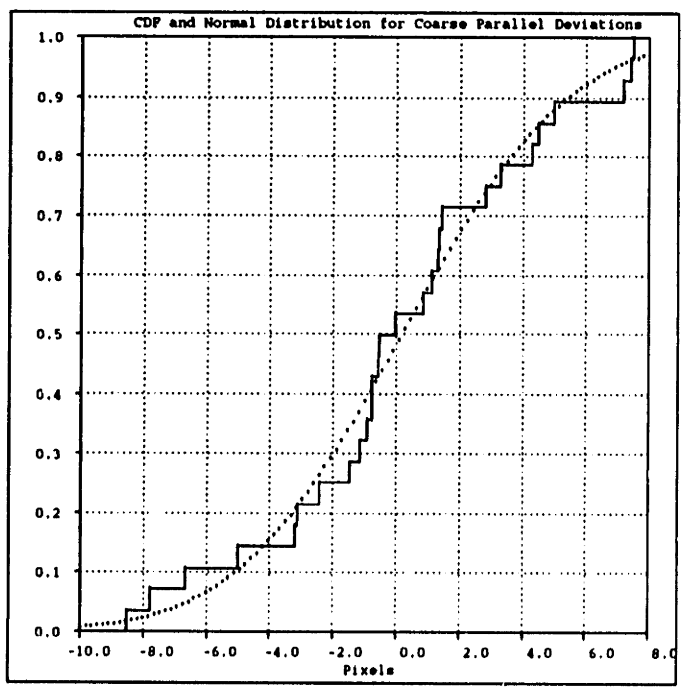


Figure 3-6: Observed Cumulative Distributions and Fitted Normal Cumulative Distributions for Coarse Features

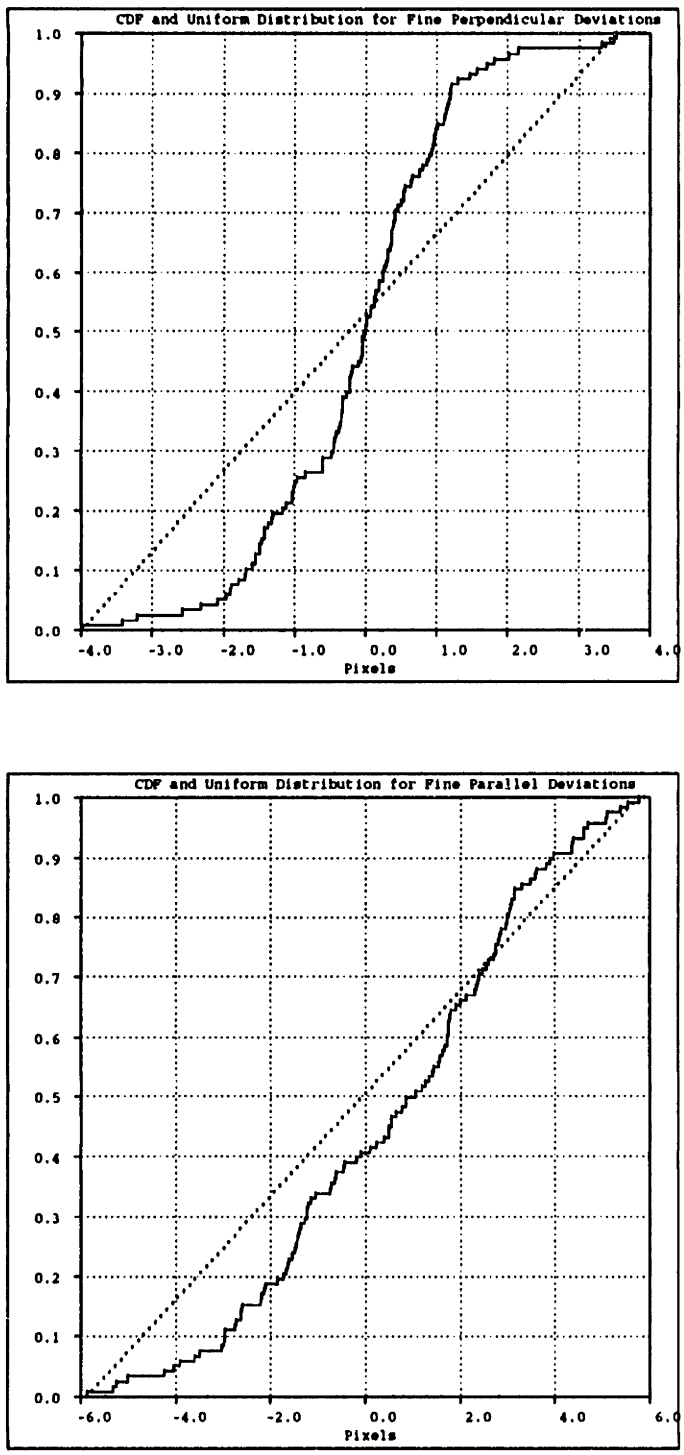
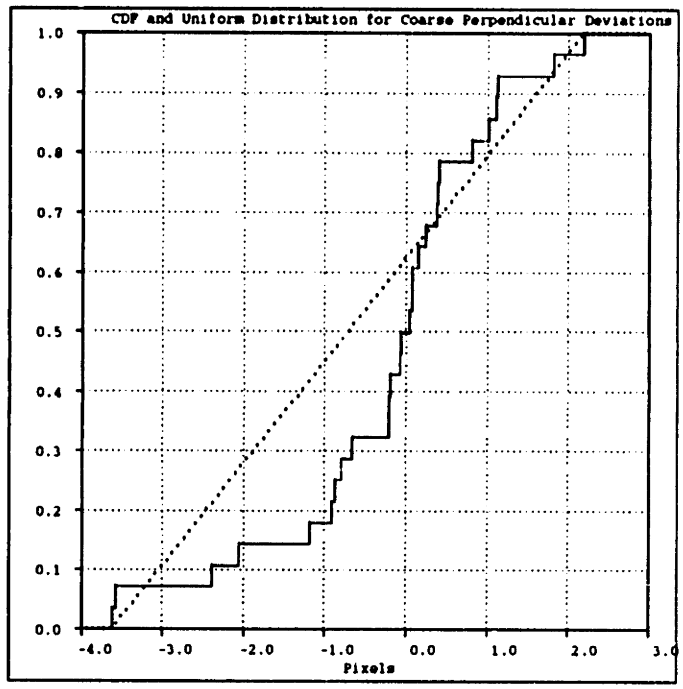


Figure 3-7: Observed Cumulative Distributions and Fitted Uniform Cumulative Distributions for Fine Features



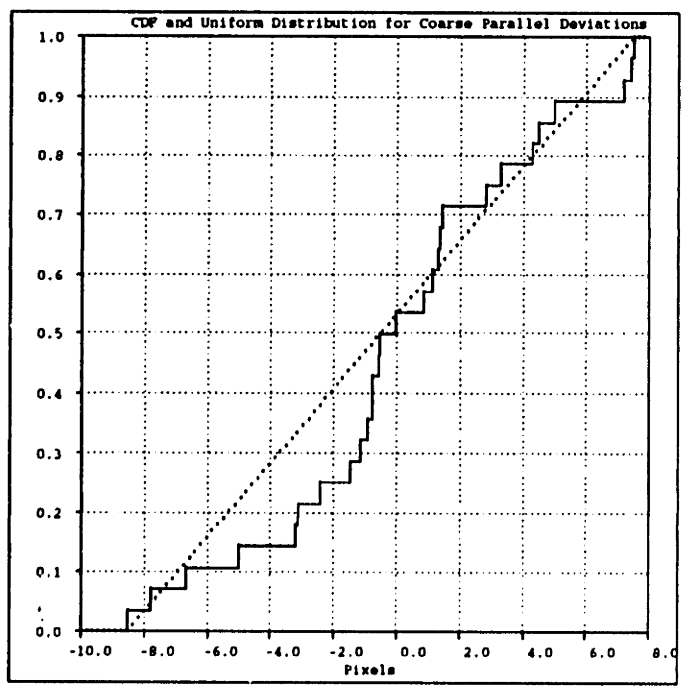


Figure 3-8: Observed Cumulative Distributions and Fitted Uniform Cumulative Distributions for Coarse Features

Deviate	N	Normal Hypothesis		Uniform Hypothesis	
		D_o	$P(D \geq D_o)$	D_o	$P(D \geq D_o)$
Fine Perpendicular	118	.0824	.3996	.2244	.000014
Fine Parallel	118	.0771	.4845	.1596	.0049
Coarse Perpendicular	28	.1526	.5317	.2518	.0574
Coarse Parallel	28	.0948	.9628	.1543	.5172

Table 3.1: Kolmogorov-Smirnov Tests

Kolmogorov-Smirnov Tests

The Kolmogorov-Smirnov (KS) test [59] is one way of analyzing the agreement between observed and fitted cumulative distributions, such as the ones in Figures 3-5 to 3-8. The KS test is computed on the magnitude of the largest difference between the observed and hypothesized (fitted) distributions. This will be referred to as D. The probability distribution on this distance, under the hypothesis that the data were drawn from the hypothesized distribution, can be calculated. An asymptotic formula is given by

$$P(D \geq D_o) = Q(\sqrt{N} D_o)$$

where

$$Q(x) = 2 \sum_{j=1}^{\infty} (-1)^{j-1} \exp\left(-2j^2 x^2\right) \ ,$$

and D_o is the observed value of D.

The results of KS tests of the consistency of the data with fitted normal and uniform distributions are shown in Table 3.1. Low values of $P(D \geq D_o)$ suggest incompatibility between the data and the hypothesized distribution. In the cases of fine perpendicular and parallel deviations, and coarse perpendicular deviations, refutation of the uniform model is strongly indicated. Strong contradictions of the fitted normal models are not indicated in any of the cases.

3.3 Oriented Stationary Statistics

The covariance matrix ψ_{ij} that appears in the model of matched image features in Equation 3.2 is allowed to depend on both the image feature and the object feature involved in the correspondence. Indexing on i allows dependence on the image feature detection process, while indexing in j allows dependence on the identity of the model feature. This is useful when some model features are know to be noisier than others. This flexibility is carried through the formalism of later chapters. Although such flexibility can be useful, substantial simplification results by assuming that the features statistics are stationary in the image, i.e. $\psi_{ij} = \psi$, for all ij. This could be reasonable if the feature fluctuations were isotropic in the image, for example. In its strict form this assumption may be too limiting, however. This section outlines a compromise approach, oriented stationary statistics, that was used in the implementations described in Chapters 6, 7, and 8.

This method involves attaching a coordinate system to each image feature. The coordinate system has its origin at the point location of the feature, and is oriented with respect to the direction of the underlying curve at the feature point. When (stationary) statistics on feature deviations are measured, they are taken relative to these coordinate systems.

3.3.1 Estimating the Parameters

The experiments reported in Sections 6.2, 7.1, and Chapter 10 use the normal model and oriented stationary statistics for matched image features. After this choice of model, it is still necessary to supply the specific parameters for the model, namely, the covariance matrices, ψ_{ij}, of the normal densities.

The parameters were estimated from observations on matches done by hand on sample images from the domain. Because of the stationarity assumption it is possible to estimate the common covariance, $\hat{\psi}$, by observing match data on one image. For

this purpose, a match was done with a mouse between features from a Mean Edge Image (these are described in Section 4.4) and a representative image from the domain. During this process, the pose of the object was the same in the two images. This produced a set of corresponding edge features. For the sake of example, the process will be described for 2D point features (described in Section 5.2). The procedure has also been used with 2D point-radius features and 2D oriented-range features, that are described in Sections 5.3 and 5.4 respectively.

Let the observed image features be described by Y_i, and the corresponding mean model features by $\hat{Y}_i$. The observed residuals between the "data" image features, and the "mean" features are $\Delta_i = Y_i - \hat{Y}_i$.

The features are derived from edge data, and the underlying edge curve has an orientation angle in the image. These angles are used to define coordinate systems specific to each image feature Y_i. These coordinate systems define rotation matrices R_i that are used to transform the residuals into the coordinate systems of the features, in the following way: $\Delta'_i = R_i \Delta_i$.

The stationary covariance matrix of the matched feature fluctuations observed in the feature coordinate systems is then estimated using the Maximum Likelihood method, as follows,

$$\hat{\psi} = \frac{1}{n} \sum_i \Delta'_i \Delta'^T_i \ .$$

Here T denotes the matrix transpose operation. This technique has some bias, but for the reasonably large sample sizes involved ($n \approx 100$) the effect is minor.

The resulting covariance matrices typically indicate larger variance for deviations along the edge curve than perpendicular to it, as suggested by the data in Figures 3-5 and 3-6.

3.3.2 Specializing the Covariance

At recognition time, it is necessary to specialize the constant covariance to each image feature. This is done by rotating it to orient it with respect to the image feature.

A covariance matrix transforms like the following product of residuals:

$$\Delta_i' \Delta_i'^T \ .$$

This is transformed back to the image system as follows,

$$R_i^T \Delta_i' \Delta_i'^T R_i \ .$$

Thus the constant covariance is specialized to the image features in the following way,

$$\psi_{ij} = R_i^T \hat{\psi} R_i \ .$$

Chapter 4

Modeling Objects

What is needed from object models? For recognition, the main issue lies in predicting the image features that will appear in an image of the object. Should the object model be a monolithic 3D data structure? After all, the object itself is 3D. In this chapter, some pros and cons of monolithic 3D models are outlined. An alternative approach, interpolation of views, is proposed. The related problem of obtaining the object model data is discussed, and it is proposed that the object model data be obtained by taking pictures of the object. An automatic method for this purpose is described. Additionally, a means of edge detection that captures the average edges of an object is described.

4.1 Monolithic 3D Object Models

One motivation for using 3D object models in recognition systems is the observation that computer graphics techniques can be used to synthesize convincing images from 3D models in any pose desired.

For some objects, having a single 3D model seems a natural choice for a recognition system. If the object is polygonal, and is represented by a list of 3D line segments and vertices, then predicting the features that will appear in a given high resolution view

is a simple matter. All that is needed is to apply a pose dependent transformation to each feature, and to perform a visibility test.

For other objects, such as smoothly curved objects, the situation is different. Predicting features becomes more elaborate. In video imagery, occluding edges (or *limbs*) are often important features. Calculating the limb of a smooth 3D surface is usually complicated. Ponce and Kriegman [58] describe an approach for objects modeled by parametric surface patches. Algebraic elimination theory is used to relate image limbs to the model surfaces that generated them. Brooks' vision system, Acronym [10], also recognized curved objects from image limbs. It used generalized cylinders to model objects. A drawback of this approach is that it is awkward to realistically modeling typical objects, like telephones or automobiles, with generalized cylinders.

Predicting reduced resolution image features is another difficulty with monolithic 3D models. This is a drawback because doing recognition with reduced resolution features is an attractive strategy: with fewer features less search will be needed. One solution would be to devise a way of smoothing 3D object models such that simple projection operations would accurately predict reduced resolution edge features. No such method is known to the author.

Detecting reduced resolution image features is straightforward. Good edge features of this sort may be obtained by smoothing the grayscale image before using an edge operator. This method is commonly used with the Canny edge operator [13], and with the Marr-Hildreth operator [53].

An alternative approach is to do projections of the object model at full resolution, and then to do some kind of smoothing of the image. It isn't clear what sort of smoothing would be needed. One possibility is to do photometrically realistic projections (for example by ray tracing rendering), perform smoothing in the image, and then use the same feature detection scheme as is used on the images presented for recognition. This method is likely to be too expensive for practical recognition system that need to perform large amounts of prediction. Perhaps better ways of doing this

will be found.

Self occlusion is an additional complexity of the monolithic 3D model approach. In computer graphics there are several ways of dealing with this issue, among them hidden line and z-buffer methods. These methods are fairly expensive, at least in comparison to sparse point projections.

In summary, monolithic 3D object models address some of the requirements for predicting images for recognition, but the computational cost may be high.

4.2 Interpolation of Views

One approach to avoiding the difficulties discussed in the previous section is to use an image-based approach to object modeling. Ullman and Basri [71] have discussed such approaches. There is some biological evidence that animal vision systems have recognition subsystems that are attuned to specific views of faces [25]. This may provide some assurance that image-based approaches to recognition aren't unreasonable.

An important issue with image-based object modeling concerns how to predict image features in a way that covers the space of poses that the object may assume.

Bodies undergoing rigid motion in space have six degrees of freedom, three in translation, and three in rotation. This six parameter pose space may be split into two parts – the first part being translation and in image-plane rotations (four parameters) – the second part being out of image-plane rotations (two parameters: the "view sphere").

Synthesizing views of an object that span the first part of pose space can often be done using simple and efficient linear methods of translation, rotation, and scale in the plane. This approach can be precise under orthographic projection with scaling, and accurate enough in some domains with perspective projection. Perspective projection is often approximated in recognition systems by 3D rotation combined with orthographic projection and scaling. This has been called the *weak perspective*

approximation [70].

The second part of pose space, out of plane rotation, is more complicated. The approach advocated in this research involves tesselating the view sphere around the object, and storing a view of the object for each vertex of the tesselation. Arbitrary views will then entail, at most, small out of plane rotations from stored views. These views may be synthesized using interpolation. The Linear Combination of Views method of Ullman and Basri [71], works well for interpolating between nearby views (and more distant ones, as well).

Conceptually, the interpolation of views method caches pre-computed predictions of images, saving the expense of repeatedly computing them during recognition. If the tesselation is dense enough, difficulties owing to large changes in aspect may be avoided.

Breuel [9] advocates a view-based approach to modeling, without interpolation.

4.3 Object Models from Observation

How can object model features be acquired for use in the interpolation of views framework? If a detailed CAD model of the object is available, then views might be synthesized using graphical rendering programs (this approach was used in the (single view) laser radar experiment described in Section 7.3).

Another method is to use the object itself as its own model, and to acquire views by taking pictures of the object. This process can make use of the feature extraction method that is used on images at recognition time. An advantage of this scheme is that an accurate CAD style model isn't needed. Using the run-time feature extraction mechanism of the recognition system automatically selects the features that will be salient at recognition time, which is otherwise a potentially difficult problem.

One difficulty with the models from observation approach is that image features tend to be somewhat unstable. For example, the presence and location of edge features

is influenced by illumination conditions, as illustrated in the following figures. Figure 4-1 shows a series of nine grayscale images where the only variation is in lighting. A corresponding set of edge images is shown in 4-2. The edge operator used in preparing the images is described in Section 4.4. The standard deviation of the smoothing operator was 2 pixels.

4.4 Mean Edge Images

It was pointed out above that the instability of edge features is a potential difficulty of acquiring object model features from observation. The Mean Edge Image method solves this problem by making edge maps that are averaged over variations due to illumination changes.

Brightness edges may be characterized as the ridges of a measure of brightness variation. This is consistent with the common notion that edges are the 1D loci of maxima of changes in brightness. The edge operator used in Figure 4-2 is an example of this style of edge detector. It is a ridge operator applied to the squared discrete gradient of smoothed images. Here, the squared discrete gradient is the measure of brightness variation. This style of edge detection was described by Mercer [57]. The mathematical definition of the ridge predicate is that the gradient is perpendicular to the direction having the most negative second directional derivative. Another similar definition of edges was proposed Haralick [37]. For a general survey of edge detection methods, see *Robot Vision*, by Horn [39].

The preceding characterization of image edges generalizes naturally to mean edges. Mean edges are defined to be ridges in the average measure of brightness fluctuation. In this work, average brightness fluctuation over a set of pictures is obtained by averaging the squared discrete gradient of the (smoothed) images.

Figure 4-3 shows the averaged squared gradient of smoothed versions of the images that appear in Figure 4-1. Recall that only the lighting changed between these images.

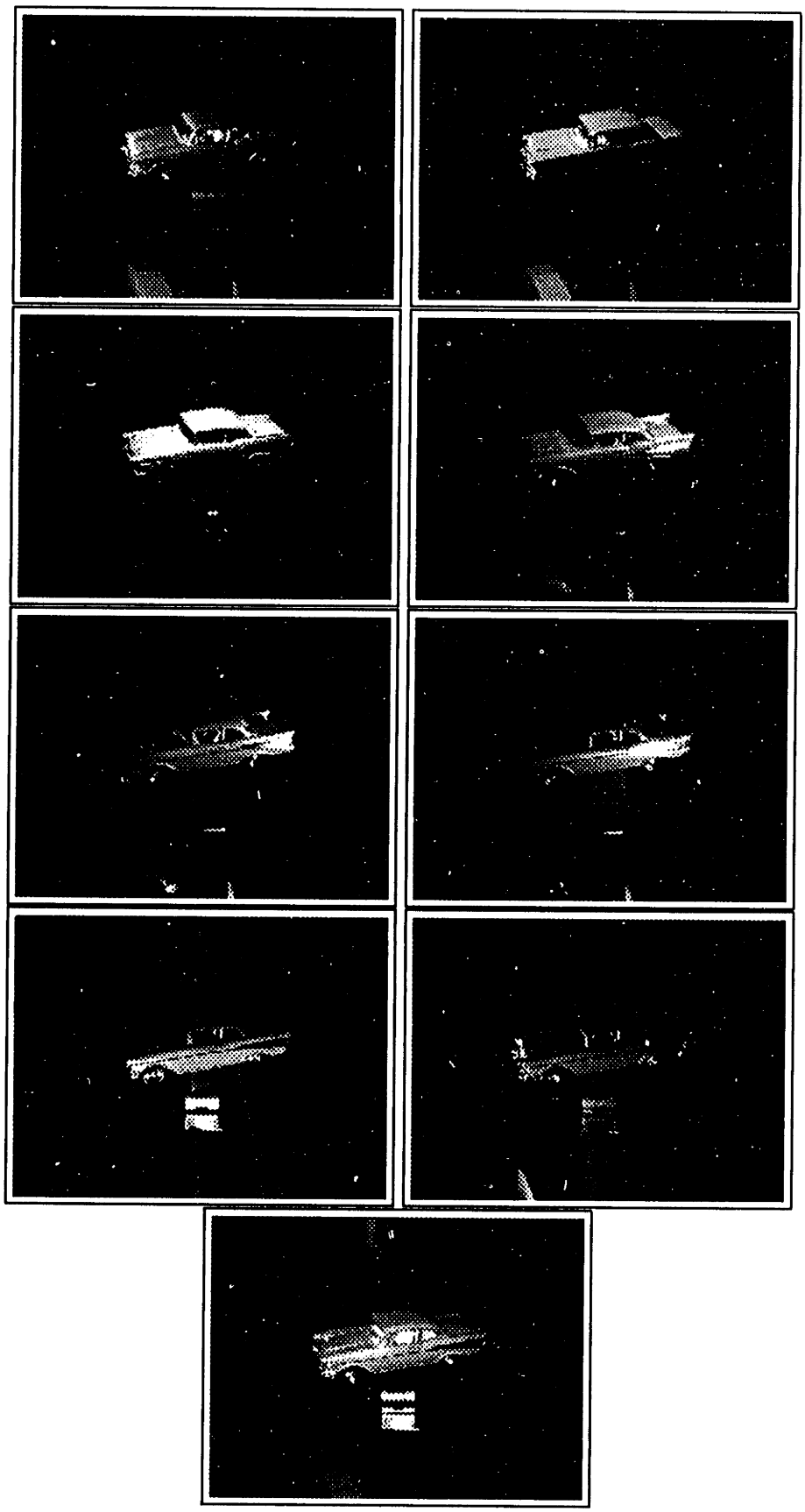


Figure 4-1: Grayscale Images

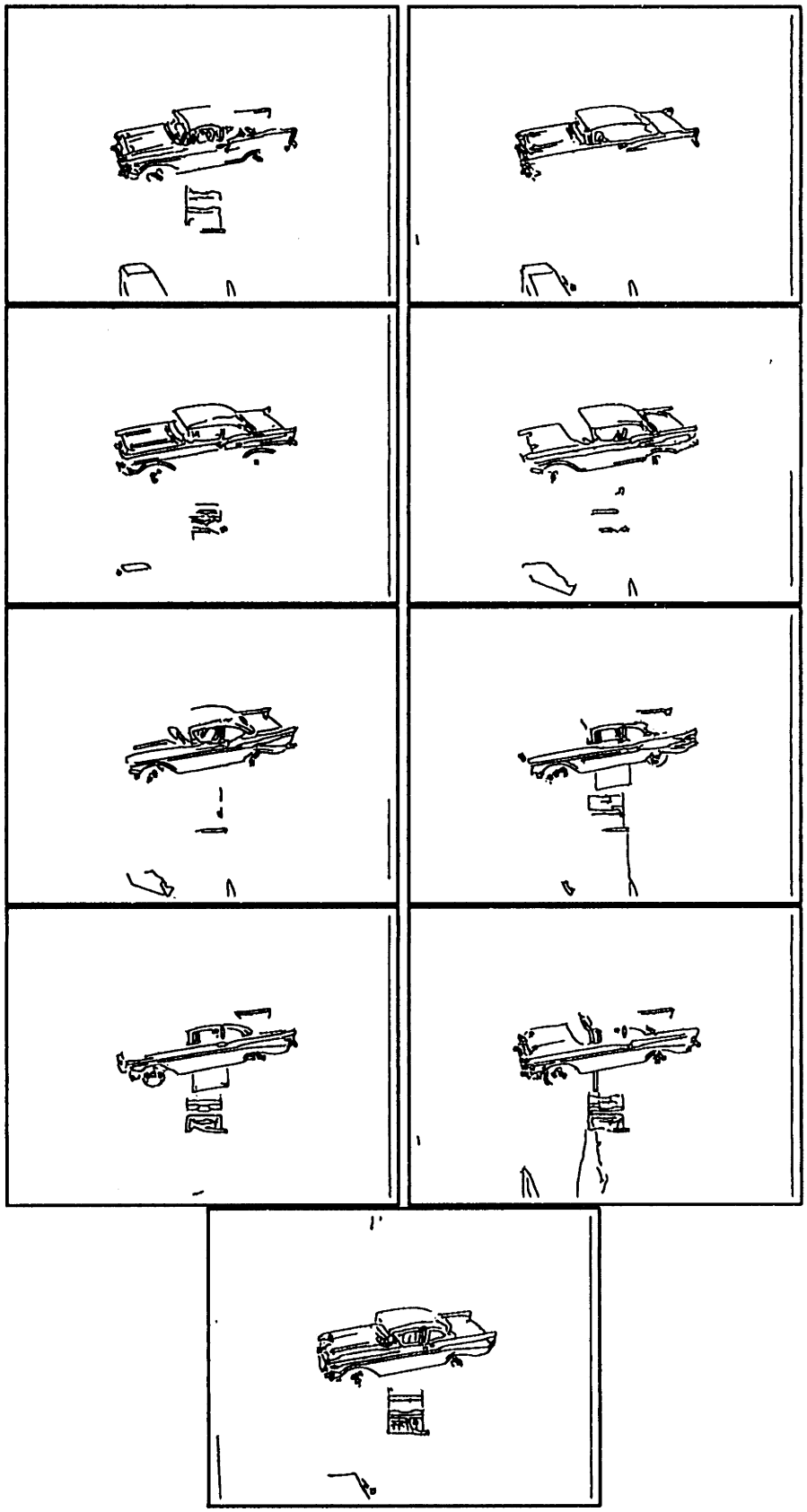


Figure 4-2: Edge Images

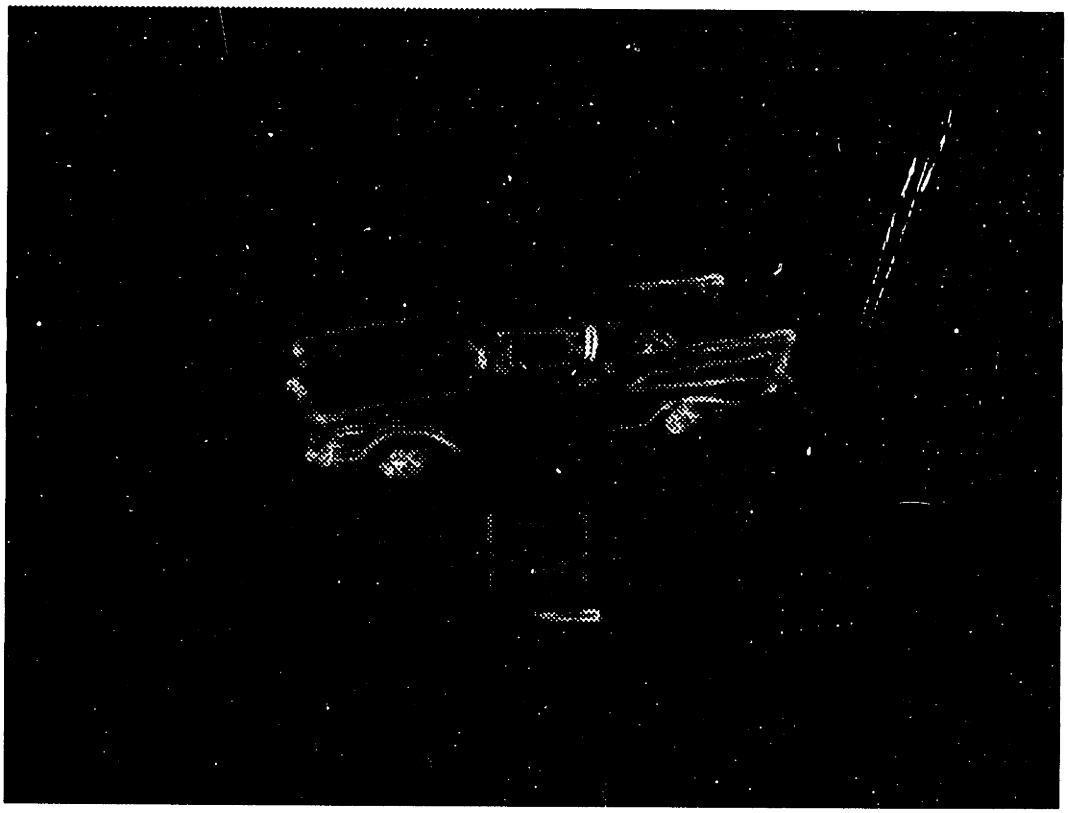


Figure 4-3: Averaged Squared Gradient of Smoothed Images

Figure 4-4 shows the ridges from the image of Figure 4-3. Hysteresis thresholding based on the magnitude of the averaged squared gradient has been used to suppress weak edges. Such hysteresis thresholding is used with the Canny edge operator. Note that this edge image is relatively immune to specular highlights, in comparison to the individual edge images of Figure 4-4.

4.5 Automatic 3D Object Model Acquisition

This section outlines a method for automatic 3D object model acquisition that combines interpolation of views and Mean Edge Images. The method involves automatically acquiring (many) pictures of the object under various combinations of pose and illumination. A preliminary implementation of the method was used to acquire object model features for the 3D recognition experiment discussed in Section 10.4.

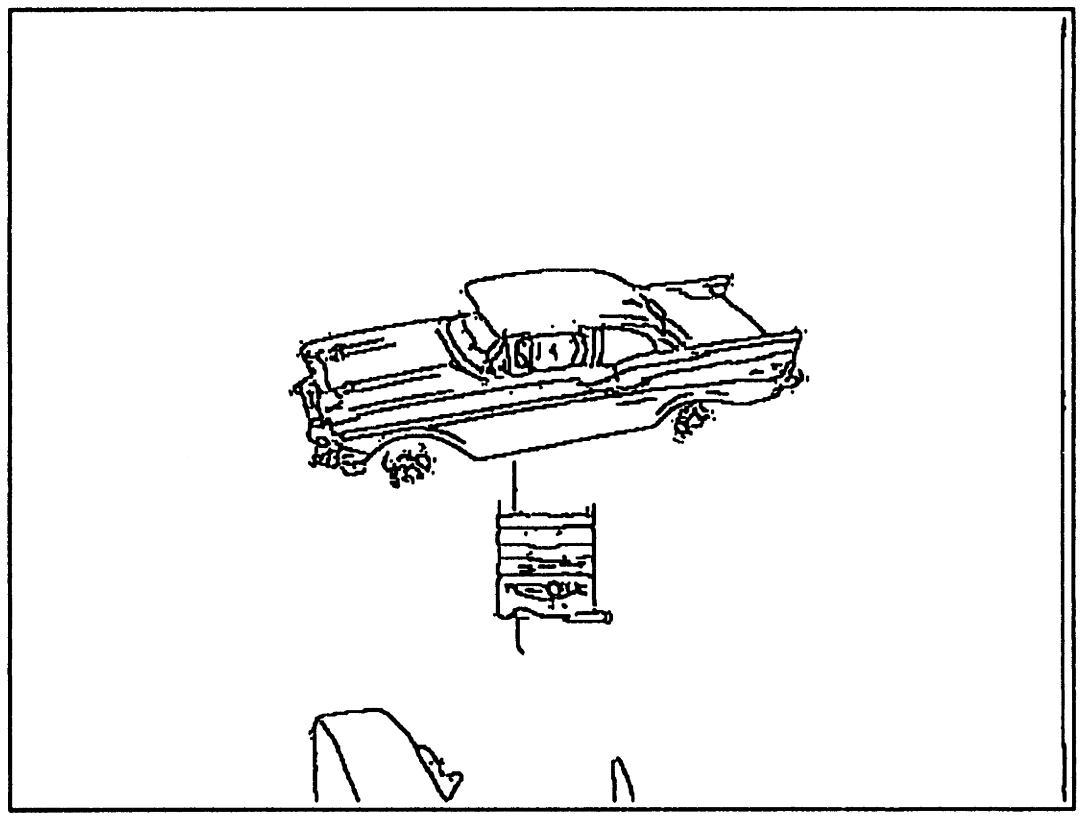


Figure 4-4: Ridges of Average Squared Gradient of Smoothed Images

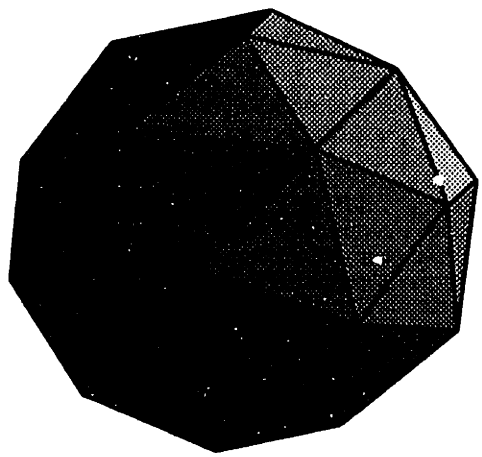


Figure 4-5: A Pentakis Dodecahedron

The object, a plastic car model, was mounted on the tool flange of a PUMA 560 robot. A video camera connected to a Sun Microsystems VFC video digitizer was mounted near the robot.

For the purpose of Interpolation of Views object model construction, the view sphere around the object was tesselated into 32 view points, the vertices of a pentakis dodecahedron (one is illustrated in Figure 4-5). At each view point a "canonical pose" for the object was constructed that oriented the view point towards the camera, while keeping the center of the object in a fixed position.

Nine different configurations of lighting were arranged for the purpose of constructing Mean Edge Images. The lighting configurations were made by moving a spotlight to nine different position that illuminated the object. The lamp positions roughly covered the view hemisphere centered on the camera.

The object was moved to the canonical poses corresponding to the 21 vertices in

the upper part (roughly 2/3) of the object's view sphere. At each of these poses, pictures were taken with each of the nine lamp positions.

Mean Edge Images at various scales of smoothing were constructed for each of the canonical poses. Object model features for recognition experiments described in Chapter 8 were derived from these Mean Edge Images. Twenty of the images from one such set of Mean Edge Images are displayed in Figures 4-6 and 4-7.

Two of these Mean Edge Images were used in an experiment in 3D recognition using a two-view Linear Combination of Views method. This method requires correspondences among features at differing views. These correspondences were established by hand, using a mouse.

It is likely that such feature correspondence could be derived from the results of a motion program. Shashua's motion program [65], which combines geometry and optical flow, was tested on images from the experimental setup and was able to establish good correspondences at the pixel level, for views separated by 4.75 degrees. This range could be increased by a sequential bootstrapping process. If correspondences can be automatically determined, then the entire process of building view-based models for 3D objects can be made fully automatic.

After performing the experiments reported in Chapter 10, it became apparent that the views were separated by too large of an angle (about 38 degrees) for establishing a good amount of feature correspondence between some views. This problem may be relieved by using more views. Using more views also makes automatic determination of correspondences easier. If the process of model construction is fully automatic, having a relatively large number of views is potentially workable.

The work of Taylor and Reeves [69] provides some evidence for the feasibility of multiple-view-based recognition. They describe a classification-based vision system that uses a library of views from a 252 vertex icosahedron-based tesselation of the view sphere. Their views were separated by 6.0 to 8.7 degrees. They report good classification of aircraft silhouettes using this approach.

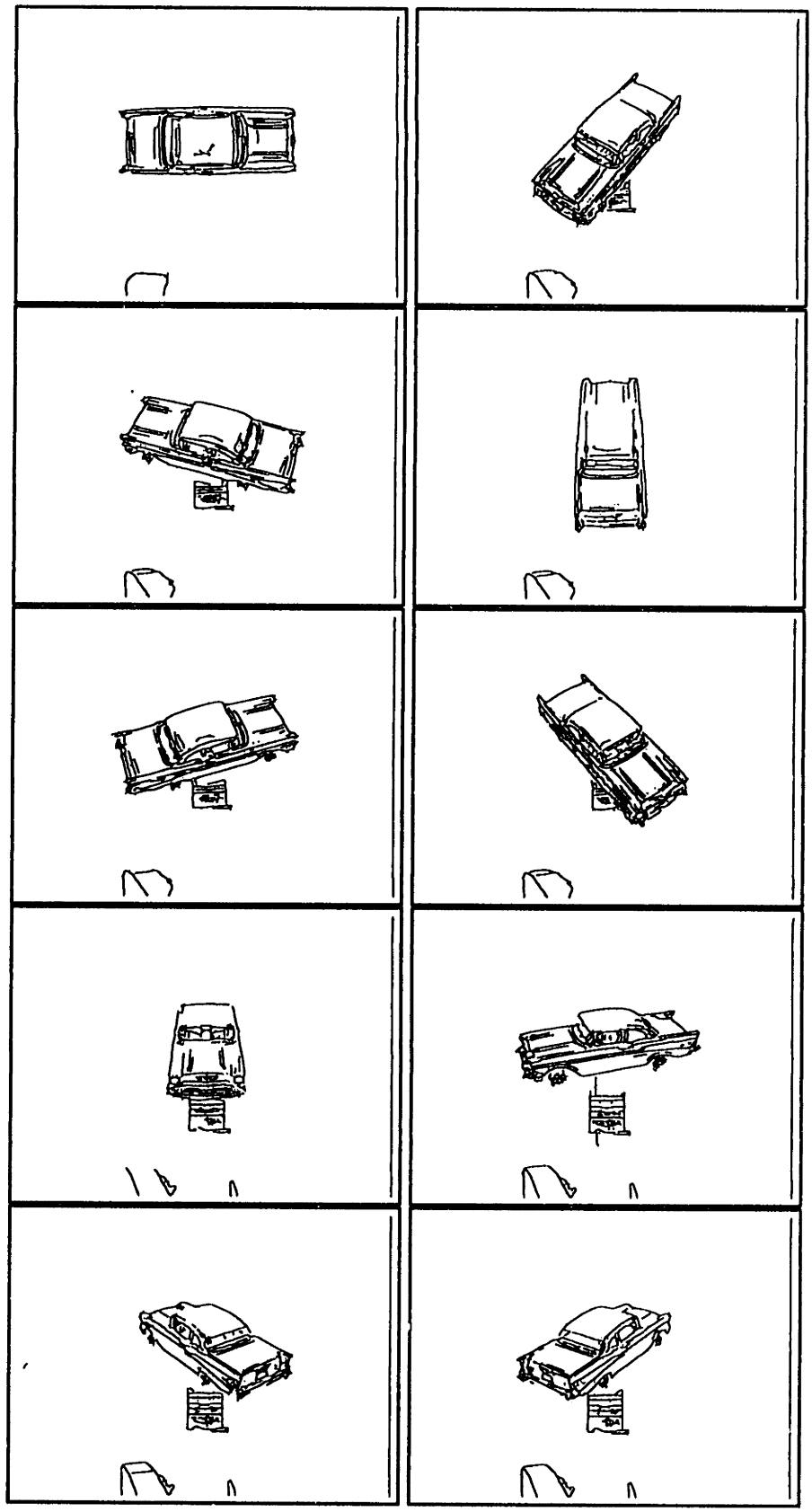


Figure 4-6: Mean Edge Images at Canonical Poses

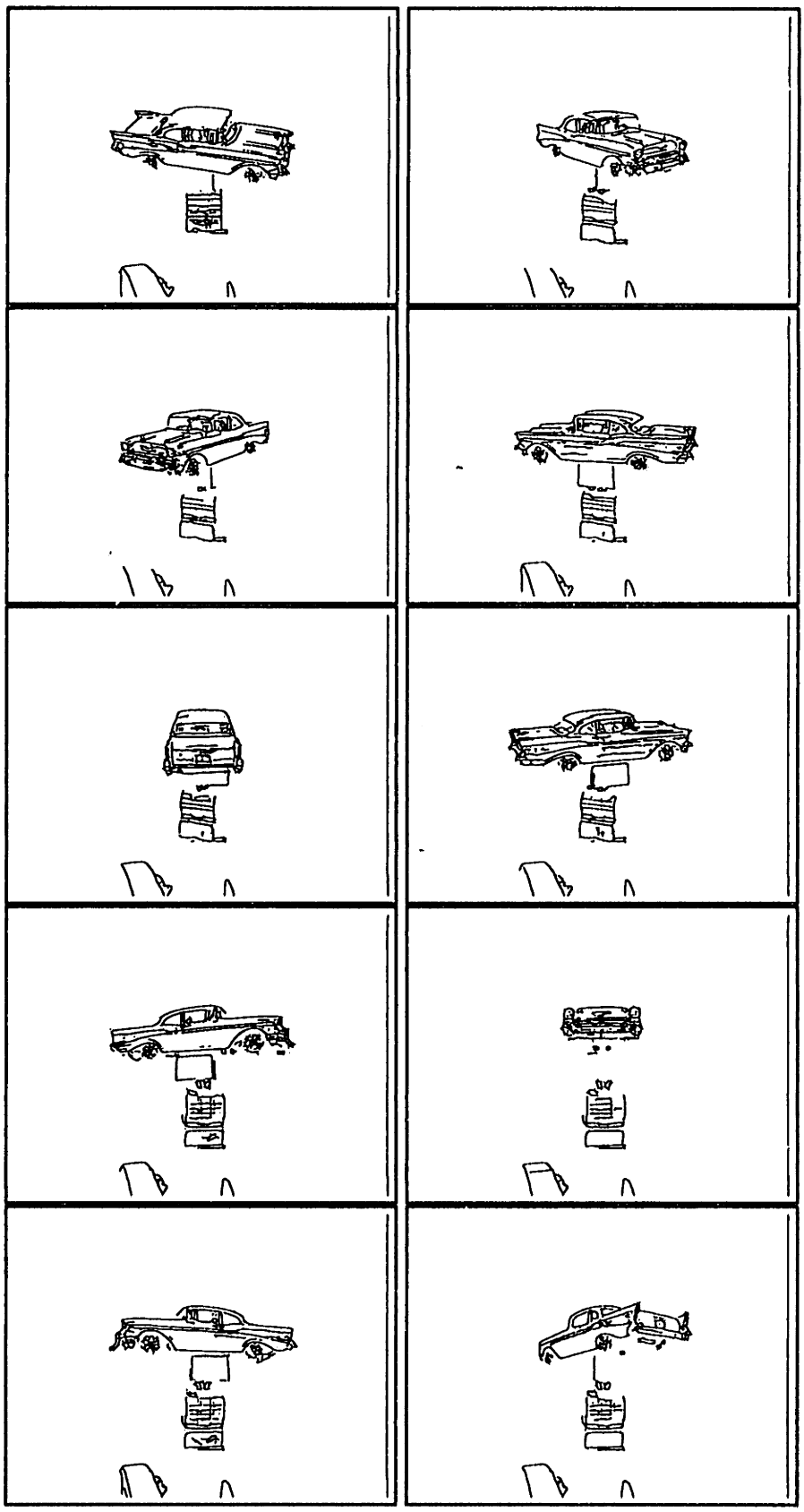


Figure 4-7: Mean Edge Images at Canonical Poses

Chapter 5

Modeling Projection

This chapter is concerned with the representations of image and object features, and with the projection of object features into the image, given the pose of the object. Four different formulations are described, three of which are used in experiments reported in other chapters.

The first three models described in this chapter are essentially 2D, the transformations comprise translation, rotation, and scaling in the plane. Such methods may be used for single views of 3D objects via the weak perspective approximation, as described in [70]. In this scheme, perspective projection is approximated by orthographic projection with scaling. Within this approximation, these methods can handle four of the six parameters of rigid body motion – everything but out of plane rotations.

The method described in Section 5.5, is based on Linear Combination of Views, a view-based 3D method that was developed by Ullman and Basri [71].

5.1 Linear Projection Models

Pose determination is often a component of model-based object recognition systems, including the systems described in this thesis. Pose determination is frequently framed

as an optimization problem. The pose determination problem may be significantly simplified if the feature projection model is linear in the pose vector. The systems described in this thesis use projection models having this property, this enables solving the embedded optimization problem using least squares. Least squares is advantageous because unique solutions may be obtained easily in closed form. This is a significant advantage, since the embedded optimization problem is solved many times during the course of a search for an object in a scene.

All of the formulations of projection described below are linear in the parameters of the transformation. Because of this they may be written in the following form:

$$\eta_i = \mathcal{P}(M_i, \beta) = M_i\beta \ . \tag{5.1}$$

The pose of the object is represented by β, a column vector, the object model feature by M_i, a matrix. η_i, the projection of the model feature into the image by pose β, is a column vector.

Although this particular form may seem odd, it a natural one if the focus is on solving for the pose and the object model features are constants.

5.2 2D Point Feature Model

The first, and simplest, method to be described was used by Faugeras and Ayache in their vision system HYPER [1]. It is defined as follows: $\eta_i = M_i\beta$, where

$$\eta_i = \begin{bmatrix} p'_{ix} \\ p'_{iy} \end{bmatrix} \qquad M_i = \begin{bmatrix} p_{ix} & -p_{iy} & 1 & 0 \\ p_{iy} & p_{ix} & 0 & 1 \end{bmatrix} \qquad \text{and} \qquad \beta = \begin{bmatrix} \mu \\ \nu \\ t_x \\ t_y \end{bmatrix} \ .$$

The coordinates of object model point i are p_{ix} and p_{iy}. The coordinates of the

model point i, projected into the image by pose β, are p'_{ix} and p'_{iy}. This transformation is equivalent to rotation by θ, scaling by s, and translation by T, where

$$T = \begin{bmatrix} t_x \\ t_y \end{bmatrix} \qquad s = \sqrt{\mu^2 + \nu^2} \qquad \theta = \arctan\left(\frac{\nu}{\mu}\right) .$$

This representation has an un-symmetrical way of representing the two classes of features, which seems odd due to their essential equivalence, however the trick facilitates the linear formulation of projection given in Equation 5.1.

In this model, rotation and scale are effected by analogy to the multiplication of complex numbers, which induces transformations of rotation and scale in the complex plane. This analogy may be made complete by noting that the algebra of complex numbers $a + ib$ is isomorphic with that of matrices of the form

$$\begin{bmatrix} a & b \\ -b & a \end{bmatrix}$$

.

5.3 2D Point-Radius Feature Model

This section describes an extension of the previous feature model that incorporates information about the normal and curvature at a point on a curve (in addition to the coordinate information).

There are advantages in using richer features in recognition – they provide more constraints, and can lead to space and time efficiencies. These potential advantages must be weighed against the practicality of detecting the richer features. For example, there is incentive to construct features incorporating higher derivative information at a point on a curve; however, measuring higher derivatives of curves derived from video imagery is probably impractical, because each derivative magnifies the noise present

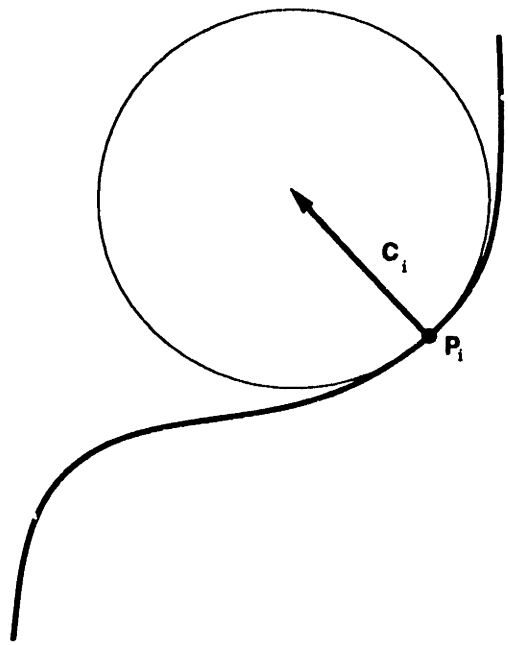


Figure 5-1: Edge Curve, Osculating Circle, and Radius Vector

in the data.

The feature described here is a compromise between richness and detectability. It is defined as follows $\eta_i = M_i\beta$, where

$$\eta_i = \begin{bmatrix} p'_{ix} \\ p'_{iy} \\ c'_{ix} \\ c'_{iy} \end{bmatrix} \qquad M_i = \begin{bmatrix} p_{ix} & -p_{iy} & 1 & 0 \\ p_{iy} & p_{ix} & 0 & 1 \\ c_{ix} & -c_{iy} & 0 & 0 \\ c_{iy} & c_{ix} & 0 & 0 \end{bmatrix} \quad \text{and} \quad \beta = \begin{bmatrix} \mu \\ \nu \\ t_x \\ t_y \end{bmatrix} .$$

The point coordinates and β are as above. c_{ix} and c_{iy} represent the radius vector of the curve's osculating circle that touches the point on the curve, as illustrated in Figure 5-1. This vector is normal to the curve. Its length is the inverse of the curvature at the point. The counterparts in the image are given by c'_{ix} and c'_{iy}. With this model, the radius vector c rotates and scales as do the coordinates p, but it does not translate. Thus, the aggregate feature translates, rotates and scales correctly.

This feature model is used in the experiments described in Sections 6.2, 7.4, and

10.1 When the underlying curvature goes to zero, the length of the radius vector diverges, and the direction becomes unstable. This has been accommodated in the experiments by truncating c. Although this violates the "transforms correctly" criterion, the model still works well.

5.4 2D Oriented-Range Feature Model

This feature projection model is very similar to the one described previously. It was designed for use in range imagery instead of video imagery. Like the previous feature, it is fitted to fragments of image edge curves. In this case, the edges label discontinuities in range. It is defined just as above in Section 5.3, but the interpretation of c is different. The point coordinates and β are as above. As above, c_{ix} and c_{iy} are a vector whose direction is perpendicular to the (range discontinuity) curve fragment. The difference is that rather than encoding the inverse of the curvature, the length of the vector encodes instead the inverse of the range at the discontinuity. The counterparts in the image are given by c'_{ix} and c'_{iy}. The aggregate feature translates, rotates and scales correctly when used with imaging models where the object features scale according to the inverse of the distance to the object. This holds under perspective projection with attached range labels when the object is small compared to the distance to the object.

This model was used in the experiments described in Section 7.3.

5.5 Linear Combination of Views

The technique used in the above methods for synthesizing rotation and scale amounts to making linear combinations of the object model with a copy of it that has been rotated 90 degrees in the plane.

In their paper, "Recognition by Linear Combination of Models" [71], Ullman and Basri describe a scheme for synthesizing views under 3D orthography with rotation

and scale that has a linear parameterization. They show that the space of images of an object is a subspace of a linear space that is spanned by the components of a few images of an object. They discuss variants of their formulation that are based on two views, and on three and more views. Recovering conventional pose parameters from the linear combination coefficients is described in [60].

The following is a brief explanation of the two-view method. The reader is referred to [71] for a fuller description. Point projection from 3D to 2D under orthography, rotation, and scale is a linear transformation. If two (2D) views are available, along with the transformations that produced them (as in stereo vision), then there is enough data to invert the transformations and solve for the 3D coordinates (three equations are needed, four are available). The resulting expression for the 3D coordinates will be a linear equation in the components of the two views. New 2D views may then be synthesized from the 3D coordinates by yet another linear transformation. Compounding these linear operations yields an expression for new 2D views that is linear in the components of the original two views. There is a quadratic constraint on the 3D to 2D transformations, due to the constraints on rotation matrices. The usual Linear Combination of Views approach makes use of the above linearity property while synthesizing new views with general linear transformations (without the constraints). This practice leads to two extra parameters that control stretching transformations in the synthesized image. It also reduces the need to deal with camera calibrations – the pixel aspect ratio may be accommodated in the stretching transformations.

The following projection model uses a two view variant of the Linear Combination of Views method to synthesize views with limited 3D rotation and scale. Additionally, translation has been added in a straightforward way. $\eta_i = M_i\beta$, where

$$\eta_i = \begin{bmatrix} \eta_{ix} \\ \eta_{iy} \end{bmatrix} \qquad M_i = \begin{bmatrix} p_{ix} & 0 & q_{ix} & 0 & p_{iy} & 0 & 1 & 0 \\ 0 & p_{iy} & 0 & q_{iy} & 0 & p_{ix} & 0 & 1 \end{bmatrix}$$

and

$$\beta = \left[\begin{array}{cccccccc} \beta_0 & \beta_1 & \beta_2 & \beta_3 & \beta_4 & \beta_5 & \beta_6 & \beta_7 \end{array} \right]^T \quad .$$

The coordinates of the i'th point in one view are p_{ix} and p_{iy}; in the other view they are q_{ix} and q_{iy}.

When this projection model is used, β does not in general describe rigid transformation, but it is nevertheless called the pose vector for notational consistency.

This method is used in the experiment described in Section 10.4.

Chapter 6

MAP Model Matching

MAP Model Matching [1] (MMM) is the first of two statistical formulations of object recognition to be discussed in this thesis. It builds on the models of features and correspondences, objects, and projection that are described in the previous chapters. MMM evaluates joint hypotheses of match and pose in terms of their posterior probability, given an image. MMM is the starting point for the second formulation of object recognition, Posterior Marginal Pose Estimation (PMPE), which is described in Chapter 7.

The MMM objective function is amenable to search in correspondence space, the space of all possible assignments from image features to model and background features. This style of search has been used in many recognition systems, and it is used here in a recognition experiment involving low resolution edge features.

It is shown that under certain conditions, searching in pose space for maxima of the MMM objective function is equivalent to robust methods of chamfer matching [47].

[1] Early versions of this work appeared in [74] and [75].

6.1 Objective Function for Pose and Correspondences

In this section an objective function for evaluating joint hypotheses of match and pose using the MAP criterion will be derived.

Briefly, probability densities of image features, conditioned on the parameters of match and pose ("the parameters"), are combined with prior probabilities on the parameters using Bayes' rule. The result is a posterior probability density on the parameters, given an observed image. An estimate of the parameters is then formulated by choosing them so as to maximize their a-posteriori probability. (Hence the term *MAP*. See Beck and Arnold's textbook [4] for a discussion of MAP estimation.) MAP estimators are especially practical when used with normal probability densities.

This research focuses on feature based recognition. The probabilistic models of image features described in Chapter 3 are used. Initially, image features are assumed to be mutually independent (this is relaxed in Section 6.1.1). Additionally, matched image features are assumed to be normally distributed about their predicted positions in the image, and unmatched (background) features are assumed to be uniformly distributed in the image. These densities are combined with a prior model of the parameters. When a linear projection model is used, a simple objective function for match and pose results.

As described in Chapter 2, the image that is to be analyzed is represented by a set of v-dimensional column vectors.

$$Y = \{Y_1, Y_2, \ldots, Y_n\} \ , \ Y_i \in R^v \ .$$

The object model is denoted by M,

$$M = \{M_1, M_2, \ldots, M_m\} \ .$$

When linear projection models are used, as discussed in Chapter 5, the object features will be represented by real matrices: $M_j \in R^{v \times z}$ (z is defined below).

The parameters to be estimated in matching are the correspondences between image and object features, and the pose of the object in the image. As discussed in Section 2.1, the state of match, or correspondences, is described by the variable Γ:

$$\Gamma = \{\Gamma_1, \Gamma_2, \ldots, \Gamma_n\} \ , \ \Gamma_i \in M \cup \{\perp\} \ .$$

Here $\Gamma_i = M_j$ means that image feature i corresponds to object model feature j, and $\Gamma_i = \perp$ means that image feature i is due to the background.

The pose of the object is a real vector: $\beta \in R^z$. A projection function, $\mathcal{P}()$, maps object model features into the v-dimensional image coordinate space according to the pose,

$$\mathcal{P}(M_i, \beta) \in R^v \ .$$

The probabilistic models of image features described in Chapter 3 may be written as follows:

$$p(Y_i \mid \Gamma, \beta) = \begin{cases} \frac{1}{W_1 W_2 \cdots W_v} & \text{if } \Gamma_i = \perp \\ G_{\psi_{ij}}(Y_i - \mathcal{P}(M_j, \beta)) & \text{if } \Gamma_i = M_j \end{cases} \tag{6.1}$$

where

$$G_{\psi_{ij}}(x) = (2\pi)^{-\frac{v}{2}} |\psi_i|^{-\frac{1}{2}} \exp\left(-\frac{1}{2} x^T \psi_i^{-1} x\right) \ .$$

Here ψ_{ij} is the covariance matrix associated with image feature i and object model feature j. Thus image features arising from the background are uniformly distributed over the image feature coordinate space (the extent of the image feature coordinate space along dimension i is given by W_i), and matched image features are normally distributed about their predicted locations in the image. In some applications ψ could be independent if i and j – an assumption that the feature statistics are stationary in the image, or ψ may depend only on i, the image feature index. The latter is the case when the oriented stationary statistics model is used (see Section 3.3).

Assuming independent features, the joint probability density on image feature coordinates may be written as follows

$$p(Y \mid \Gamma, \beta) = \prod_i p(Y_i \mid \Gamma, \beta) = \prod_{i:\Gamma_i=\perp} \frac{1}{W_1 W_2 \cdots W_v} \prod_{ij:\Gamma_i=M_j} \mathrm{G}_{\psi_{ij}}(Y_i - \mathcal{P}(M_j, \beta)) \ . \tag{6.2}$$

This assumption often holds when sensor noise dominates in feature fluctuations.

The next step in the derivation is the construction of a joint prior on correspondences and pose. In Chapter 2, probabilistic models of feature correspondences were discussed. The independent correspondence model is used here for simplicity. Use of the Markov correspondence model is discussed in the following section. The probability that image feature i belongs to the background is B_i, while the remaining probability is uniformly distributed for correspondences to the m object model features. In some situations, B_i may be a constant, independent of i. Recalling Equations 2.1 and 2.6,

$$p(\Gamma) = \prod_i p(\Gamma_i) \qquad \text{and} \qquad p(\Gamma_i) = \begin{cases} B_i & \text{if } \Gamma_i = \perp \\ \frac{1-B_i}{m} & \text{otherwise} \ . \end{cases} \tag{6.3}$$

Prior information on the pose is assumed to be supplied as a normal density,

$$p(\beta) = \mathrm{G}_{\psi_\beta}(\beta - \beta_0)$$

where

$$\mathrm{G}_{\psi_\beta}(x) = (2\pi)^{\frac{-z}{2}} |\psi_\beta|^{-\frac{1}{2}} \exp\left(-\frac{1}{2} x^T \psi_\beta^{-1} x\right) \ .$$

Here ψ_β is the covariance matrix of the pose prior and z is the dimensionality of the pose vector, β. With the combination of normal pose priors and linear projection models the system is closed in the sense that the resulting pose estimate will also be normal. This is convenient for coarse-fine, as discussed in Section 6.4. If little is known about the pose a-priori, the prior may be made quite broad. This is expected to be often the case. If nothing is known about the pose beforehand, the pose prior

may be left out. In that case the resulting criterion for evaluating hypotheses will be based on Maximum Likelihood for pose, and on MAP for correspondences.

Assuming independence of the correspondences and the pose (before the image is compared to the object model), a mixed joint probability function may be written as follows,

$$p(\Gamma, \beta) = \mathrm{G}_{\psi_\beta}(\beta - \beta_0) \prod_{i:\Gamma_i=\perp} B_i \prod_{i:\Gamma_i\neq\perp} \frac{1 - B_i}{m} \ .$$

This a good assumption when view-based approaches to object modeling are used (these are discussed in Chapter 4 and used in the experiments described in Chapter 10). (With general 3D rotation it is inaccurate, as the visibility of features depends on the orientation of the object.) This probability function on match and pose is now used with Bayes' rule as a prior for obtaining the posterior probability of Γ and β:

$$p(\Gamma, \beta \mid Y) = \frac{p(Y \mid \Gamma, \beta) p(\Gamma, \beta)}{p(Y)} \ , \tag{6.4}$$

where $p(Y) = \sum_\Gamma \int d\beta \ p(Y \mid \Gamma, \beta) p(\Gamma, \beta)$ is a normalization factor that is formally the probability of the image. It is a constant with respect to Γ and β, the parameters being estimated.

The MAP strategy is used to obtain estimates of the correspondences and pose by maximizing their posterior probability with respect to Γ and β, as follows

$$\widehat{\Gamma, \beta} = \arg\max_{\Gamma, \beta} p(\Gamma, \beta \mid Y) \ .$$

For convenience, an objective function, L, is introduced that is a scaled logarithm of $p(\Gamma, \beta \mid Y)$. The same estimates will result if the maximization is instead carried out over L.

$$\widehat{\Gamma, \beta} = \arg\max_{\Gamma, \beta} L(\Gamma, \beta)$$

where

$$L(\Gamma, \beta) \equiv \ln \left(\frac{p(\Gamma, \beta \mid Y)}{C} \right) \ . \tag{6.5}$$

The denominator in Equation 6.5 is a constant that has been chosen to cancel constants from the numerator. Its value, which is independent of Γ and β is

$$C = \frac{B_1 B_2 \cdots B_n}{(W_1 W_2 \cdots W_v)^n} (2\pi)^{\frac{-s}{2}} |\psi_\beta|^{\frac{-1}{2}} \frac{1}{p(Y)} \ .$$

After some manipulation the objective function may be expressed as

$$L(\Gamma,\beta) = -\frac{1}{2}(\beta-\beta_o)^T \psi_\beta^{-1}(\beta-\beta_0) + \sum_{ij:\Gamma_i=M_j} [\lambda_{ij} - \frac{1}{2}(Y_i - \mathcal{P}(M_j,\beta))^T \psi_{ij}^{-1}(Y_i - \mathcal{P}(M_j,\beta))] \tag{6.6}$$

, where

$$\lambda_{ij} = \ln\left(\frac{1}{(2\pi)^{\frac{s}{2}} m} \frac{(1-B_i)}{B_i} \frac{W_1 W_2 \cdots W_v}{|\psi_{ij}|^{\frac{1}{2}}}\right) \ . \tag{6.7}$$

When a linear projection model is used, $\mathcal{P}(M_j,\beta) = M_j\beta$. (Linear projection models were discussed in Chapter 5.) In this case, the objective function takes the following simple form

$$L(\Gamma,\beta) = -\frac{1}{2}(\beta-\beta_o)^T \psi_\beta^{-1}(\beta-\beta_0) + \sum_{ij:\Gamma_i=M_j} [\lambda_{ij} - \frac{1}{2}(Y_i - M_i\beta)^T \psi_{ij}^{-1}(Y_i - M_j\beta)] \ . \tag{6.8}$$

When the background probability is constant, and when the feature covariance matrix determinant is constant (as when oriented stationary statistics are used), the formulas simplify further -

$$\lambda = \ln\left(\frac{1}{(2\pi)^{\frac{s}{2}} m} \frac{(1-B)}{B} \frac{W_1 W_2 \cdots W_v}{|\hat{\psi}|^{\frac{1}{2}}}\right) \ , \tag{6.9}$$

and

$$L(\Gamma,\beta) = -\frac{1}{2}(\beta-\beta_o)^T \psi_\beta^{-1}(\beta-\beta_0) + \sum_{ij:\Gamma_i=M_j} [\lambda - \frac{1}{2}(Y_i - M_j\beta)^T \psi_i^{-1}(Y_i - M_j\beta)] \ . \tag{6.10}$$

Here, $\hat{\psi}$ is the stationary feature covariance matrix, and ψ_i is the specialized feature covariance matrix. These were discussed in Section 3.3.

The first term of the objective function of Equation 6.8 expresses the influence of the prior on the pose. As discussed above, when a useful pose prior isn't available, this term may be dropped.

The second term has a simple interpretation. It is a sum taken over those image features that are matched to object model features. The λ_{ij} are fixed rewards for making correspondences, while the quadratic forms are penalties for deviations of observed image features from their expected positions in the image. Thus the objective function evaluates the amount of the image explained in terms of the object, with penalties for mismatch. This objective function is particularly simple in terms of β. When Γ is constant, β and its (posterior) covariance are estimated by weighted least squares. When using an algorithm based on search in correspondence space, the estimate of β can be cheaply updated by using the techniques of sequential parameter estimation. (See Beck and Arnold [4].) The λ_{ij} describe the relative value of a match component or extension in a way that allows direct comparison to the entailed mismatch penalty. The values of these trade-off parameter(s) are supplied by the theory (in Equation 6.7) and are given in terms of measurable domain statistics.

The form of the objective function suggests an optimization strategy: make correspondences to object features in order to accumulate correspondence rewards while avoiding penalties for mismatch. It is important that the λ_{ij} be positive, otherwise a winning strategy is be to make no matches to the object at all. This condition defines a critical level of image clutter, beyond which the MAP criteria assigns the feature to the background. λ_{ij} describes the dependence of the value of matches on the amount of background clutter. If background features are scarce, then correspondences to object features become more important.

This objective function provides a simple and uniform way to evaluate match and pose hypotheses. It captures important aspects of recognition: the amount of image explained in terms of the object, as well as the metrical consistency of the hypothesis; and it trades them off in a rational way based on domain statistics. Most

previous approaches have not made use of both criteria simultaneously in evaluating hypotheses, thereby losing some robustness.

6.1.1 Using the Markov Correspondence Model

When the Markov correspondence model of Section 2.3 is used instead of the independent correspondence model, the functional form of the objective function of Equation 6.6 remains essentially unchanged, aside from gaining a new term that captures the influence of the interaction of neighboring features. The names of some of the constants changes, reflecting the difference between Equations 2.2 and 2.4. Noting that $p(\Gamma, \beta \mid Y)$ is linear in $p(\Gamma)$, it can be seen that the new term in the logarithmic objective function will be:

$$\sum_{i=1}^{n-1} \ln r_i(\Gamma_i, \Gamma_{i+1}) \ .$$

As before, when an algorithm based on search in correspondence space is used, the estimate of β can still be cheaply updated. A change in an element of correspondence, some Γ_i, will now additionally entail the update of two of the terms in the expression above.

6.2 Experimental Implementation

In this section an experiment demonstrating the use of the MMM objective function is described. The intent is to demonstrate the utility of the objective function in a domain of features that have significant fluctuations. The features are derived from real images. The domain is matching among features from low-resolution edge images. The point-radius feature model discussed in Section 5.3 is used. Oriented stationary statistics, as described in Section 3.3, are used to model the feature fluctuations, so that $\lambda_{ij} = \lambda_i$.

6.2.1 Search in Correspondence Space

Good solutions of the objective function of Equation 6.8 are sought by a search in correspondence space. Search over the whole exponential space is avoided by heuristic pruning.

An objective function that evaluates a configuration of correspondences, or match (described by Γ), may be obtained as follows:

$$\mathcal{L}(\Gamma) = \max_{\beta} L(\Gamma, \beta) \ .$$

This optimization is quadratic in β and is carried out by least squares. Sequential techniques are used so that the cost of extending a partial match by one correspondence is $O(1)$.

The space of correspondences may be organized as a directed-acyclic-graph (DAG) by the following parent-child relation on matches. A point in correspondence space, or *match* is a child of another match if there is some i such that $\Gamma_i = \perp$ in the parent, and $\Gamma_i = M_j$, for some j, in the child, and they are otherwise the same. Thus, the child has one more assignment to the model than the parent does. This DAG is rooted in the match where all assignments are to the background. All possible matches are reachable from the root. A fragment of an example DAG of this kind is illustrated in Figure 6-1. Components of matches that are not explicit in the figure are assigned to the background.

Heuristic beam search, as described in [64], is used to search over matches for good solutions of $\mathcal{L}$. Success depends on the heuristic that there aren't many impostors in the image. An impostor is a set of image features that scores well but isn't a subset of the optimum match implied by the objective function. Another way of stating the heuristic is that the best match to $n + 1$ object features is likely to contain the best match to n object features.

The search method used in the experiments employs a bootstrapping mechanism

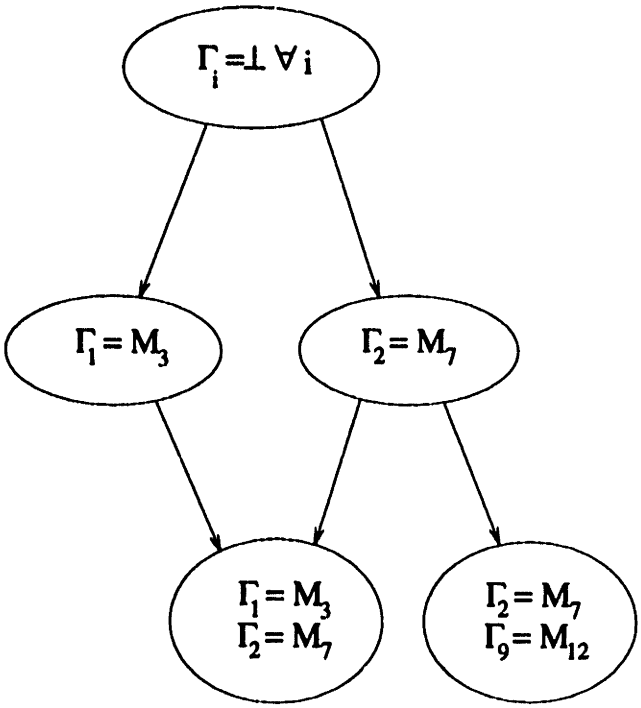


Figure 6-1: Fragment of Correspondence Space DAG

based on distinguished features. Object features 1, 2 and 3 are special, and must be detected. The scheme could be made robust by considering more initial triples of object features. Alternatively, indexing methods could be used as an efficient and robust means to initiate the search. Indexing methods are described by Clemens and Jacobs [19], and in Section 9.1.

The algorithm that was used is outlined below.

BEAM-SEARCH(M, Y)

CURRENT $\leftarrow$ {Γ: exactly one image feature is matched to each of M_1 M_2 and M_3}

;; the rest are assigned to the background.

Prune CURRENT according to $\mathcal{L}$. Keep 50 best.

Iterate to Fixpoint:

Add to CURRENT all children of members of CURRENT

Prune CURRENT according to $\mathcal{L}$. Keep N best.

;; N is reduced from 20 to 5 as the search proceeds.

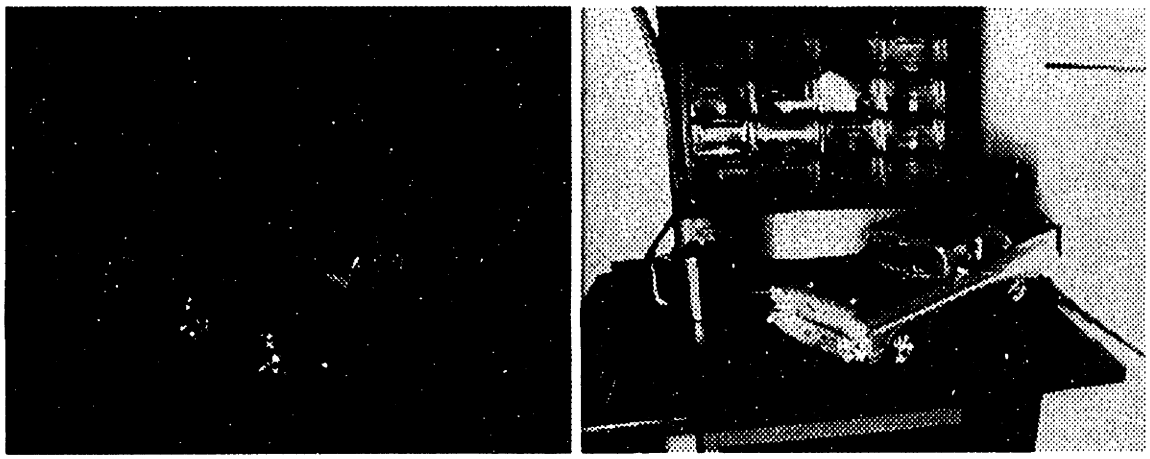


Figure 6-2: Images used for Matching

Return(CURRENT)

Sometimes an extension of a match will produce one that is already in CURRENT, that was reached in a different sequence of extensions. When this happens, the matches are coalesced. This condition is efficiently detected by testing for near equality of the scores of the items in CURRENT. Because the features are derived from observations containing some random noise, it is very unlikely that two hypotheses having differing matches will achieve the same score, since the score is partly based on summed squared errors.

6.2.2 Example Search Results

The search method described in the previous section was used to obtain good matches in a domain of features that have significant fluctuations. The features were derived from real images. A linear projection model was used.

Images used for matching are shown in Figure 6-2. The object model was derived from a set of 16 images, of which the image on the left is an example. In this set, only the light source position varied. The image features used in the search were derived from the image on the right.

The features used for matching were derived from the edge maps shown in Figure

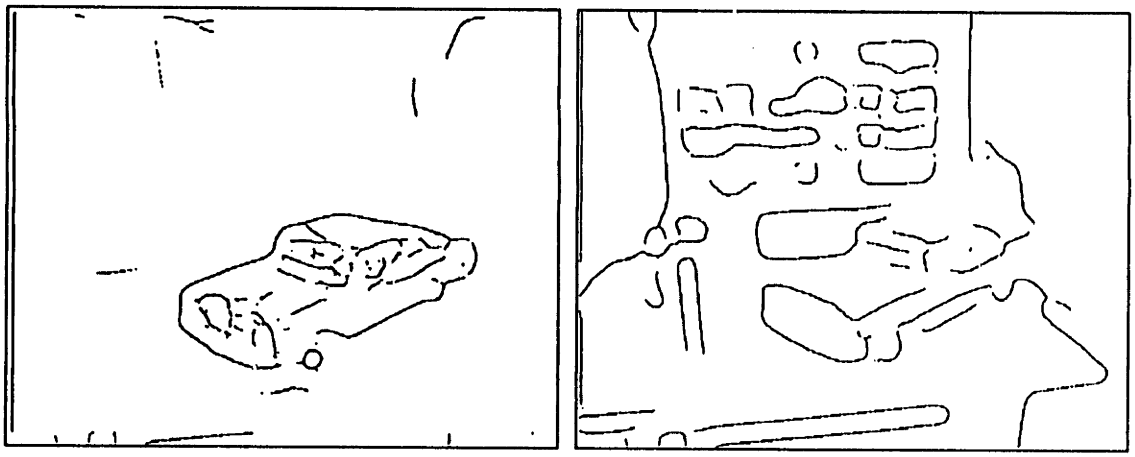


Figure 6-3: Edge Maps used for Matching

6-3. The image on the left shows the object model edges and the image on the right shows the image edges. These edges are from the Canny edge detector [13]. The smoothing standard deviation is eight pixels – these are low resolution edge maps. The object model edges were derived from a set of 16 edge maps, corresponding to the 16 images described above. The object model edges are essentially the mean edges with respect to fluctuations induced by variations in lighting. (Low resolution edges are sensitive to lighting.) They are similar to the Mean Edge Images described in Section 4.4.

The features used in matching are shown in Figure 6-4. These are point-radius features, as described in Section 5.3. The point coordinates of the features are indicated by dots, while the normal vector and curvature are illustrated by arc fragments. Each feature represents 30 edge pixels. The 40 object features appear in the upper picture, the 125 image features lower picture. The distinguished features used in the bootstrap of the search are indicated with circles. The object features have been transformed to a new pose to insure generality.

The parameters that appear in the objective function are: B, the background probability and $\hat{\psi}$, the stationary feature covariance. These were derived from a match done by hand in the example domain. The oriented stationary statistics model of Section 3.3 was used here. (A normal model of feature fluctuations is implicit in

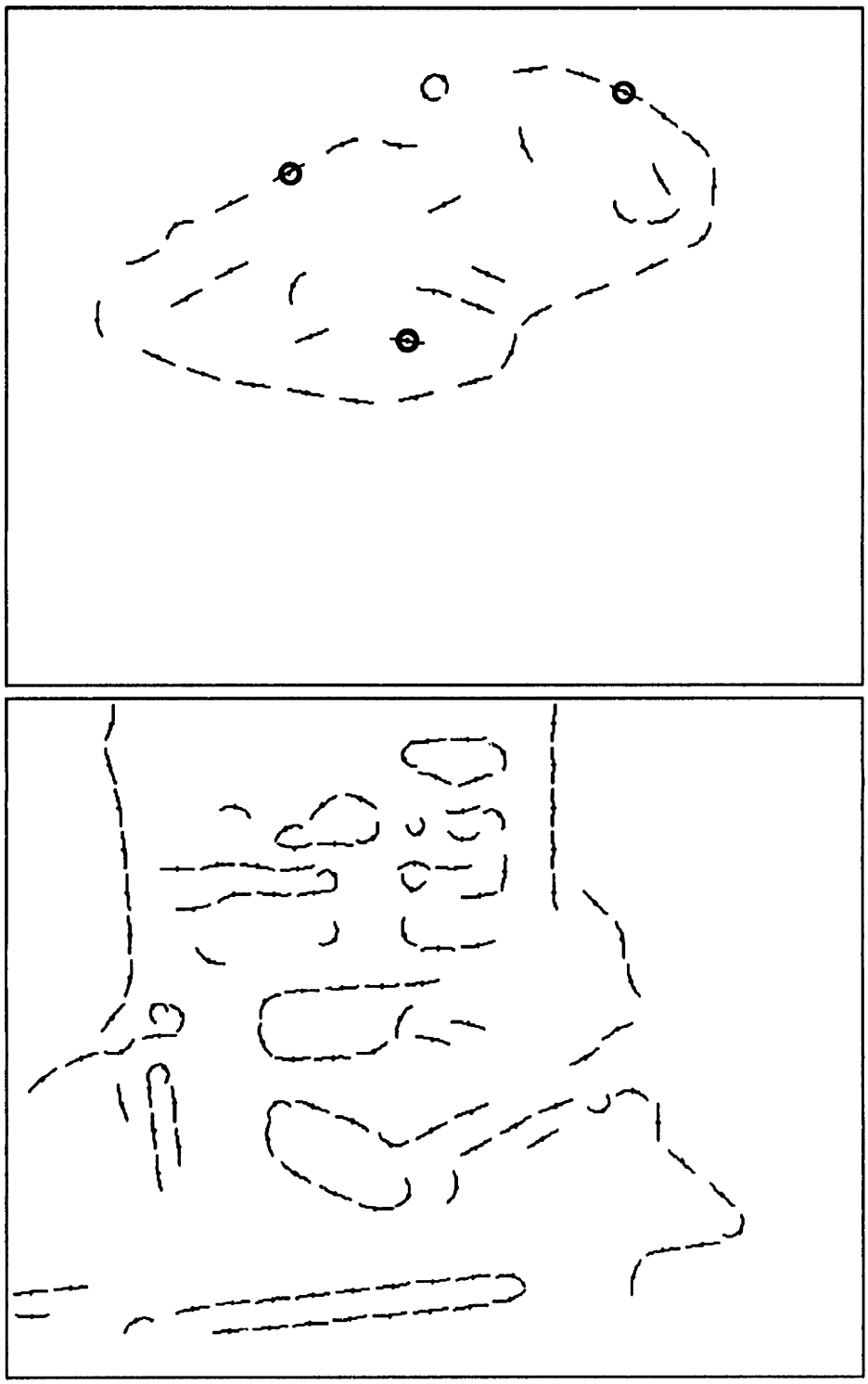


Figure 6-4: Point-Radius Features used for Matching

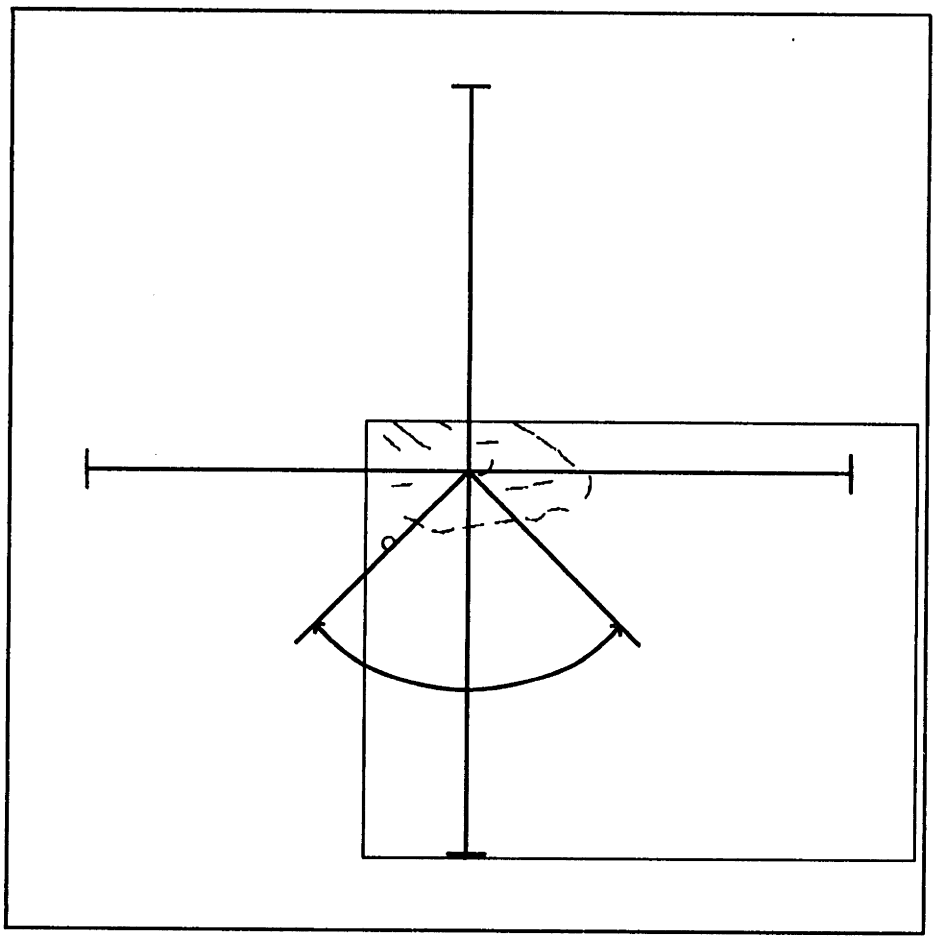


Figure 6-5: Pose Prior used in Search

the objective function of Equation 6.8. This was found to be a good model in this domain.)

A loose pose prior was used. This pose prior is illustrated in Figure 6-5. The prior places the object in the upper left corner of the image. The one standard deviation intervals of position and angle are illustrated. The one standard deviation variation of scale is 30 percent. The actual pose of the object is within the indicated one standard deviation bounds. This prior was chosen to demonstrate that the method works well despite a loose pose prior.

The best results of the beam search appear in Figure 6-6. In the upper image, the object features are delineated with heavy lines. They are located according to the pose associated with the best match. In the lower image, the object features and

image features are illustrated, while the 18 correspondences associated with the best match appear as heavy lines and dots.

The object features located according to the poses associated with the five best matches are seen in Figure 6-7. The results are difficult to distinguish because the poses are very similar.

6.3 Search in Pose Space

This section will explore searching the MMM objective function in pose space. Connections to robust chamfer matching will be described.

A pose estimate is sought by ordering the search for maxima of the MMM objective function as follows,

$$\hat{\beta} = \arg\max_{\beta} \max_{\Gamma} L(\Gamma, \beta) \ .$$

Substituting the objective function from Equation 6.6 yields

$$\hat{\beta} = \arg\max_{\beta} \max_{\Gamma} \sum_{ij:\Gamma_i=M_j} [\lambda_{ij} - \frac{1}{2}(Y_i - \mathcal{P}(M_j,\beta))^T \psi_{ij}^{-1}(Y_i - \mathcal{P}(M_j,\beta))] \ .$$

The pose prior term has been dropped in the interest of clarity. It would be easily retained as an additional quadratic term.

This equation may be simplified with the following definition,

$$D_{ij}(x) \equiv \frac{1}{2} x^T \psi_{ij}^{-1} x \ .$$

$D_{ij}(x)$ may be thought of as a generalized squared distance between observed and predicted features. It has been called the squared Mahalonobis distance [22].

The pose estimator may now be written as

$$\hat{\beta} = \arg\max_{\beta} \max_{\Gamma} \sum_{ij:\Gamma_i=M_j} [\lambda_{ij} - D_{ij}(Y_i - \mathcal{P}(M_j,\beta))] \ ,$$

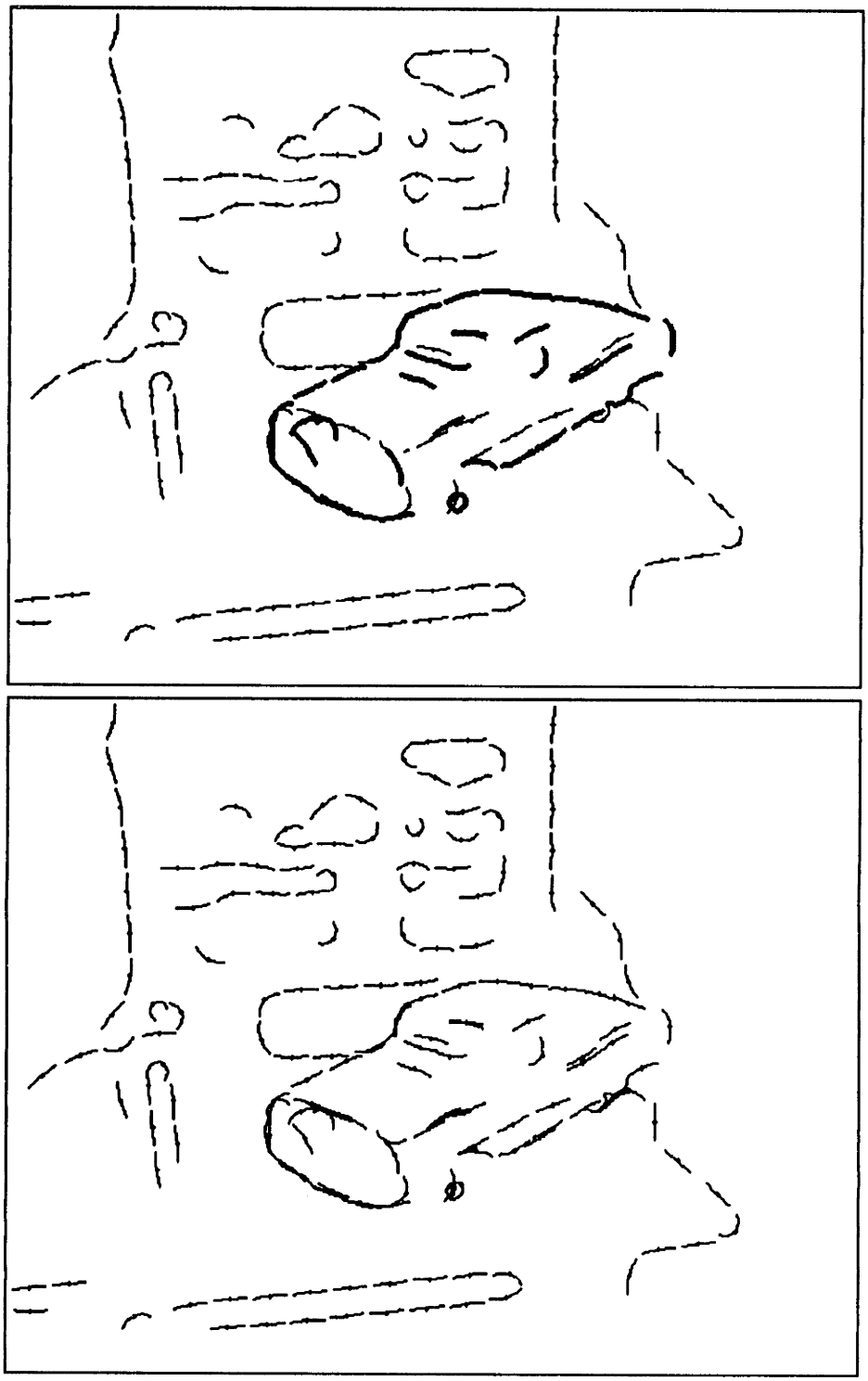


Figure 6-6: Best Match Results: Pose and Correspondences

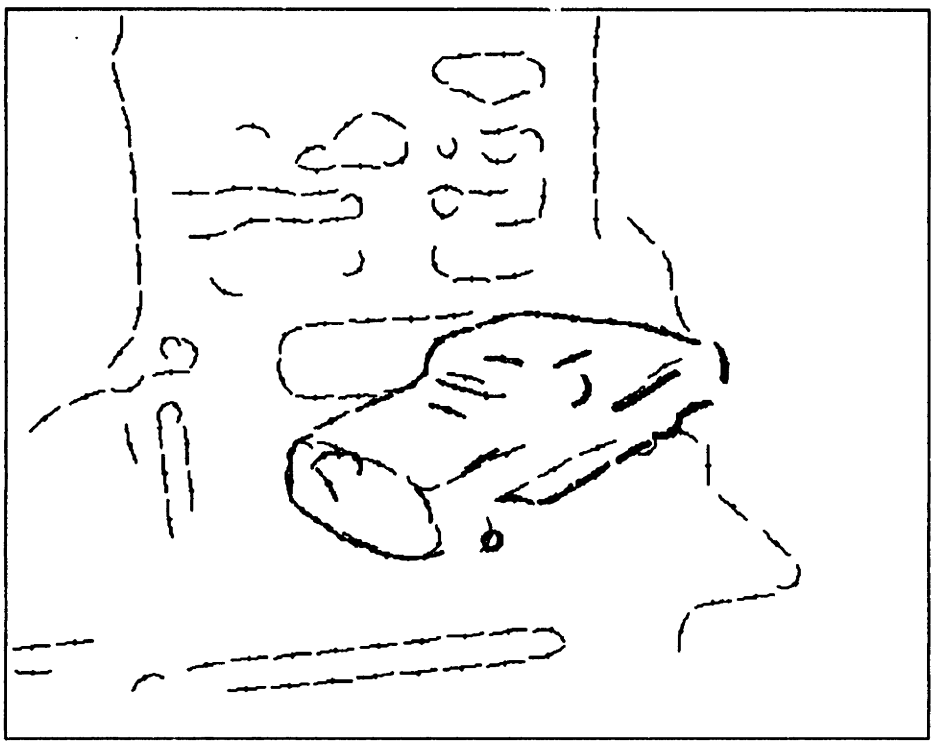


Figure 6-7: Best Five Match Results

or equivalently, as a minimization rather that maximization,

$$\hat{\beta} = \arg\min_{\beta} \min_{\Gamma} \sum_{ij:\Gamma_i=M_j} [D_{ij}(Y_i - \mathcal{P}(M_j, \beta)) - \lambda_{ij}] \ .$$

The sum is taken over those image features that are assigned to model features (not the background) in the match. It may be re-written in the following way,

$$\hat{\beta} = \arg\min_{\beta} \sum_{i} \min_{\Gamma_i} \begin{cases} 0 & \text{if } \Gamma_i = \perp \\ D_{ij}(Y_i - \mathcal{P}(M_j, \beta)) - \lambda_{ij} & \text{if } \Gamma_i = M_j \end{cases} ,$$

or as

$$\hat{\beta} = \arg\min_{\beta} \sum_{i} \min(0, \min_{j} D_{ij}(Y_i - \mathcal{P}(M_j, \beta)) - \lambda_{ij}) \ .$$

If the correspondence reward is independent of the model feature (this holds when oriented stationary statistics are used), $\lambda_{ij} = \lambda_i$. In this case, λ_i may be added to

each term in the sum without affecting the minimizing pose, yielding the following form for the pose estimator,

$$\hat{\beta} = \arg\min_{\beta} \sum_{i} \min(\lambda_i, \min_{j} D_{ij}(Y_i - \mathcal{P}(M_j, \beta))) \ . \tag{6.11}$$

This objective function is easily interpreted – it is the sum, taken over image features of a saturated penalty. The penalty (before saturation) is the smallest generalized squared distance from the observed image feature to some projected model feature. The penalty $\min_j D_{ij}(x - \mathcal{P}(M_j, \beta))$ has the form of a Voronoi surface, as described by Huttenlocher et. al. [42]. They describe a measure of similarity on image patterns, the Hausdorff distance, that is the upper envelope (maximum) of Voronoi surfaces. The measure used here differs in being saturated, and by using the sum of Voronoi surfaces, rather than the upper envelope. In their work, the upper envelope offers some reduction in the complexity of the measure, and facilitates the use of methods of computational geometry for explicitly computing the measure in 2 and 3 dimensional spaces.

Computational geometry methods might be useful for computing the objective function of Equation 6.11. In higher dimensional pose spaces (4 or 6, for example) KD-tree methods may be the only such techniques currently available. Breuel has used KD-tree search algorithms in feature matching.

Next a connection will be shown between MMM search in pose space and a method of robust chamfer matching. First, the domain of MMM is simplified in the following way. Full stationarity of feature fluctuations is assumed (as covered in Section 3.3). Further, the feature covariance is assumed to be isotropic. With these assumptions we have $\psi_{ij} = \sigma^2 I$, and $D_{ij} = \frac{1}{2\sigma^2}|x|^2$. Additionally, assuming constant background probability, we have $\lambda_{ij} = \lambda$. The pose estimator of Equation 6.11 may now be

written in the following simplified form,

$$\hat{\beta} = \arg\min_{\beta} \sum_{i} \min(\lambda, \min_{j}(\frac{1}{2\sigma^2} |Y_i - \mathcal{P}(M_j, \beta)|^2)) \ .$$

When the projection function is linear, invertible, and distance preserving, (2D and 3D rigid transformations satisfy these properties), the estimator may be expressed as follows,

$$\hat{\beta} = \arg\min_{\beta} \sum_{i} \min(\lambda, \min_{j}(\frac{1}{2\sigma^2} |\mathcal{P}^{-1}(Y_i, \beta) - M_j|^2)) \ .$$

This may be further simplified to

$$\hat{\beta} = \arg\min_{\beta} \sum_{i} \min(\lambda, d^2(\mathcal{P}^{-1}(Y_i, \beta))) \ , \tag{6.12}$$

by using the following definition of a minimum distance function.

$$d(x) \equiv \frac{1}{\sqrt{2}\sigma} \min_{j} |x - M_j| \ . \tag{6.13}$$

Chamfering methods may be used to tabulate approximations of $d^2(x)$ in an image-like array that is indexed by pixel coordinates. Chamfer-based approaches to image registration problems use the array to facilitate fast evaluation of pose objective functions. Barrow et al. [3] describe an early method where the objective function is the sum over model features of the distance from the projected model feature to the nearest image feature. Borgefors [8] recommends the use of RMS distance rather than summed distance in the objective function.

Recently, Jiang et al. [47] described a method of robust chamfer matching. In order to make the method less susceptible to disturbance by outliers and occlusions, they added saturation to the RMS objective function of Borgefors. Their objective

function has the following form

$$\frac{1}{3}(\frac{1}{n}\sum_j \min(t^2, d_j^2))^{\frac{1}{2}} \ ,$$

where d_j^2 is the squared distance from the j'th projected model point to the nearest image point. Aside from the constants and square root, which don't affect the minimizing pose, this objective function is equivalent to Equation 6.12 if the role of image and model features is reversed, and the sense of the projection function is inverted. Jiang et al. show impressive results using robust chamfer matching to register multi-modal 3D medical imagery.

6.4 Extensions

MAP Model Matching performs well on low resolution imagery in which feature uncertainty is significant. It could be used to bootstrap a coarse-fine approach to model matching, yielding good results with reasonable running times. Coarse-fine approaches have proven successful in stereo matching applications. (See Grimson [33] and Barnard [2].) A coarse-fine strategy is straightforward in the framework described here. In a hierarchy, the pose estimate from solving the objective function at one scale is used as a prior for the estimation at the next. Having a good prior on the pose will greatly reduce the amount of searching required at high resolution.

Finding a tractable model that incorporates pose dependent visibility conditions would be useful for applying MMM in non view-based recognition.

6.5 Related Work

The HYPER vision system of Ayache and Faugeras [1] uses sequential linear-least-squares pose estimation as well as the linear 2D point feature and projection model described in Section 5.2. HYPER is described as a search algorithm. Different criteria

are used to evaluate candidate matches and to evaluate competing "whole" hypotheses. An ad hoc threshold is used for testing a continuous measure of the metrical consistency of candidate match extensions. Whole match hypotheses are evaluated according to the amount of image feature accounted for – although not according to overall metrical consistency. HYPER works well on real images of industrial parts.

Goad outlined a Bayesian strategy of match evaluation based on feature and background statistics in his paper on automatic programming for model-based vision [29]. In his system, search was controlled by thresholds on probabilistic measures of the reliability and plausibility of matches.

Lowe describes in general terms the application of Bayesian techniques in his book on Visual Recognition [51]. He treats the minimization of expected running time of recognition. In addition he discusses selection among numerous objects.

Object recognition matching systems often use a strategy that can be summarized as a search for the maximal matching that is consistent. Consistency is frequently defined to mean that the matching image feature is within finite bounds of its expected position (bounded error models). Cass' system [14] is one example. Such an approach may be cast in the framework defined here by assuming uniform probability density functions for the feature deviations. Pose solution with this approach is likely to be more complicated than the sequential linear-least-squares method that can be used when feature deviations have normal models. Cass' approach effectively finds the global optimum of its objective function. It performs well on occluded or fragmented real images.

Beveridge, Weiss and Riseman [6] use an objective function for line segment based recognition that is similar to the one described here. In their work, the penalty for deviations is quadratic, while the reward for correspondence is non-linear (exponential) in the amount of missing segment length. (By contrast, the reward described in this paper is, for stationary models, linear in the length of aggregate features.) The trade-off parameters in their objective function were determined empirically. Their

system gives good performance in a domain of real images.

Burns and Riseman [12] and Burns [11] describe a classification based recognition system. They focus on the use of description networks for efficiently searching among multiple objects with a recursive indexing scheme.

Hanson and Fua [27] [26] describe a general objective function approach to image understanding. They use a minimum description length (MDL) criterion that is designed to work with generic object models. The approach presented here is tailored for specific object models.

6.6 Summary

A MAP model matching technique for visual object recognition has been described. The resulting objective function has a simple form when normal feature deviation models and linear projection models are used. Experimental results were shown indicating that MAP Model Matching works well in low resolution matching, where feature deviations are significant. Related work was discussed.

Chapter 7

Posterior Marginal Pose Estimation

In the previous chapter on MAP Model Matching the object recognition problem was posed as an optimization problem resulting from a statistical theory. In that formulation, complete hypotheses consist of a description of the correspondences between image and object features, as well as the pose of the object. The method was shown to provide effective evaluations of match and pose.

The formulation of recognition that is described in this chapter, Posterior Marginal Pose Estimation [1] (PMPE), builds on MAP Model Matching. It provides a smooth objective function for evaluating the pose of the object – without commitment to a particular match. The pose is the most important aspect of the problem, in the sense that knowing the pose enables grasping or other interaction with the object.

In this chapter, the objective function is explored by probing in selected parts of pose space. The domain of these experiments is features derived from synthetic laser radar range imagery, and grayscale video imagery. A limited pose space search is performed in the video experiment.

In Chapter 8 the Expectation – Maximization (EM) algorithm is discussed as a

[1] An early version of this work appeared in [76]

means of searching for local maxima of the objective function in pose space.

Additional experiments in object recognition using the PMPE objective function are described in Chapter 10. There, the EM algorithm is used in conjunction with an indexing method that generates initial hypotheses.

7.1 Objective Function for Pose

The following method was motivated by the observation that in heuristic searches over correspondences with the objective function of MAP Model Matching, hypotheses having implausible matches scored poorly in the objective function. The implication was that summing posterior probability over all the matches (at a specific pose) might provide a good pose evaluator. This has proven to be the case. Although intuitively, this might seem like an odd way to evaluate a pose, it is at least democratic in that all poses are evaluated in the same way. The resulting pose estimator is smooth, and is amenable to local search in pose space. It is not tied to specific matches – it is perhaps in keeping with Marr's recommendation that computational theories of vision should try to satisfy a principle of least commitment [52].

Additional motivation was provided by the work by Yuille, Geiger and Bülthoff on stereo [78]. They discussed computing disparities in a statistical theory of stereo where a marginal is computed over matches.

In MAP Model Matching, joint hypotheses of match and pose were evaluated by their posterior probability, given an image – $p(\Gamma, \beta \mid Y)$. Γ and β stand for correspondences and pose, respectively, and Y for the image features. The posterior probability was built from specific models of features and correspondences, objects, and projection that were described in the previous chapters. The present formulation will first be described using the independent correspondence model. Use of the Markov correspondence model will be described in the following section.

Here we use the same strategy for evaluating object poses: they are evaluated

by their posterior probability, given an image: $p(\beta \mid Y)$. The posterior probability density of the pose may be computed from the joint posterior probability on pose and match, by formally taking the marginal over possible matches:

$$p(\beta \mid Y) = \sum_{\Gamma} p(\Gamma, \beta \mid Y) \ .$$

In Section 6.1, Equation 6.4, $p(\Gamma, \beta \mid Y)$ was obtained via Bayes' rule from probabilistic models of image features, correspondences, and the pose. Substituting for $p(\Gamma, \beta \mid Y)$, the posterior marginal may be written as

$$p(\beta \mid Y) = \sum_{\Gamma} \frac{p(Y \mid \Gamma, \beta) p(\Gamma, \beta)}{p(Y)} \ . \tag{7.1}$$

Using equations 2.1 (the independent feature model) and 6.2, we may express the posterior marginal of β in terms of the component densities:

$$p(\beta \mid Y) = \frac{1}{p(Y)} \sum_{\Gamma_1} \sum_{\Gamma_2} \cdots \sum_{\Gamma_n} \prod_i p(Y_i \mid \Gamma, \beta) \prod_i p(\Gamma_i) p(\beta)$$

or

$$p(\beta \mid Y) = \frac{p(\beta)}{p(Y)} \sum_{\Gamma_1} \sum_{\Gamma_2} \cdots \sum_{\Gamma_n} \prod_i [p(Y_i \mid \Gamma_i, \beta) p(\Gamma_i)] \ .$$

Breaking one factor out of the product gives

$$p(\beta \mid Y) = \frac{p(\beta)}{p(Y)} \sum_{\Gamma_1} \sum_{\Gamma_2} \cdots \sum_{\Gamma_n} \left[\prod_{i=1}^{n-1} [\, p(Y_i \mid \Gamma_i, \beta) p(\Gamma_i)] \right] p(Y_n \mid \Gamma_n, \beta) p(\Gamma_n) \ ,$$

or

$$p(\beta \mid Y) = \frac{p(\beta)}{p(Y)} \sum_{\Gamma_1} \sum_{\Gamma_2} \cdots \sum_{\Gamma_{n-1}} \left[\prod_{i=1}^{n-1} [\, p(Y_i \mid \Gamma_i, \beta) p(\Gamma_i)] \right] \left[\sum_{\Gamma_n} p(Y_n \mid \Gamma_n, \beta) p(\Gamma_n) \right] \ .$$

Continuing in similar fashion yields

$$p(\beta \mid Y) = \frac{p(\beta)}{p(Y)} \prod_i \left[\sum_{\Gamma_i} p(Y_i \mid \Gamma_i, \beta) p(\Gamma_i) \right] .$$

This may be written as

$$p(\beta \mid Y) = \frac{p(\beta)}{p(Y)} \prod_i p(Y_i \mid \beta) , \tag{7.2}$$

since

$$p(Y_i \mid \beta) = \sum_{\Gamma_i} p(Y_i \mid \Gamma_i, \beta) p(\Gamma_i) . \tag{7.3}$$

Splitting the Γ_i sum into its cases gives,

$$p(Y_i \mid \beta) = p(Y_i \mid \Gamma_i = \perp, \beta) p(\Gamma_i = \perp) + \sum_{M_j} p(Y_i \mid \Gamma_i = M_j, \beta) p(\Gamma_i = M_j) .$$

Substituting the densities assumed in the model of Section 6.1 in Equations 6.1 and 2.2 then yields

$$p(Y_i \mid \beta) = \frac{1}{W_1 \cdots W_v} B_i + \sum_{M_j} \mathrm{G}_{\psi_{ij}}(Y_i - \mathcal{P}(M_j, \beta)) \frac{1 - B_i}{m} . \tag{7.4}$$

Installing this into Equation 7.2 leads to

$$p(\beta \mid Y) = \frac{B_1 B_2 \cdots B_n}{(W_1 W_2 \cdots W_v)^n} \frac{p(\beta)}{p(Y)} \prod_i \left[1 + \sum_{M_j} \frac{W_1 \cdots W_v}{m} \frac{1 - B_i}{B_i} \mathrm{G}_{\psi_{ij}}(Y_i - \mathcal{P}(M_j, \beta)) \right]$$

As in Section 6.1 the objective function for Posterior Marginal Pose Estimation is defined as the scaled logarithm of the posterior marginal probability of the pose,

$$L(\beta) \equiv \ln \left[\frac{p(\beta \mid Y)}{C} \right] ,$$

where, as before,

$$C = \frac{B_1 B_2 \cdots B_n}{(W_1 W_2 \cdots W_v)^n} (2\pi)^{\frac{-s}{2}} |\psi_\beta|^{\frac{-1}{2}} \frac{1}{p(Y)} \ .$$

This leads to the following expression for the objective function (use of a normal pose prior is assumed)

$$L(\beta) = -\frac{1}{2}(\beta - \beta_o)^T \psi_\beta^{-1} (\beta - \beta_0) + \sum_i \ln \left[1 + \sum_{M_j} \frac{W_1 \cdots W_v}{m} \frac{1 - B_i}{B_i} \mathrm{G}_{\psi_{ij}}(Y_i - \mathcal{P}(M_j, \beta)) \right] \tag{7.5}$$

.

This objective function for evaluating pose hypotheses is a smooth function of the pose. Methods of continuous optimization may be used to search for local maxima, although starting values are an issue.

The first term in the PMPE objective function (Equation 7.5) is due to the pose prior. It is a quadratic penalty for deviations from the nominal pose. The second term essentially measures the degree of alignment of the object model with the image. It is a sum taken over image features of a smooth non-linear function that peaks up positively when the pose brings object features into alignment with the image feature in question. The logarithmic term will be near zero if there are no model features close to the image feature in question.

In a straightforward implementation of the objective function, the cost of evaluating a pose is $O(mn)$, since it is essentially a non-linear double sum over image and model features.

7.2 Using the Markov Correspondence Model

When the Markov Correspondence model of Section 2.3 is used instead of the independent correspondence model, the summing techniques of the previous section no longer apply. Because of this, a computationally attractive closed form formula

for the posterior probability no longer obtains. Nevertheless, it will be shown that the posterior probability at a pose can still be efficiently evaluated using dynamic programming.

Referring to Equation 7.1, and using the independence of match and pose in the prior (discussed in Section 6.1), the posterior marginal probability of a pose may be written as follows,

$$p(\beta \mid Y) = \sum_{\Gamma} \frac{p(Y \mid \Gamma, \beta) p(\Gamma) p(\beta)}{p(Y)} \ .$$

Using Equations 2.3 and 6.1,

$$p(\beta \mid Y) = \frac{p(\beta)}{p(Y)} \sum_{\Gamma} \ p(Y_1 \mid \Gamma_1, \beta) p(Y_2 \mid \Gamma_2, \beta) \cdots p(Y_n \mid \Gamma_n, \beta) \ q(\Gamma_1) q(\Gamma_2) \cdots q(\Gamma_n)$$
$$r_1(\Gamma_1, \Gamma_2) r_2(\Gamma_2, \Gamma_3) \cdots r_{n-1}(\Gamma_{n-1}, \Gamma_n)$$

This may be re-written as follows,

$$p(\beta \mid Y) = \frac{p(\beta)}{p(Y)} \sum_{\Gamma_1 \Gamma_2 \ldots \Gamma_n} \left[\prod_{i=1}^{n} c_i(\Gamma_i) \prod_{i=1}^{n-1} r_i(\Gamma_i, \Gamma_{i+1}) \right] \ , \tag{7.6}$$

where

$$c_i \equiv p(Y_i \mid \Gamma_i, \beta) q(\Gamma_i) \ .$$

Here, the dependence of c on β has been suppressed for notational brevity.

Next it will be shown that $p(\beta \mid Y)$ may be written using a recurrence relation:

$$p(\beta \mid Y) = \frac{p(\beta)}{p(Y)} \sum_{\Gamma_n} h_{n-1}(\Gamma_n) c_n(\Gamma_n) \ , \tag{7.7}$$

where

$$h_1(a) \equiv \sum_{b} c_1(b) r_1(b, a) \tag{7.8}$$

and

$$h_{n+1}(a) \equiv \sum_{b} h_n(b) c_{n+1}(b) r_{n+1}(b, a) \ . \tag{7.9}$$

Expanding Equation 7.7 in terms of the recurrence relation,

$$p(\beta \mid Y) = \frac{p(\beta)}{p(Y)} \sum_{\Gamma_n} \left[\sum_{\Gamma_{n-1}} h_{n-2}(\Gamma_{n-1}) c_{n-1}(\Gamma_{n-1}) r_{n-1}(\Gamma_{n-1}, \Gamma_n) \right] c_n(\Gamma_n) \ ,$$

or

$$p(\beta \mid Y) = \frac{p(\beta)}{p(Y)} \sum_{\Gamma_{n-1}\Gamma_n} h_{n-2}(\Gamma_{n-1}) \prod_{i=n-1}^{n} c_i(\Gamma_i)\, r_{n-1}(\Gamma_{n-1}, \Gamma_n) \ .$$

Again using the recurrence relation,

$$p(\beta \mid Y) = \frac{p(\beta)}{p(Y)} \sum_{\Gamma_{n-1}\Gamma_n} \left[\sum_{\Gamma_{n-2}} h_{n-3}(\Gamma_{n-2}) c_{n-2}(\Gamma_{n-2}) r_{n-2}(\Gamma_{n-2}, \Gamma_{n-1}) \right]$$

$$\cdot \prod_{i=n-1}^{n} c_i(\Gamma_i)\, r_{n-1}(\Gamma_{n-1}, \Gamma_n) \ ,$$

or

$$p(\beta \mid Y) = \frac{p(\beta)}{p(Y)} \sum_{\Gamma_{n-2}\Gamma_{n-1}\Gamma_n} h_{n-3}(\Gamma_{n-2}) \prod_{i=n-2}^{n} c_i(\Gamma_i) \prod_{i=n-2}^{n-1} r_i(\Gamma_i, \Gamma_{i+1}) \ .$$

Continuing in similar fashion leads to

$$p(\beta \mid Y) = \frac{p(\beta)}{p(Y)} \sum_{\Gamma_2\Gamma_3 \ldots \Gamma_n} h_1(\Gamma_2) \prod_{i=2}^{n} c_i(\Gamma_i) \prod_{i=2}^{n-1} r_i(\Gamma_i, \Gamma_{i+1}) \ ,$$

and now using the base expression for $h_1(\cdot)$,

$$p(\beta \mid Y) = \frac{p(\beta)}{p(Y)} \sum_{\Gamma_2\Gamma_3 \ldots \Gamma_n} \left[\sum_{\Gamma_1} c_1(\Gamma_1) r_1(\Gamma_1, \Gamma_2) \right] \prod_{i=2}^{n} c_i(\Gamma_i) \prod_{i=2}^{n-1} r_i(\Gamma_i, \Gamma_{i+1}) \ ,$$

or finally,

$$p(\beta \mid Y) = \frac{p(\beta)}{p(Y)} \sum_{\Gamma_1\Gamma_2 \ldots \Gamma_n} \left[\prod_{i=1}^{n} c_i(\Gamma_i) \prod_{i=1}^{n-1} r_i(\Gamma_i, \Gamma_{i+1}) \right] \ ,$$

which is the same as Equation 7.6. This completes the verification of Equation 7.7.

Next, a dynamic programming algorithm will be described that efficiently evaluates an objective function that is proportional to the posterior marginal probability

of a pose. The objective function is $\frac{p(Y)}{p(\beta)}p(\beta \mid Y)$. The algorithm is a direct implementation of the recurrence defined in Equations 7.7, 7.8 and 7.9, that builds a table of values of $h_i(\cdot)$ from the bottom up. Note that $h_i(b)$ only has two values, depending on whether $b = \perp$ or not. In the following description, the symbol $\top$ is used to stand for an anonymous model feature. $H_{..}$ denotes array locations that store values of h_i, and $H(\cdot,\cdot,\cdot)$ is an access function, defined below, that accesses the stored values.

;;; Use Dynamic Programming to evaluate PMPE with Markov Correspondence Model.
EVALUATE-POSE(β)

$H_{1\perp} \leftarrow \sum_b C(1,b,\beta)r_1(b,\perp)$

$H_{1\top} \leftarrow \sum_b C(1,b,\beta)r_1(b,\top)$

For $i \leftarrow 2$ To $N-1$

$H_{i\perp} \leftarrow \sum_b H(i-1,b)C(i,b,\beta)r_{n+1}(b,\perp)$

$H_{i\top} \leftarrow \sum_b H(i-1,b)C(i,b,\beta)r_{n+1}(b,\top)$

RETURN $(\sum_b H(N-1,b)C(n,b,\beta))$

;;; Define the auxiliary function C.
$C(i,b,\beta)$

RETURN$(p(Y_i \mid b\beta)q(b))$

;;; Access values of H stored in a table.
H(a,b)

IF $b = \perp$ RETURN $(H_{a\perp})$

ELSE RETURN $(H_{a\top})$

The loop in EVALUATE-POSE executes $O(n)$ times, and each time through the loop does $O(m)$ evaluations of the summands, so the complexity is $O(mn)$. This

has the same complexity as a straightforward implementation of the PMPE objective function when the Markov model is not used (Equation 7.5).

The summing technique used here was described by Cheeseman [17] in a paper about using maximum-entropy methods in expert systems.

7.3 Range Image Experiment

An experiment investigating the utility of Posterior Marginal Pose Estimation is described in this section. Additional experiments are described in Chapter 10.

The objective function of Equation 7.5 was sampled in a domain of synthetic range imagery. The feasibility of coarse-fine search methods was investigated by sampling smoothed variants of the objective function.

7.3.1 Preparation of Features

The preparation of the features used in the experiment is summarized in Figure 7-1. The features were oriented-range features, as described in Section 5.4. Two sets of features were prepared, the "model features", and the "image features".

The object model features were derived from a synthetic range image of an M35 truck that was created using the ray tracing program associated with the BRL CAD Package [23]. The ray tracer was modified to produce range images instead of shaded images. The synthetic range image appears in the upper left of Figure 7-2.

In order to simulate a laser radar, the synthetic range image described above was corrupted with simulated laser radar sensor noise, using a sensor noise model that is described by Shapiro, Reinhold, and Park [62]. In this noise model, measured ranges are either valid or anomalous. Valid measurements are normally distributed, and anomalous measurements are uniformly distributed. The corrupted range image appears in Figure 7-2 on the right. To simulate post sensor processing, the corrupted image was "restored" via a statistical restoration method of Menon and Wells [56].

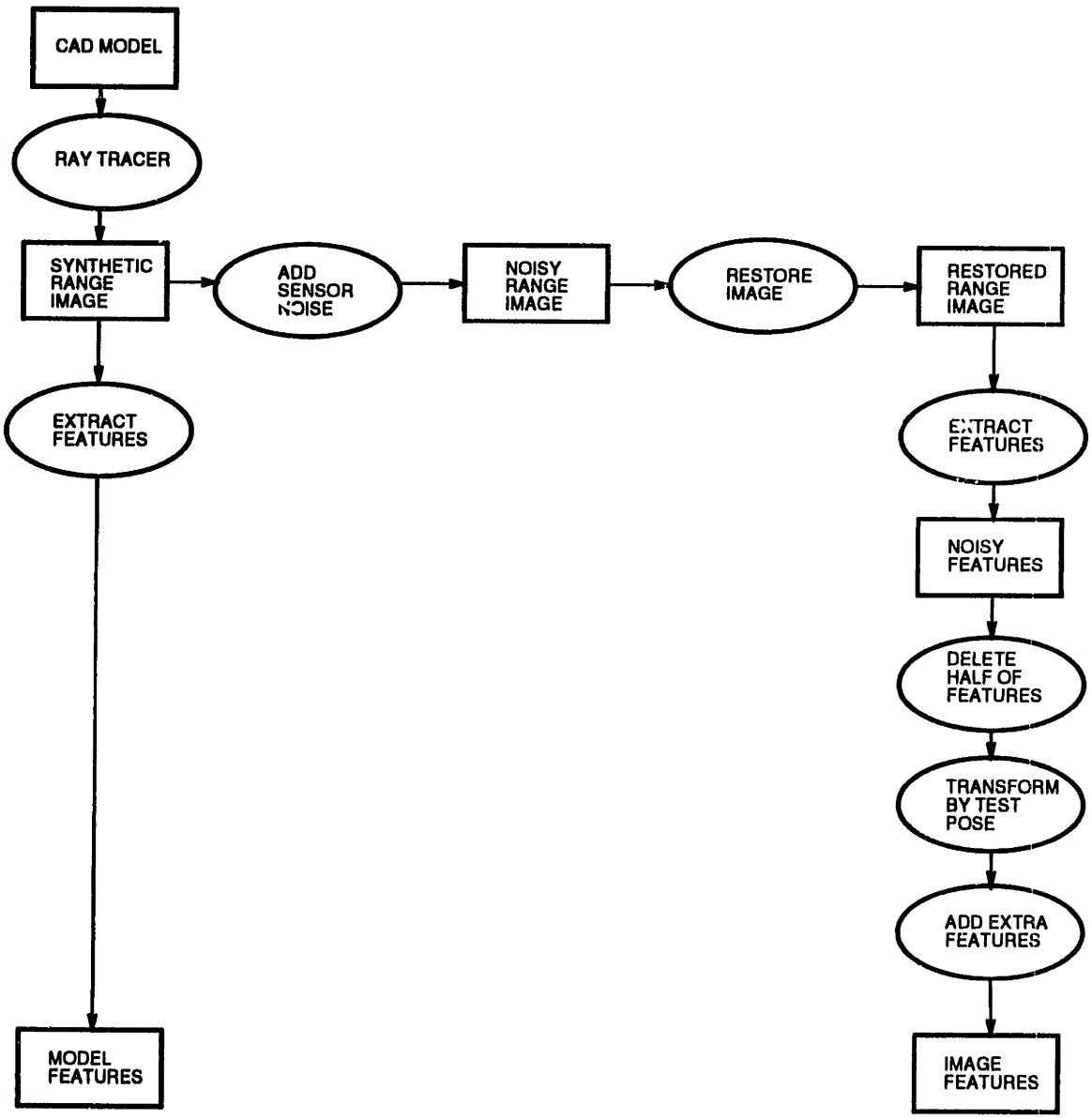


Figure 7-1: Preparation of Features

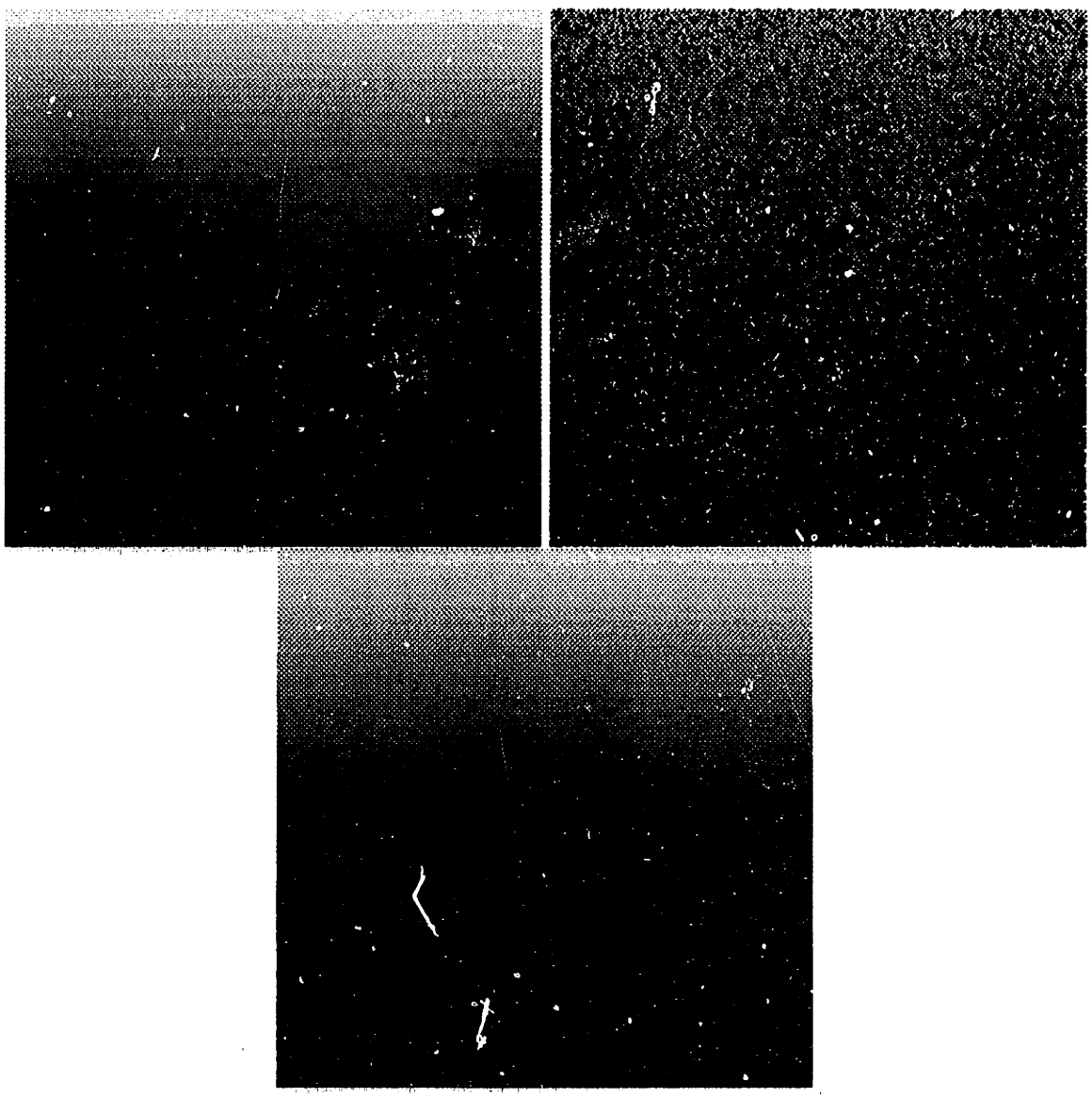


Figure 7-2: Synthetic Range Image, Noisy Range Image, and Restored Range Image

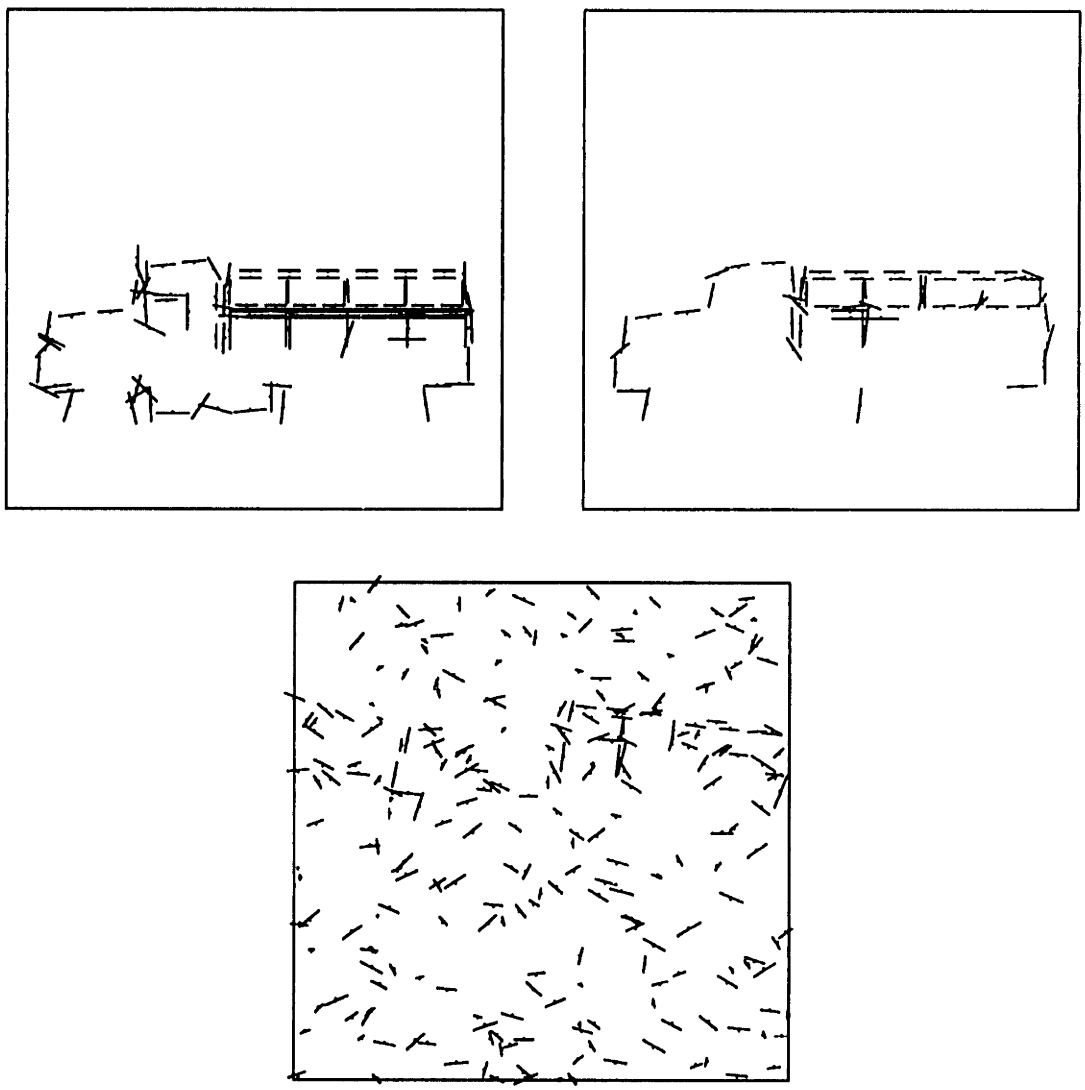


Figure 7-3: Model Features, Noisy Features, and Image Features

The restored range image appears in the lower position of Figure 7-2.

Oriented range features, as described in Section 5.4, were extracted from the synthetic range image, for use as model features – and from the restored range image, these are called the noisy features. The features were extracted from the range images in the following manner. Range discontinuities were located by thresholding neighboring pixels, yielding range discontinuity curves. These curves were then segmented into approximately 20-pixel-long segments via a process of line segment approximation. The line segments (each representing a fragment of a range discontinuity curve) were then converted into oriented range features in the following manner. The X and Y coordinates of the feature were obtained from the mean of the pixel coordinates. The normal vector to the pixels was gotten via least-squares line fitting. The range to the feature was estimated by taking the mean of the pixel ranges on the near side of the discontinuity. This information was packaged into an oriented-range feature, as described in Section 5.4. The model features are shown in the first image of Figure 7-3. Each line segment represents one oriented-range feature, the ticks on the segments indicate the near side of the range discontinuity. There are 113 such object features.

The noisy features, derived from the restored range image, appear in the second image of Figure 7-3. There are 62 noisy features. Some features have been lost due to the corruption and restoration of the range image. The set of image features was prepared from the noisy features by randomly deleting half of the features, transforming the survivors according to a test pose, and adding sufficient randomly generated features so that $\frac{1}{8}$ of the features are due to the object. The 248 image features appear in the third image of Figure 7-3.

7.3.2 Sampling The Objective Function

The objective function of Equation 7.5 was sampled along four straight lines passing through the (known) location in pose space of the test pose. Oriented stationary

statistics were used, as described in Section 3.3. The stationary feature covariance was estimated from a hand match done with a mouse between the model features and the noisy features. The background rate parameter B was set to $\frac{7}{8}$.

Samples taken along a line through the location of the true pose in pose space, parallel to the X axis are shown in Figure 7-4. This corresponds to moving the object along the X axis. The first graph shows samples taken along a 100 pixel length (the image is 256 pixels square). The second graph of Figure 7-4 shows samples taken along a 10 pixel length, and the third graph shows samples taken along a 1 pixel length. The X coordinate of the test pose is 55.5, the third graph shows the peak of the objective function to be in error by about one twentieth pixel.

Samples taken along a line parallel to the μ axis of pose space are shown in Figure 7-5. This corresponds to a simultaneous change in scale and angular orientation of the object.

Each of the above graphs represents 50 equally spaced samples. The samples are joined with straight line segments for clarity. Sampling was also done parallel to the Y and ν axes with similar results.

The sampling described in this section shows that in the experimental domain the objective function has a prominent, sharp peak near the correct location. Some local maxima are also apparent. The observed peak may not be the dominant peak – no global searching was performed.

Coarse-Fine Sampling

Additional sampling of the objective of Equation 7.5 was performed to investigate the feasibility of coarse-fine search techniques. A coarse-fine search method for finding maxima of the pose-space objective function would proceed as follows. Peaks are initially located at a coarse scale. At each stage, the peak from the previous scale is used as the starting value for a search at the next (less smooth) scale.

The objective function was smoothed by replacing the stationary feature covari-

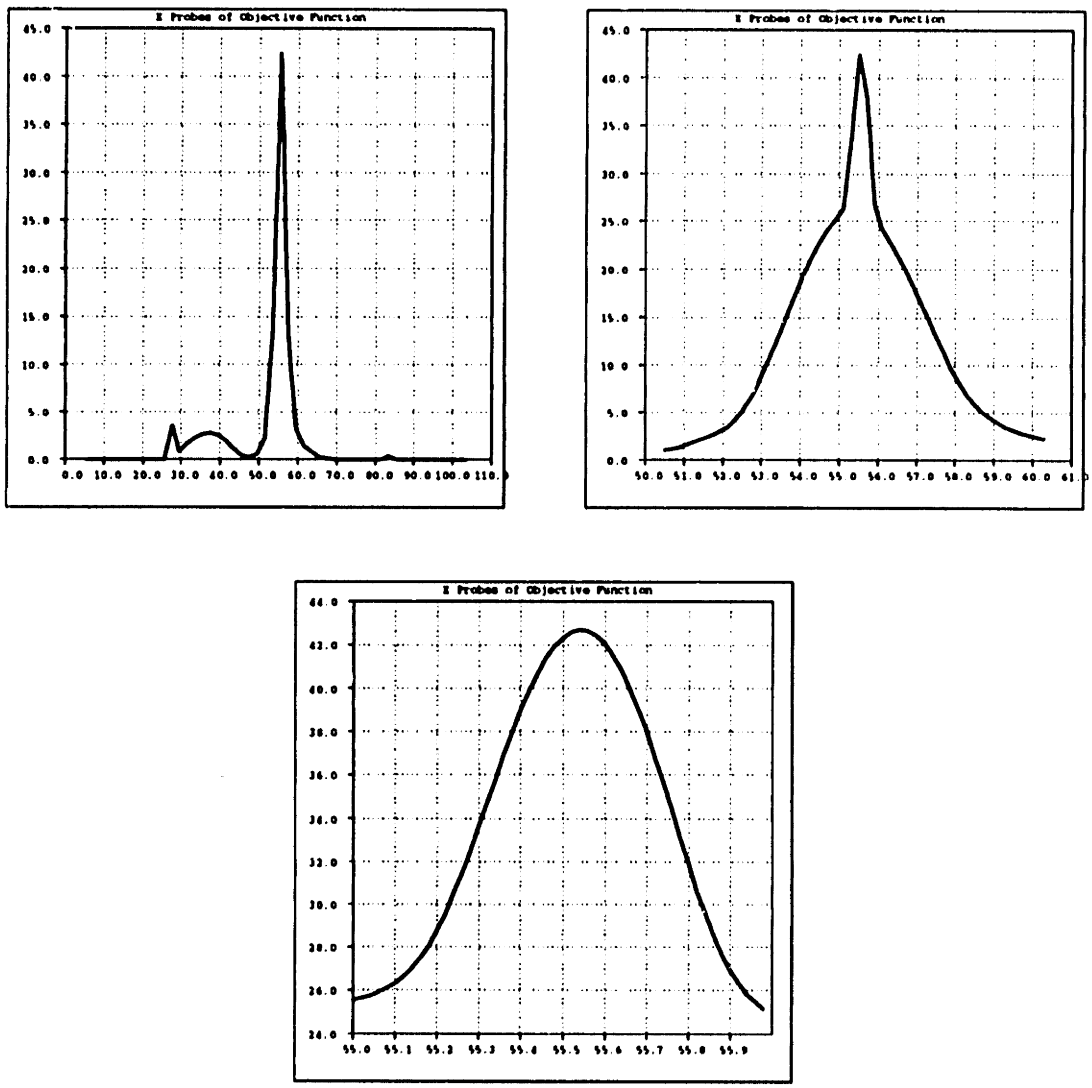


Figure 7-4: Objective Function Samples Along X-Oriented Line Through Test Pose, Lengths: 100 Pixels, 10 Pixels, 1 Pixel

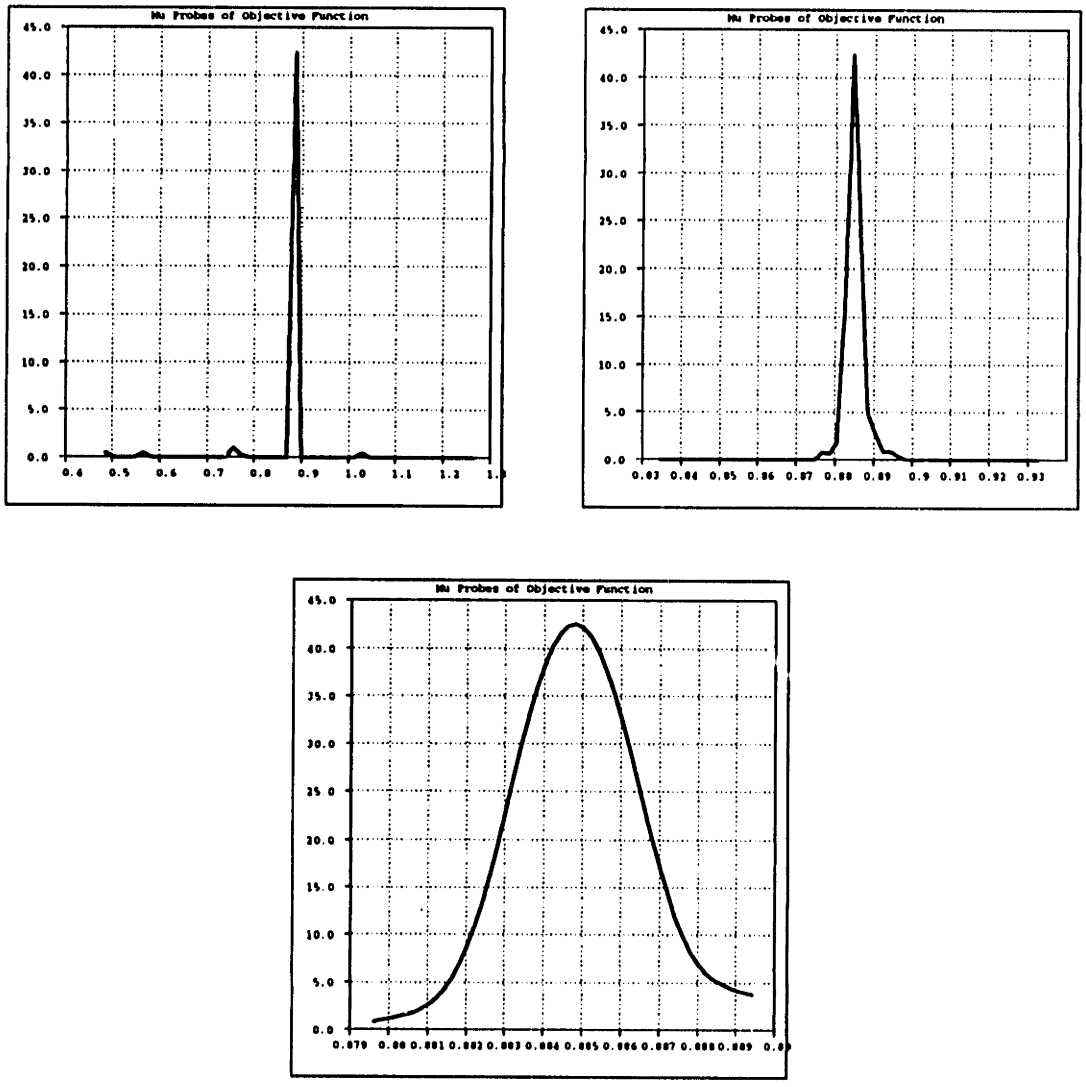


Figure 7-5: Objective Function Samples Along μ-Oriented Line Through Test Pose, Lengths: .8, .1, and .01

ance matrix $\hat{\psi}$ in the following manner:

$$\hat{\psi} \leftarrow \hat{\psi} + \psi_s \ .$$

The effect of the smoothing matrix ψ_s is to increase the spatial scale of the covariance matrices that appear in the objective function.

Probes along the X axis through the known location of the test pose, with various amounts of smoothing are shown in Figure 7-6. The smoothing matrices used in the probing were as follows, in the same order as the figures.

$$\text{DIAG}((.1)^2, (.1)^2, (10.0)^2, (10.0)^2) \ ,$$

$$\text{DIAG}((.025)^2, (.025)^2, (2.5)^2, (2.5)^2) \ ,$$

and

$$\text{DIAG}((.01)^2, (.01)^2, 1.0, 1.0) \ .$$

where DIAG(·) constructs diagonal matrices from its arguments. These smoothing matrices were determined empirically. (No smoothing was performed in the fourth figure.)

These smoothed sampling experiments indicate that coarse-fine search may be feasible in this domain. In Figure 7-6 it is apparent that the peak at one scale may be used as a starting value for local search in the next scale. This indicates that a final line search along the X axis could use the coarse fine strategy. It is not sufficient evidence that such a strategy will work in general. As before, there is no guarantee that the located maximum is the global maximum.

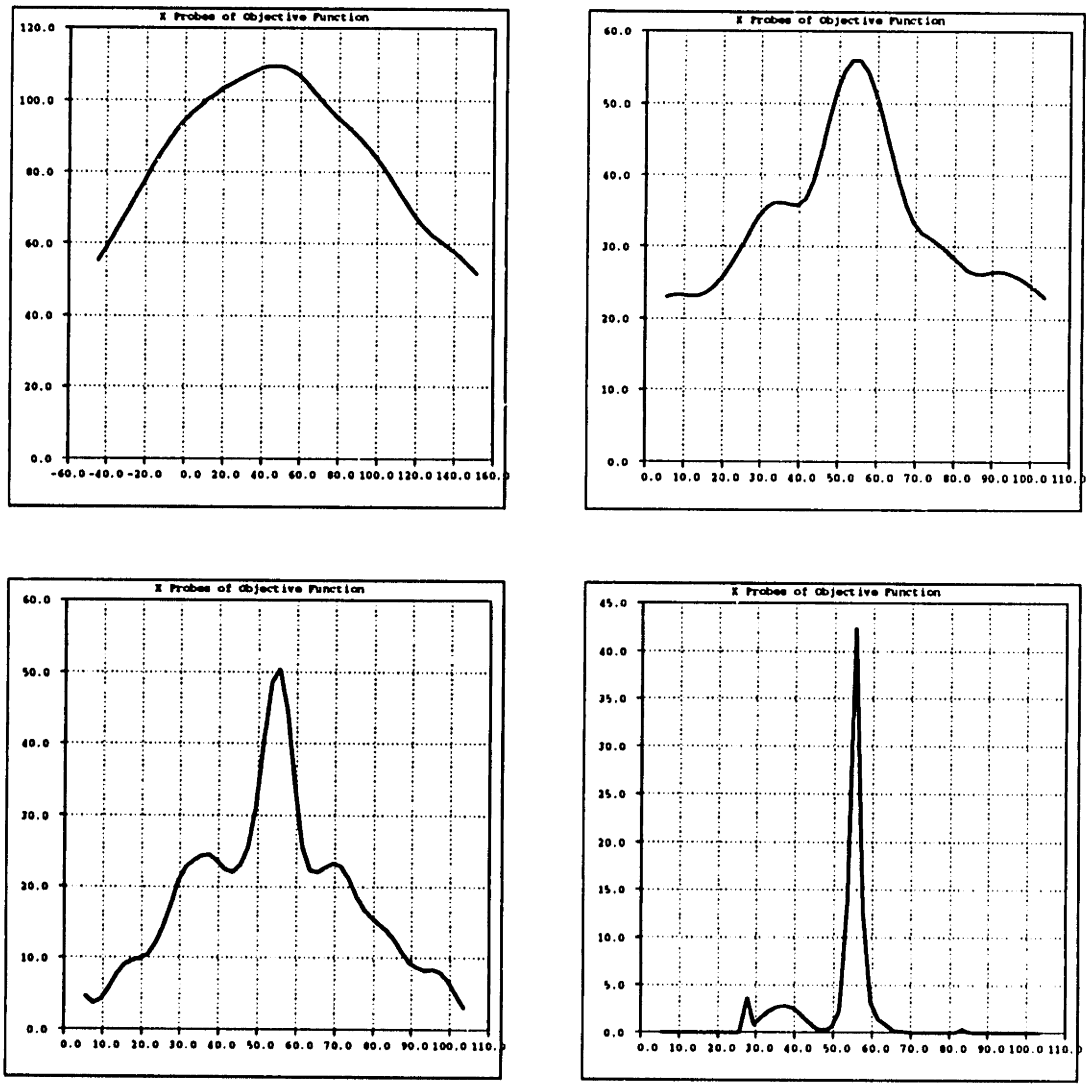


Figure 7-6: X Probes in Smoothed Objective Function

7.4 Video Image Experiment

In this section, another experiment with the PMPE objective function is described. The features are point-radius features derived from video images. A local search in pose space is carried out, and the objective function, and a smoothed variant, are probed in the vicinity of the peak.

7.4.1 Preparation of Features

The features used in this experiment are the same as those used in the MAP Model Matching correspondence search experiment reported in Section 6.2. They are point-radius features, as described in Section 5.3. The features appear in Figure 6-4.

7.4.2 Search in Pose Space

A search was carried out in pose space from a starting value that was determined by hand. The search was implemented with Powell's method [59] of multidimensional non-linear optimization. Powell's method is similar to the conjugate-gradient method, but derivatives are not used. The line minimizations were carried out with Brent's method [59], which uses successive parabolic approximations. The pose resulting from the search is illustrated in Figure 7-7. This result is close to the best result from the MAP Model Matching correspondence search experiment. That result is reproduced here in Figure 7-8. It is comforting that these two substantially different search methods (combinatorial versus continuous) provide similar answers in, at least, one experiment.

7.4.3 Sampling The Objective Function

Samples were taken along four straight lines passing through the peak in the objective function resulting from the search in pose space reported above. (In the range experiment, sampling was done through the known true pose.) The results are illus-

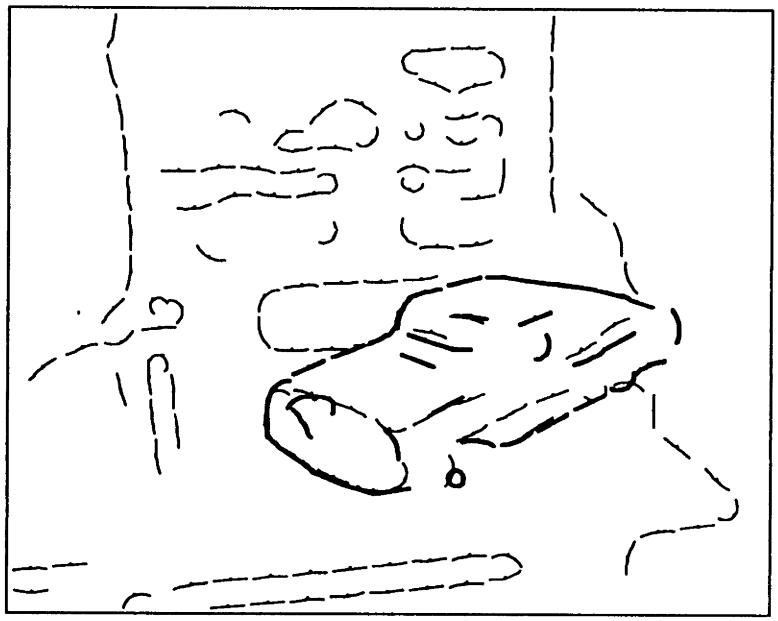


Figure 7-7: Results of Search in Pose Space

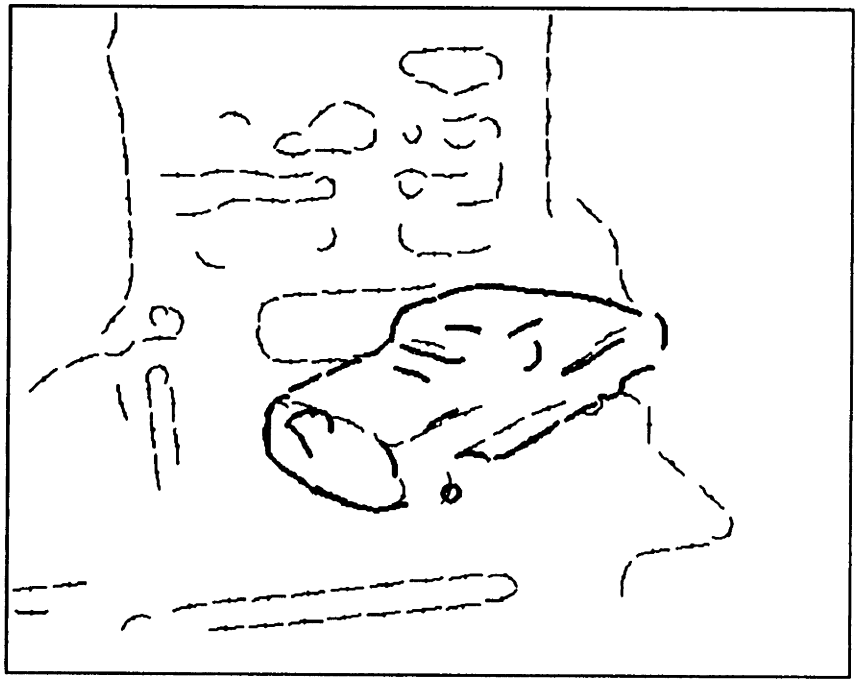


Figure 7-8: Best Results from MAP Model Matching Correspondence Search

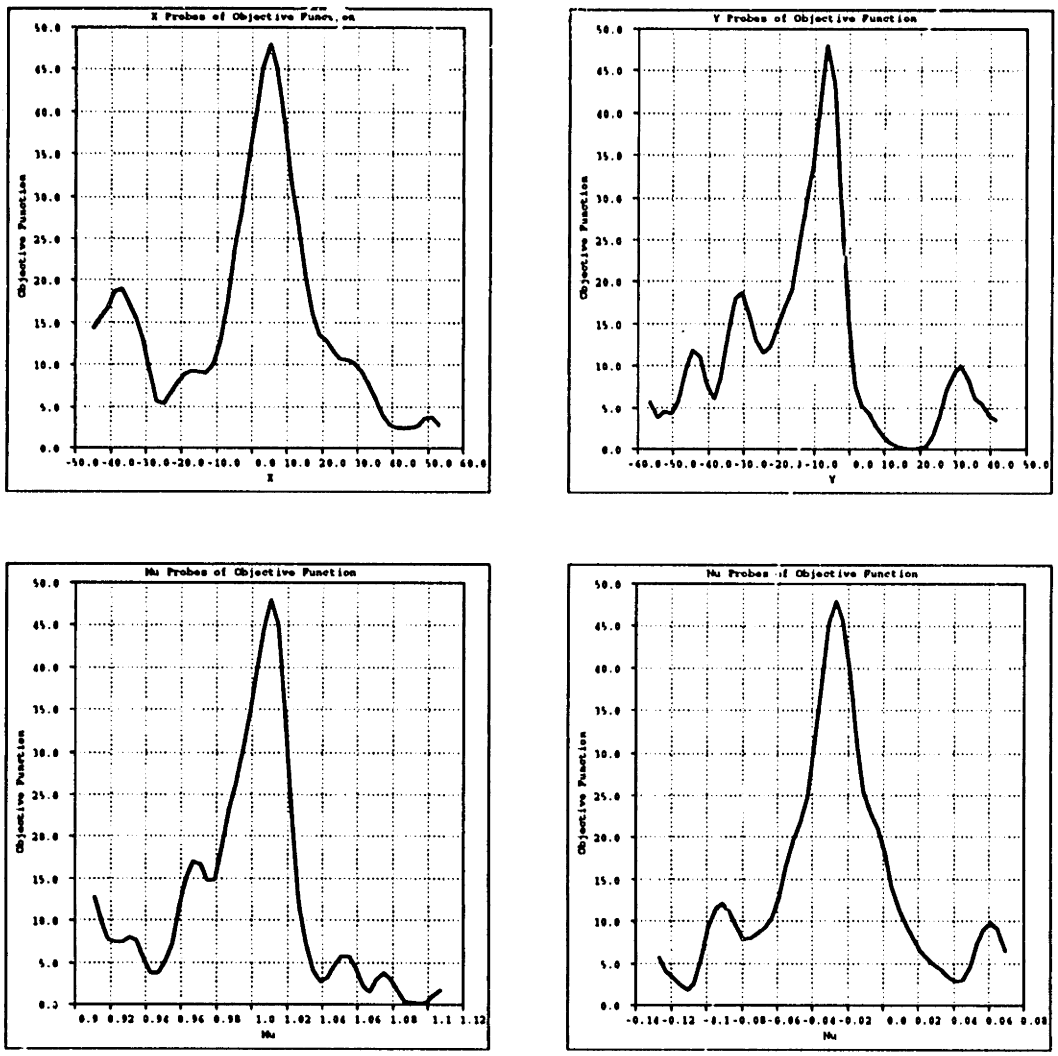


Figure 7-9: Probes of Objective Function Peak

trated in Figure 7-9. The peak in this data is not as sharp as the peak in the range experiment reported in the previous section. This is likely due to the fact that the features used in the video experiment are substantially less constraining that those used in the range experiment – which have good range information in them.

Sampling of the objective function with smoothing was also performed, as in Section 7.3.2.

Smoothing was performed at one scale. The smoothing matrix was

$$\text{DIAG}((.03)^2, (.03)^2, (3.0)^2, (3.0)^2) \ .$$

Probing, performed in the same manner as in Figure 7-9 was performed on the smoothed objective function. The results are shown in Figure 7-10. In comparison to the range image experiment, local maxima are more of an issue here. This may be partly due to the background features here having more structure than the randomly generated background features used in the range image experiment. Because of this, anomalous pose estimates (where the pose corresponding to the global maximum of the objective function is seriously in error) may be more likely in this domain than in the range experiment.

7.5 Relation to Robust Estimation

This section describes a relationship between PMPE and robust estimation. By simplifying the domain a robust estimator of position is obtained. A connection between the simplified robust estimator and neural networks is discussed.

Consider the following simplifications of the domain:

- drop the pose prior
- the object has one feature
- the image is one-dimensional with width W
- the pose is a scalar
- the projection function translates: $\mathcal{P}(\cdot, \beta) = \beta$

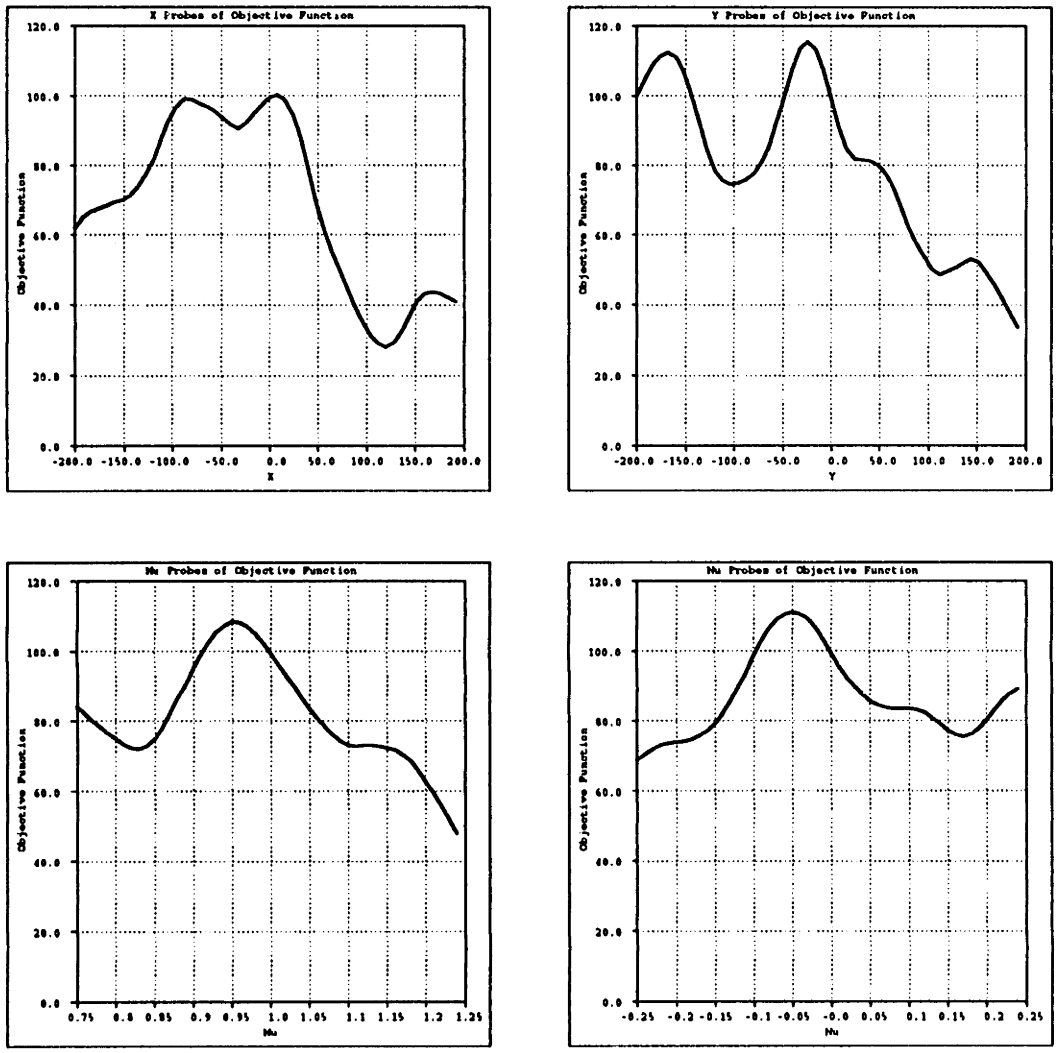


Figure 7-10: Probes of Smoothed Objective Function

With these simplifications, the observation model of Equation 6.1 becomes

$$p(Y_i \mid \Gamma, \beta) = \begin{cases} \frac{1}{W} & \text{if } \Gamma_i = \perp \\ \mathrm{G}_\sigma(Y_i - \beta) & \text{otherwise} \end{cases} ,$$

where

$$\mathrm{G}_\sigma(x) = \frac{1}{\sqrt{2\pi}\sigma} \exp\left(-\frac{x^2}{2\sigma^2}\right) .$$

In this simplified domain Γ may be interpreted as a collection of variables that describe the validity of their corresponding measurements in Y. Thus, $\Gamma_i \neq \perp$ may be interpreted as meaning that Y_i is valid, and $\Gamma_i = \perp$ as Y_i being invalid. $p(Y_i)$ is defined to be zero outside of the range $[\frac{-W}{2}, \frac{W}{2}]$.

The prior on correspondences of Equation 2.2 takes the following form

$$p(\Gamma_i) = \begin{cases} B & \text{if } \Gamma_i = \perp \\ 1 - B & \text{otherwise} \end{cases} .$$

Using Bayes' rule and the independence of Γ_i and β allows the following probability of a sample and its validity,

$$p(Y_i, \Gamma_i \mid \beta) = p(Y_i \mid \Gamma_i, \beta) p(\Gamma_i) = \begin{cases} \frac{B}{W} & \text{if } \Gamma_i = \perp \\ (1 - B)\mathrm{G}_\sigma(Y_i - \beta) & \text{otherwise} \end{cases} . \qquad (7.10)$$

The probability of a sample may now be expressed by taking a marginal over the probability in Equation 7.10, as follows,

$$p(Y_i \mid \beta) = \sum_{\Gamma_i} p(Y_i, \Gamma_i \mid \beta) = \frac{B}{W} + (1 - B)\mathrm{G}_\sigma(Y_i - \beta) .$$

Defining an objective function as a log likelihood of β

$$L(\beta) = \ln\left[\prod_i p(Y_i \mid \beta)\right] ,$$

leads to the analog of the PMPE objective function for this simplified domain,

$$L(\beta) = \sum_i \ln \left[\frac{B}{W} + (1 - B)\mathrm{G}_\sigma(Y_i - \beta) \right] \quad . \tag{7.11}$$

This may also be written

$$L(\beta) = \sum_i S(Y_i - \beta) \tag{7.12}$$

where

$$S(x) = \ln \left[\frac{B}{W} + (1 - B)G_\sigma(x) \right] \quad ,$$

This is the Maximum Likelihood objective function for estimating the mean of a normal population of variance σ^2, that is contaminated with a uniform population of width W, where the fraction of the mixture due to the uniform population is B.

The function $S(x)$ is approximately quadratic when the residual is small, and approaches a constant when the residual is large. When B goes to zero, $S(x)$ becomes quadratic, and the estimator becomes least squares, for the case of a pure normal population. When $-S(x)$ is viewed as a penalty function, it is seen to provide a quadratic penalty for small residuals, as least squares does, but the penalty saturates when residuals become large. Robust estimation is concerned with estimators that are, like this one, less sensitive to outliers that least squares. As with many robust estimators, the resulting optimization problem is more difficult than least squares, since the objective function is non-convex. This estimator falls into the class of re-descending M-estimators as discussed by Huber [41].

PMPE is somewhat different from robust estimation in that the saturating aspect of the objective function not only decreases the influence of "outliers" (by analogy, the background features), it also reduces the influence of image features that don't correspond to (are not close to) a given object feature.

7.5.1 Connection to Neural Network Sigmoid Function

There is an important connection between the estimator of Equation 7.12 and the sigmoid function of neural networks,

$$\sigma(x) = \frac{1}{1 + \exp(-x)} \ .$$

The sigmoid function is a smooth variant of a logical switching function that has been used for modeling neurons. It has been used extensively by the neural network community in the construction of networks that classify and exhibit some forms of learning behavior. The NETtalk neural network of Sejnowski and Rosenberg [61] is a well know example.

It turns out that, under some conditions on the parameters, the sigmoid function of x^2 is approximately equal to $S(x)$, ignoring shifting and scaling. This near equality is illustrated in Figure 7-11.

The two functions that are plotted in the figure are

$$f(x) = 2.0[\sigma(x^2) - .5] \qquad \text{and} \qquad g(x) = \frac{\ln[.25 + .75\exp(-x^2)]}{\ln[.25]} \ .$$

The upper graph shows $f(x) and g(x)$ plotted together, while the lower graph shows their difference. It can be seen that they agree to better than one percent.

Because of this near equality, for a special case of the parameters, a network that evaluates the ML estimator of Equation 7.12 for a contaminated normal population will have the form illustrated in Figure 7-12.

This network, with its arrangement of sigmoid and sum units seems to fit the definition of a neural network.

The robust estimator of Equation 7.12, and its neural network approximation, are (approximately) optimal for locating a Gaussian cluster in uniform noise.

A similar neural network realization of the PMPE objective function would likewise be near optimal for locating an object against a uniform background.

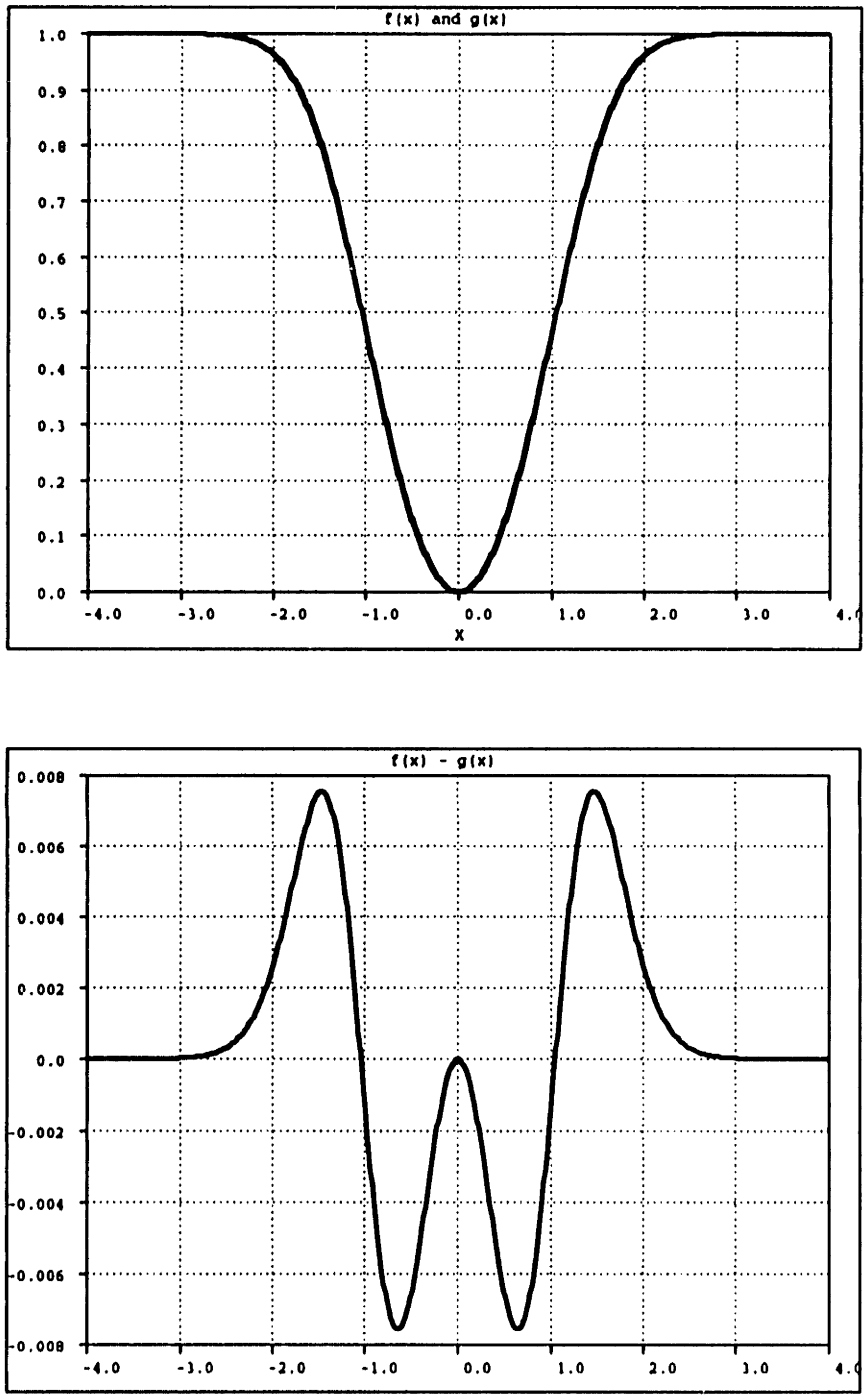


Figure 7-11: $f(x)$ and $g(x)$, and $f(x) - g(x)$

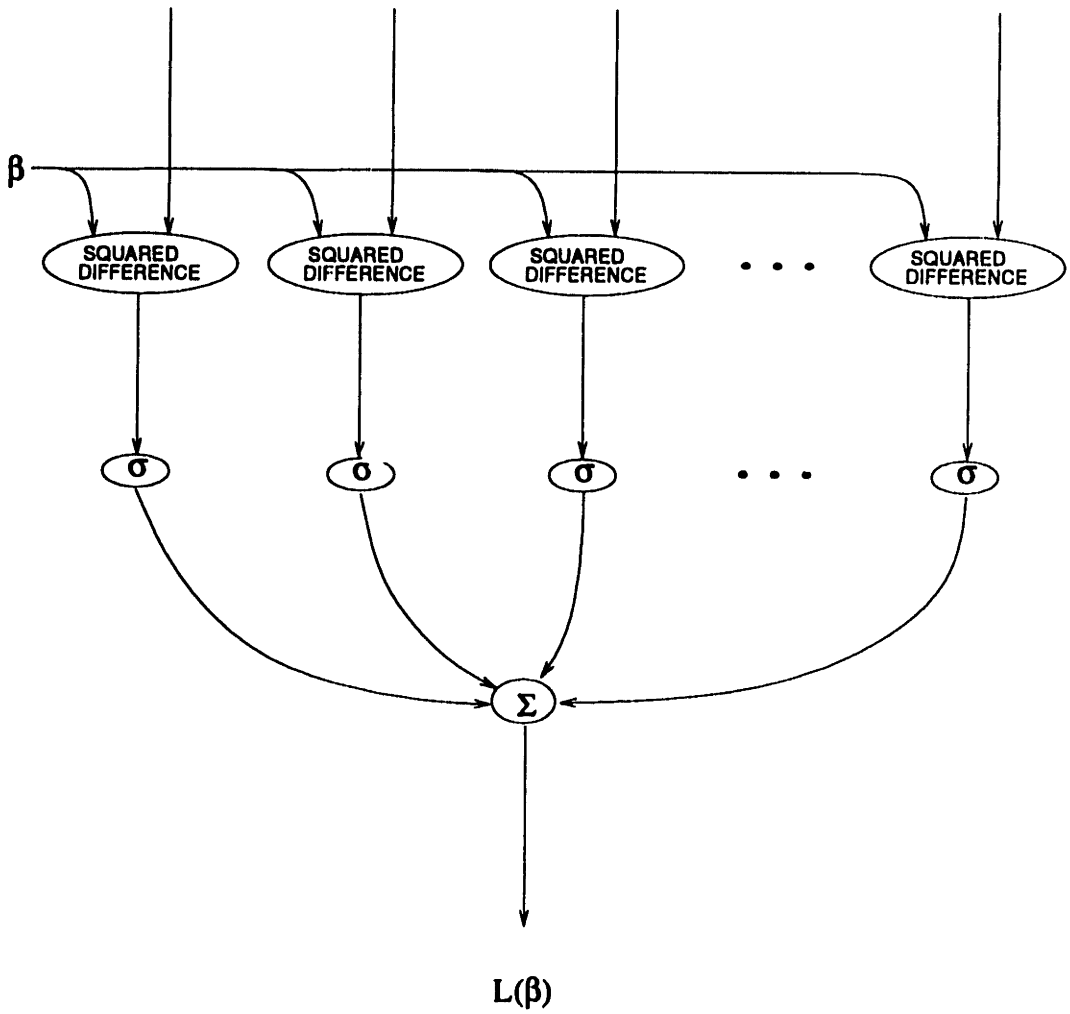


Figure 7-12: Network Implementation of MAP Estimator for Contaminated Normal

7.6 PMPE Efficiency Bound

This section provides a lower bound on the covariance matrix of the PMPE estimator. Estimators of vector parameters (like pose) may be characterized by the covariance matrix of the estimates they produce. The Cramer-Rao bound provides a lower limit for the covariance matrix of unbiased estimators. Unbiased estimators that achieve this bound are call *efficient* estimators. Discussions of estimator efficiency and Cramer-Rao bounds appear in [63] and [72].

The Cramer-Rao bound on the covariance matrix of estimators of β based on observations of X is given by the inverse of the Fisher information matrix,

$$\mathrm{COV}(\hat{\beta}) \geq I_F^{-1}(\beta) \ .$$

Here, $\mathrm{COV}(\cdot)$ denotes the covariance matrix of the random vector argument. This matrix inequality means that $\mathrm{COV}(\hat{\beta}) - I_F^{-1}(\beta)$ is positive semi-definite.

The Fisher information matrix is defined as follows,

$$I_F(\beta) \equiv \mathrm{E}_X([\nabla_\beta \ln p(X \mid \beta)][\nabla_\beta \ln p(X \mid \beta)]^T)$$

where ∇_β is the gradient with respect to β, which yields a column-vector, and $\mathrm{E}_X(\cdot)$ stands for the expected value of the argument with respect to $p(X)$.

The covariance matrix, and the Cramer-Rao bound, of the PMPE estimator are difficult to calculate. Instead, the Cramer-Rao bound and efficiency will be determined for estimators that have access to both observed features Y_i, *and* the correspondences Γ_i. The Cramer-Rao bound for these "complete-data" estimators will be found, and it will be shown that there are no efficient complete-data estimators. Because of this, the PMPE estimator is subject to the same bound as the complete-data estimators, and the PMPE estimator cannot be efficient. This follows, because the PMPE estimator can be considered to be technically a complete-data estimator that

ignores the correspondence data.

In terms of the complete-data estimator, the Fisher information has the following form,

$$I_F(\beta) \equiv \mathrm{E}_{Y,\Gamma}([\nabla_\beta \ln p(Y,\Gamma \mid \beta)][\nabla_\beta \ln p(Y,\Gamma \mid \beta)]^T) \ . \tag{7.13}$$

Assuming independence of feature coordinates and of correspondences, the probability of the complete-data is

$$p(Y,\Gamma \mid \beta) = \prod_i p(Y_i,\Gamma_i \mid \beta) \ .$$

Using Bayes rule and the independence of Γ and β,

$$p(Y_i,\Gamma_i \mid \beta) = p(Y_i \mid \Gamma_i,\beta)p(\Gamma_i) \ . \tag{7.14}$$

Referring to Equations 6.1 and 6.3, and using constant background probability B, and linear projection, the complete-data component probability may be written as follows,

$$p(Y_i,\Gamma_i \mid \beta) = \begin{cases} \frac{B}{W_1 W_2 \ldots W_v} & \text{if } \Gamma_i = \perp \\ \frac{1-B}{m} \mathrm{G}_{\psi_{ij}}(Y_i - M_j\beta) & \text{if } \Gamma_i = M_j \ . \end{cases}$$

Working towards an expression for the Fisher information, we differentiate the complete-data probability to obtain

$$\nabla_\beta \ln p(Y,\Gamma \mid \beta) = \nabla_\beta \ln \prod_i p(Y_i,\Gamma_i \mid \beta) = \sum_i \frac{\nabla_\beta p(Y_i,\Gamma_i \mid \beta)}{p(Y_i,\Gamma_i \mid \beta)} \ .$$

When $\Gamma_i = \perp$, $\nabla_\beta p(Y_i,\Gamma_i \mid \beta) = 0$, otherwise, in the case $\Gamma_i = M_j$,

$$\nabla_\beta p(Y_i,\Gamma_i \mid \beta) = \nabla_\beta \frac{1-B}{m} \mathrm{G}_{\psi_{ij}}(Y_i - M_j\beta) \ .$$

Differentiating the normal density (a formula for this appears in 8.3), gives

$$\nabla_\beta p(Y_i, \Gamma_i \mid \beta) = (-)\frac{1-B}{m} \mathrm{G}_{\psi_{ij}}(Y_i - M_j\beta)\, M_j^T \psi_{ij}^{-1}(Y_i - M_j\beta) \ ,$$

so that

$$\frac{\nabla_\beta p(Y_i, \Gamma_i \mid \beta)}{p(Y_i, \Gamma_i \mid \beta)} = -M_j^T \psi_{ij}^{-1}(Y_i - M_j\beta) \qquad \text{when} \qquad \Gamma_i = M_j \ .$$

Then the gradient of the complete-data probability may be expressed as

$$\nabla_\beta \ln p(Y, \Gamma \mid \beta) = - \sum_{ij:\Gamma_i = M_j} M_j^T \psi_{ij}^{-1}(Y_i - M_j\beta) \ .$$

Note that setting this expression to zero defines the Maximum Likelihood estimator for β in the complete-data case, as follows:

$$\sum_{ij:\Gamma_i = M_j} M_j^T \psi_{ij}^{-1} Y_i = \sum_{ij:\Gamma_i = M_j} M_j^T \psi_{ij}^{-1} M_j \hat{\beta} \ ,$$

or

$$\hat{\beta} = \left(\sum_{ij:\Gamma_i = M_j} M_j^T \psi_{ij}^{-1} M_j \right)^{-1} \sum_{ij:\Gamma_i = M_j} M_j^T \psi_{ij}^{-1} Y_i \ . \tag{7.15}$$

This estimator is linear in Y. The inverse has been assumed to exist – it will exist, provided certain linear independence conditions are met, and enough correspondences to model features appear in the match. This typically requires two to four correspondences in the applications described here.

Returning to the Fisher information, we need to evaluate the expectation:

$$I_F = \mathrm{E}_{Y,\Gamma} \left(\left[\sum_{ij:\Gamma_i = M_j} M_j^T \psi_{ij}^{-1} \epsilon_{ij} \right] \left[\sum_{ij:\Gamma_i = M_j} M_j^T \psi_{ij}^{-1} \epsilon_{ij} \right]^T \right) \ ,$$

where the ij'th residual has been written as follows,

$$\epsilon_{ij} \equiv Y_i - M_j\beta \ .$$

Re-naming indices and pulling out the sums gives

$$I_F = \mathrm{E}_{Y,\Gamma}\left(\sum_{ij:\Gamma_i=M_j} \sum_{i'j':\Gamma_{i'}=M_{j'}} M_j^T\psi_{ij}^{-1}\epsilon_{ij}\epsilon_{i'j'}^T\psi_{i'j'}^{-1}M_{j'}\right) \ .$$

Referring to Equation 7.14, the expectation may be split and moved as follows,

$$I_F = \mathrm{E}_{\Gamma}\left(\sum_{ij:\Gamma_i=M_j} \sum_{i'j':\Gamma_{i'}=M_{j'}} M_j^T\psi_{ij}^{-1}\mathrm{E}_{Y|\Gamma}(\epsilon_{ij}\epsilon_{i'j'}^T)\psi_{i'j'}^{-1}M_{j'}\right) \ .$$

The inner expectation is over mutually independent Gaussian random vectors, and equals their covariance matrix when the indices agree, and is zero otherwise, so

$$I_F = \mathrm{E}_{\Gamma}\left(\sum_{ij:\Gamma_i=M_j} \sum_{i'j':\Gamma_{i'}=M_{j'}} M_j^T\psi_{ij}^{-1}\psi_{ij}\delta_{ii'}\delta_{jj'}\psi_{i'j'}^{-1}M_{j'}\right) \ .$$

This expression simplifies to the following:

$$I_F = \mathrm{E}_{\Gamma}\left(\sum_{ij:\Gamma_i=M_j} M_j^T\psi_{ij}^{-1}M_j\right) \ .$$

The sum may be re-written in the following way by using a delta function comparing Γ_i and M_j,

$$I_F = \sum_{ij}\mathrm{E}_{\Gamma}\left(\delta_{\Gamma_i M_j}\right)M_j^T\psi_{ij}^{-1}M_j = \sum_{ij}\mathrm{E}_{\Gamma_i}\left(\delta_{\Gamma_i M_j}\right)M_j^T\psi_{ij}^{-1}M_j \ .$$

The expectation is just the probability that an image feature is matched to some model feature. This is $\frac{1-B}{m}$, so the Fisher information may be written in the following

simple form,

$$I_F = \sum_{ij} \frac{1-B}{m} M_j^T \psi_{ij}^{-1} M_j \ ,$$

or as,

$$I_F = (1-B)n \ \frac{1}{mn} \sum_{ij} M_j^T \psi_{ij}^{-1} M_j \ .$$

This is an attractive result, and may be easily interpreted, in relation to the Fisher information for the pose when the correspondences are fixed (a standard linear estimator). The Fisher information in that case is $\sum_{ij} M_j^T \psi_{ij}^{-1} M_j$,it may be interpreted as the sum over matches of the per-match Fisher information.

In light of this, the complete-data Fisher information is seen to be the average of the per-match Fisher information, multiplied by the expected number of features matched to the model, $(1-B)n$.

An efficient unbiased estimator for the complete-data exists if and only if

$$\hat{\beta} = \beta + I_F^{-1}(\beta) \nabla_\beta \ln p(Y, \Gamma \mid \beta) \ .$$

This requires that the right hand side be independent of β, since the estimator $\hat{\beta}$ (Equation 7.15) is not a function of β. Expanding the right hand side,

$$\beta + \left[(1-B)n \ \frac{1}{mn} \sum_{ij} M_j^T \psi_{ij}^{-1} M_j \right]^{-1} \sum_{ij:\Gamma_i = M_j} M_j^T \psi_{ij}^{-1} (Y_i - M_j \beta) \ .$$

This is not independent of β. One way to see this is to note that the factor multiplying β in the second term is a function of Γ. Thus, no efficient estimator exists in the complete-data case, and consequently, no efficient estimator exists for PMPE.

7.7 Related Work

Green [31] and Green and Shapiro [32] describe a theory of Maximum Likelihood laser radar range profiling. The research focuses on statistically optimal detectors

and recognizers. The single pixel statistics are described by a mixture of uniform and normal components. Range profiling is implemented using the EM algorithm. Under some circumstances, least squares provides an adequate starting value. A continuation-style variant is described, where a range accuracy parameter is varied between EM convergences from a coarse value to its true value. Green [31] computes Cramer-Rao bounds for the complete-data case of Maximum Likelihood range profile estimator, and compares simulated and real-data performance to the limits.

Cass [16] [15] describes an approach to visual object recognition that searches in pose space for maximal alignments under the bounded-error model. The pose-space objective function used there is piecewise constant, and is thus not amenable to continuous search methods. The search is based on geometric formulation of the constraints on feasible transformations.

There are some connections between PMPE and standard methods of robust pose estimation, like those described by Haralick [38], and Kumar and Hanson [48]. Both can provide robust estimates of the pose of an object, based on an observed image. The main difference is that the standard methods require specification of the feature correspondences, while PMPE does not – by considering all possible correspondences. PMPE requires a starting value for the pose (as do standard methods of robust pose estimation that use non-convex objective functions).

As mentioned above, Yuille, Geiger and Bülthoff [78] discussed computing disparities in a statistical theory of stereo where a marginal is computed over matches. Yuille extends this technique [79] to other domains of vision and neural networks, among them winner-take-all networks, stereo, long-range motion, the traveling salesman problem, deformable template matching, learning, content addressable memories, and models of brain development. In addition to computing marginals over discrete fields, the Gibbs probability distribution is used. This facilitates continuation-style optimization methods by variation of the temperature parameter. There are some similarities between this approach and using coarse-fine with the PMPE objec-

tive function.

Edelman and Poggio [24] describe a method of 3D recognition that uses a trained Generalized Radial Basis Function network. Their method requires correspondences to be known during training and recognition. One similarity between their scheme and PMPE is that both are essentially arrangements of smooth unimodal functions.

There is a similarity between Posterior Marginal Pose Estimation and Hough transform (HT) methods. Roughly speaking, HT methods evaluate parameters by accumulating votes in a discrete parameter space, based on observed features. (See the survey paper by Illingworth and Kittler [44] for a discussion of Hough methods.)

In a recognition application, as described here, the HT method would evaluate a discrete pose by counting the number of feature pairings that are exactly consistent somewhere within the cell of pose space. As stated, the HT method has difficulties with noisy features. This is usually addressed by counting feature pairings that are exactly consistent somewhere nearby the cell in pose space.

The utility of the HT as a stand-alone method for recognition in the presence of noise is a topic of some controversy. This is discussed by Grimson in [34], pp. 220. Perhaps this is due to an unsuitable noise model implicit in the Hough Transform.

Posterior Marginal Pose Estimation evaluates a pose by accumulating the logarithm of posterior marginal probability of the pose over image features.

The connection between HT methods and parameter evaluation via the logarithm of posterior probability has been described by Stephens [67]. Stephens proposes to call the posterior probability of parameters given image observations "The Probabilistic Hough Transform". He provided an example of estimating line parameters from image point features whose probability densities were described as having uniform and normal components. He also states that the method has been used to track 3D objects, referring to his thesis [68] for definition of the method used.

7.8 Summary

A method of evaluating poses in object recognition, Posterior Marginal Pose Estimation, has been described. The resulting objective function was seen to have a simple form when normal feature deviation models and linear projection models are used.

Limited experimental results were shown indicating that in a domain of synthetic range discontinuity features, the objective function may have a prominent sharp peak near the correct pose. Some local maxima were also apparent. Another experiment, in which the features were derived from video images, was described. Connections to robust estimation and neural networks were examined. Bounds on the performance of simplified PMPE estimators were indicated, and relation to other work was discussed.

Chapter 8

Expectation – Maximization Algorithm

The Expectation – Maximization (EM) algorithm was introduced in its general form by Dempster, Rubin and Laird in 1978 [21]. It is often useful for computing estimates in domains having two sample spaces, where the events in one are unions over events in the other. This situation holds among the sample spaces of Posterior Marginal Pose Estimation (PMPE) and MAP Model Matching. In the original paper, the wide generality of the EM algorithm is discussed, along with several previous appearances in special cases, and convergence results are described.

In this chapter, a specific form of the EM algorithm is described for use with PMPE. It is used for hypothesis refinement in the recognition experiments that are described in Chapter 10. Issues of convergence and implementation are discussed.

8.1 Definition of EM Iteration

In this section a variant of the EM algorithm is presented for use with Posterior Marginal Pose Estimation, which was described in Chapter 7. The following modeling assumptions were used. Normal models are used for matched image features, while

uniform models are used for unmatched (background) features. If a prior on the pose is available, it is normal. The independent correspondence model is used. Additionally, a linear model is used for feature projection.

In PMPE, the pose of an object, β, is estimated by maximizing its posterior probability, given an image.

$$\hat{\beta} = \arg\max_{\beta} p(\beta \mid Y) \ .$$

A necessary condition for the maximum is that the gradient of the posterior probability with respect to the pose be zero, or equivalently, that the gradient of the logarithm of the posterior probability be zero:

$$\mathbf{0} = \nabla_{\beta} \ln p(\hat{\beta} \mid Y) \ . \tag{8.1}$$

In Section 7.1, Equation 7.2 the following formula was given for the posterior probability of the pose of an object, given an image. This assumes use of the independent correspondence model.

$$p(\beta \mid Y) = \frac{p(\beta)}{p(Y)} \prod_i p(Y_i \mid \beta) \ .$$

Imposing the condition of Equation 8.1 yields the following,

$$\mathbf{0} = \nabla_{\beta} \left[\ln \frac{1}{p(Y)} + \ln p(\hat{\beta}) + \sum_i \ln p(Y_i \mid \hat{\beta}) \right]$$

or

$$\mathbf{0} = \frac{\nabla_{\beta} p(\hat{\beta})}{p(\hat{\beta})} + \sum_i \frac{\nabla_{\beta} p(Y_i \mid \hat{\beta})}{p(Y_i \mid \hat{\beta})} \ . \tag{8.2}$$

As in Equation 7.3, we may write the feature PDF conditioned on pose in the following way,

$$p(Y_i \mid \beta) = \sum_{\Gamma_i} p(Y_i \mid \Gamma_i \beta) p(\Gamma_i) \ ,$$

or, using the specific models assumed in Section 7.1, as reflected in Equation 7.4, and

using a linear projection model,

$$p(Y_i \mid \beta) = \frac{B_i}{W_1 W_2 \cdots W_v} + \frac{1 - B_i}{m} \sum_j \mathrm{G}_{\psi_{ij}}(Y_i - M_j\beta) \ .$$

The zero gradient condition of Equation 8.2 may now be expressed as follows,

$$\mathbf{0} = \frac{\nabla_\beta p(\hat{\beta})}{p(\hat{\beta})} + \sum_i \frac{\frac{1-B_i}{m} \sum_j \nabla_\beta \mathrm{G}_{\psi_{ij}}(Y_i - M_j\hat{\beta})}{\frac{B_i}{W_1 W_2 \cdots W_v} + \frac{1-B_i}{m} \sum_j \mathrm{G}_{\psi_{ij}}(Y_i - M_j\hat{\beta})} \ .$$

With a normal pose prior,

$$p(\beta) = \mathrm{G}_{\psi_\beta}(\beta - \beta_0) \quad , \quad \text{and} \quad \nabla_\beta p(\beta) = -p(\beta)\psi_\beta^{-1}(\beta - \beta_0) \ .$$

The gradient of the other normal density is

$$\nabla_\beta \mathrm{G}_{\psi_{ij}}(Y_i - M_j\beta) = -\mathrm{G}_{\psi_{ij}}(Y_i - M_j\beta) M_j^T \psi_{ij}^{-1}(Y_i - M_j\beta) \ . \tag{8.3}$$

Returning to the gradient condition, and using these expressions (negated),

$$\mathbf{0} = \psi_\beta^{-1}(\hat{\beta} - \beta_0) + \sum_i \frac{\frac{1-B_i}{m} \sum_j \mathrm{G}_{\psi_{ij}}(Y_i - M_j\hat{\beta}) M_j^T \psi_{ij}^{-1}(Y_i - M_j\hat{\beta})}{\frac{B_i}{W_1 W_2 \cdots W_v} + \frac{1-B_i}{m} \sum_j \mathrm{G}_{\psi_{ij}}(Y_i - M_j\hat{\beta})} \ .$$

Finally, the zero gradient condition may be expressed compactly as follows,

$$\mathbf{0} = \psi_\beta^{-1}(\hat{\beta} - \beta_0) + \sum_{ij} W_{ij} M_j^T \psi_{ij}^{-1}(Y_i - M_j\hat{\beta}) \ , \tag{8.4}$$

with the following definition:

$$W_{ij} = \frac{\mathrm{G}_{\psi_{ij}}(Y_i - M_j\hat{\beta})}{\frac{B_i}{1-B_i} \frac{m}{W_1 W_2 \cdots W_v} + \sum_j \mathrm{G}_{\psi_{ij}}(Y_i - M_j\hat{\beta})} \ . \tag{8.5}$$

Equation 8.4 has the appearance of being a linear equation for the pose estimate $\hat{\beta}$ that satisfies the zero gradient condition for being a maximum. Unfortunately, it isn't

a linear equation, because W_{ij} (the "weights") are not constants, they are functions of $\hat{\beta}$. To find solutions to Equation 8.4, the EM algorithm iterates the following two steps:

- Treating the weights, W_{ij} as constants, solve Equation 8.4 as a linear equation for a new pose estimate $\hat{\beta}$. This is referred to as the M step.
- Using the most recent pose estimate $\hat{\beta}$, re-evaluate the weights, W_{ij}, according to Equation 8.5. This is referred to as the E step.

The M step is so named because, in the exposition of the algorithm in [21], it corresponds to a Maximum Likelihood estimate. As discussed there, the algorithm is also amenable to use in MAP formulations (like this one). Here the M step corresponds to a MAP estimate of the pose, given that the current estimate of the weights is correct.

The E step is so named because calculating the W_{ij} corresponds to taking the expectation of some random variables, given the image data, and that the most recent pose estimate is correct. These random variables have value 1 if the i'th image feature corresponds to the j'th object feature, and 0 otherwise. Thus, after the iteration converges, the weights provide continuous-valued estimates of the correspondences, that vary between 0 and 1.

It seems somewhat ironic that, having abandoned the correspondences as being part of the hypothesis in the formulation of PMPE, a good estimate of them has re-appeared as a byproduct of a method for search in pose space. This estimate, the posterior expectation, is the minimum variance estimator.

Being essentially a local method of non-linear optimization, the EM algorithm needs good starting values in order to converge to the right local maximum. It may be started on either step. If it is started on the E step, an initial pose estimate is required. When started on the M step, an initial set of weights is needed.

An initial set of weights can be obtained from a partial hypothesis of correspon-

dences in a simple manner. The weights associated with each set of corresponding features in the hypothesis are set to 1, the rest to 0. Indexing methods are one source of such hypotheses. In Chapter 10, Angle Pair Indexing is used to generate candidate hypotheses. In this scenario, indexing provides initial alignments, these are refined using the EM algorithm, then they are verified by examining the value of the peak of the PMPE objective function that the refinement step found.

8.2 Convergence

In the original reference [21], the EM algorithm was shown to have good convergence properties under fairly general circumstances. It is shown that the likelihood sequence produced by the algorithm is monotonic, i.e., the algorithm never reduces the value of the objective function (or in this case, the posterior probability) from one step to the next. Wu [77] claims that the convergence proof in the original EM reference is flawed, and provides another proof, as well as a thorough discussion. It is possible that it will wander along a ridge, or become stuck in a saddle point.

In the recognition experiments reported in Chapter 10 the algorithm typically converges in 10 – 40 iterations.

8.3 Implementation Issues

Some thresholding methods were used speed up the computation of the E and M steps.

The weights W_{ij} provide a measure of feature correspondence. As the algorithm operates, most of the weights have values close to zero, since most pairs of image and object feature don't align well for a given pose. In the computation of the M step, most terms were left out of the sum, based on a threshold for W_{ij}. Some representative weights from an experiment are displayed in Table 10.1 in Chapter 10.

In the E step, most of the work is in evaluating the Gaussian functions, which have

quadratic forms in them. For the reason stated above, most of these expressions have values very close to zero. The evaluation of these expressions was made conditional on a threshold test applied to the residuals $Y_i - M_j\beta$. When the (x,y) part of the residual exceeded a certain length, zero was substituted for the value of the Gaussian expression. Tables indexed by image coordinates might provide another effective way of implementing the thresholding here.

The value of the PMPE objective function is computed as a byproduct of the E step for little additional cost.

8.4 Related Work

The work of Green [31] and Green and Shapiro [32] that is discussed in Section 7.7 describes use of the EM algorithm in a theory of laser radar range profiling.

Lipson [50] describes a non-statistical method for refining alignments that iterates solving linear systems. It matches model features to the nearest image feature under the current pose hypothesis, while the method described here entertains matches to all of the image features, weighted by their probability. Lipson's method was shown to be effective and robust in an implementation that refines alignments under Linear Combination of Views.

Chapter 9

Angle Pair Indexing

9.1 Description of Method

Angle Pair Indexing is a simple method that is designed to reduce the amount of search needed in finding matches for image features in 2D recognition. It uses features having location and orientation.

An invariant property of feature pairs is used to index a table that is constructed ahead of time. The property used is the pair of angles between the feature orientations and the line joining the feature's locations. These angles are θ_1 and θ_2 in Figure 9-1. The pair of angles is clearly invariant under translation, rotation, and scaling in the plane.

Using orientations as well as point locations provides more constraint than point features. Because of this, indexing may be performed on pairs of simple features, rather than groups of three or more.

The table is constructed from the object features in a pre-processing step. It is indexed by the angle pair, and stores the pairs of object features that are consistent with the value of the angles, within the resolution of the table. The algorithm for constructing the table appears below.

A distance threshold is used to suppress entries for features that are very close.

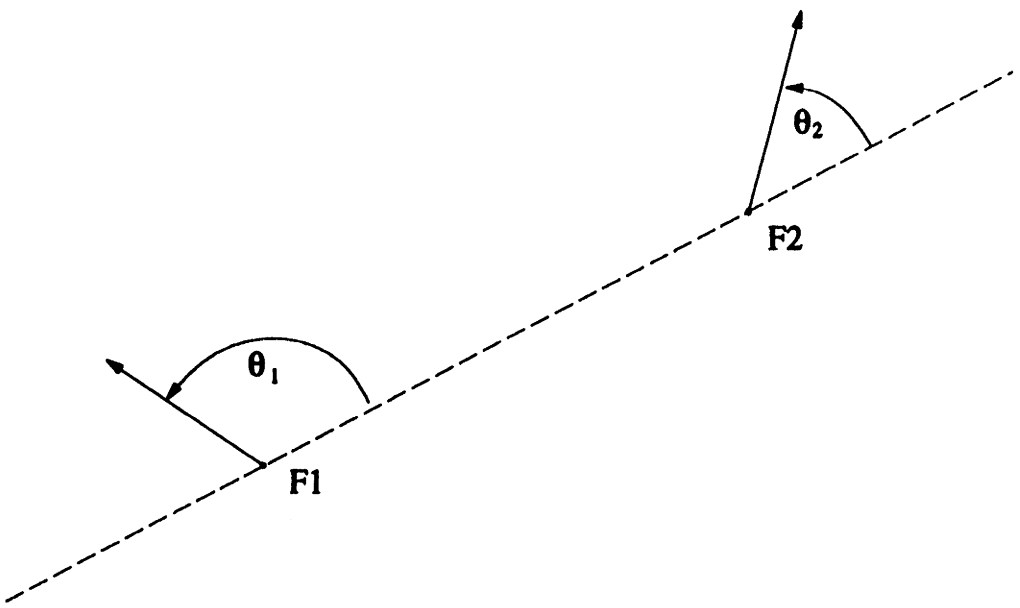


Figure 9-1: Angles for Indexing

Such features pairs yield sloppy initial pose estimates and are poor initial hypotheses for recognition.

```
;;; Given an array model-features and a table size, n
;;; fills in the 2 index array ANGLE-PAIR-TABLE by side-effect.
BUILD-ANGLE-TABLE(model-features, n, distance-threshold)
    m ← LENGTH(model-features)
    ;; First clear the table.
    For i ← 0 To m
        For j ← 0 To m
            ANGLE-PAIR-TABLE[i, j] ← ∅
    ;; Now fill in the table entries.
    For i ← 0 To m
        For j ← 0 To m
            If i ≠ j
                f1 ← model-features[i]
                f2 ← model-features[j]
```

```
If DISTANCE(f1, f2) > distance-threshold
    < q r > ← CALCULATE-INDICES(f1, f2, n)
    ANGLE-PAIR-TABLE[q, r] ← ANGLE-PAIR-TABLE[q, r] ∪ < i j >
```

The following function is used to calculate the table indices for a pair of features. Note that the indexing wraps around when the angles are increased by π. This was done because the features used in the recognition experiments described in this research are often straight edge segments, and their orientations are ambiguous by π.

```
;;; Calculate indices into ANGLE-PAIR-TABLE for a pair of features.
CALCULATE-INDICES(f1, f2, n)
    δθ ← π/n
    i ← (⌊θ1/δθ⌋ mod n)
    j ← (⌊θ2/δθ⌋ mod n)
    return(< i j >)
```

The following algorithm is used at recognition-time to generate a set of pairs of correspondences from image features to object features that have consistent values of the angle pair invariant. The indexing operation saves the expense of searching for pairs of object model features that are consistent with pairs of image features. Table entries from adjacent cells are included among the candidates to accommodate angle values that are "on the edge" of a cell boundary.

```
;;; Map over the pairs of features in an image and generate
;;; candidate pairs of feature correspondences
GENERATE-CANDIDATES(image-features, n)
    candidates ← ∅
    m ← LENGTH(image-features)
```

```
For i ← 0 To m
    For j ← i + 1 to m
        < q r > ← CALCULATE-INDICES (image-features[i], image-features[j], n)
        For δq ← -1 to 1
            For δr ← -1 to 1
                For < k l > ∈ ANGLE-PAIR-TABLE[((q + δq) mod n), ((r + δr) mod n)]
                    candidates ← candidates ∪ < < i k > < j l > >
Return(candidates)
```

9.2 Sparsification

In the recognition experiments described below and in Section 10.1, an additional technique was used to speed up recognition-time processing, and reduce the size of the table. As the table was built, a substantial fraction of the entries were left out of the table. These entries were selected at random. This strategy is based on the following observation: For the purpose of recognizing the object, it is only necessary for some feature pair from the object to be both in the table and visible in the image. If a reasonable fraction of the object is visible, a substantial number of feature pairs will be available as potential partners in a candidate correspondence pair. It is unlikely that the corresponding pairs of object model features will *all* have been randomly eliminated when the table was built, even for fairly large amounts of sparsification.

9.3 Related Work

Indexing based on invariant properties of sets of image features has been used by Lamdan and Wolfson, in their work on geometric hashing [49], and by Clemens and Jacobs [19][20], Jacobs [45], and Thompson and Mundy [70]. In those cases the

invariance is with respect to affine transformations that have eight parameters. In this work the invariance is with respect to translation, rotation, and scale in 2D, where there are four parameters. Thompson and Mundy describe an invariant called *vertex pairs*. These are based on angles relating to pairs of vertices of 3D polyhedra, and their projections into 2D. Angle Pair Indexing is somewhat similar, but is simpler – being designed for 2D from 2D recognition.

Clemens and Jacobs [19] [20], and Jacobs [45] use grouping mechanisms to select small sets of image features that are likely to belong to the same object in the scene.

Chapter 10

Recognition Experiments

This chapter describes several recognition experiments that use Posterior Marginal Pose Estimation with the EM Algorithm. The first is a complete 2D recognition system that uses Angle Pair Indexing as the first stage. In another experiment, the PMPE objective function is evaluated on numerous random alignments. Additionally, the effect of occlusions on PMPE are investigated. Finally, refinement of 3D alignments is demonstrated.

In the following experiments, image edge curves were arbitrarily subdivided into fragments for feature extraction. The recognition experiments based on these features show good performance, but the performance might be improved if a more stable subdivision technique were used.

10.1 2D Recognition Experiments

The experiments described in this section use the EM algorithm to carry out local searches in pose space of the PMPE objective function. This is used for evaluating and refining alignments that are generated by Angle Pair Indexing. A coarse – fine approach is used in refining the alignments produced by Angle Pair Indexing. To this end, two sets of features are used, coarse features and fine features.

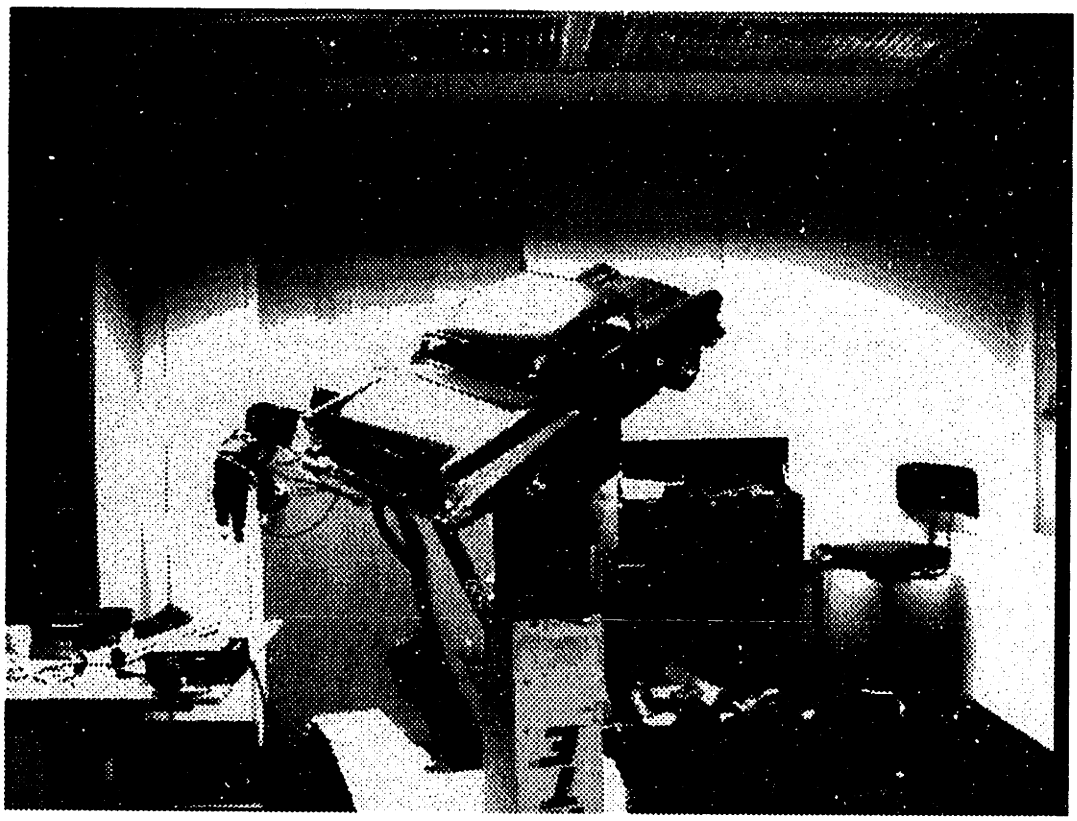


Figure 10-1: Grayscale Image

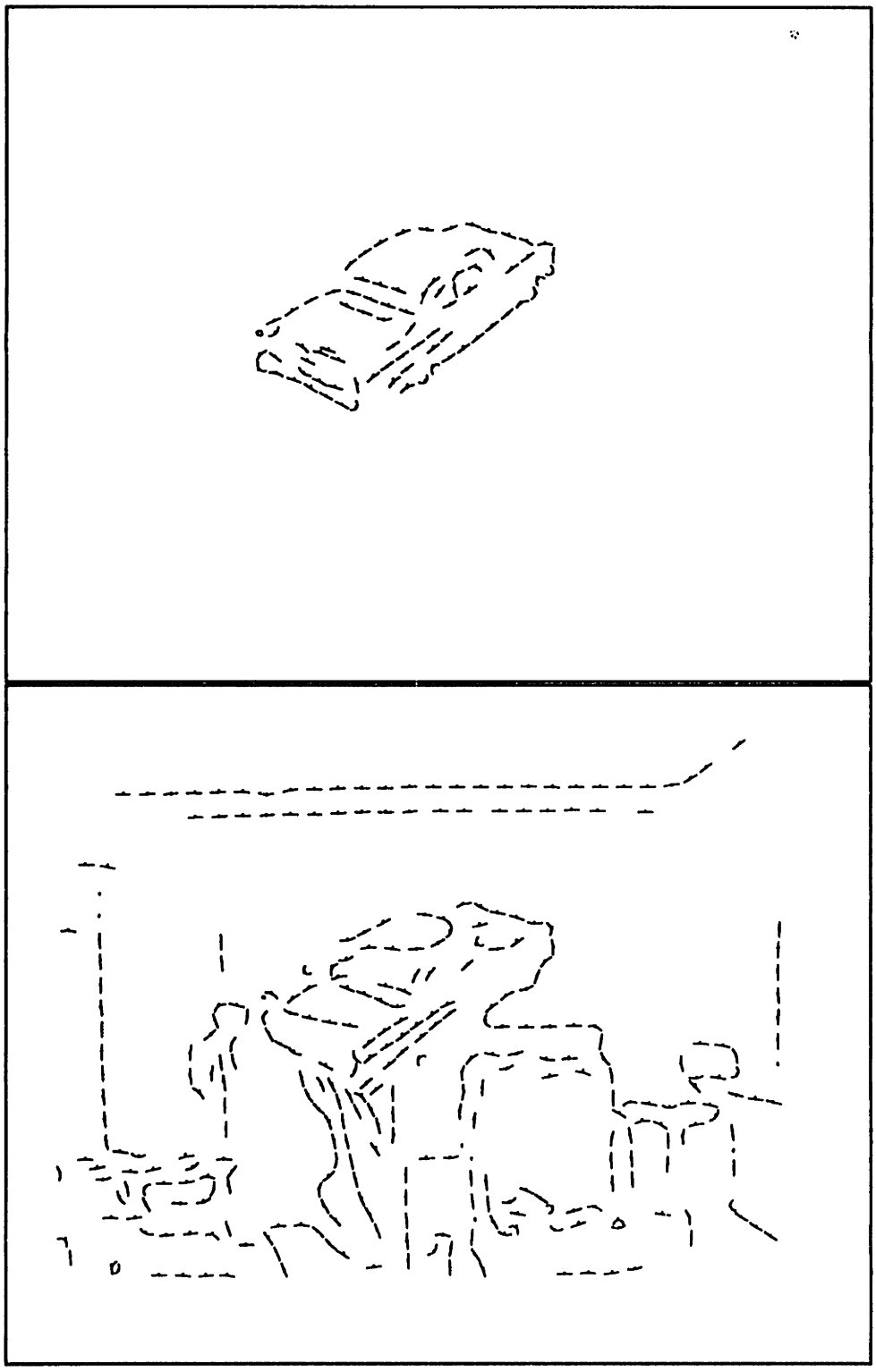


Figure 10-2: Coarse Model and Image Features

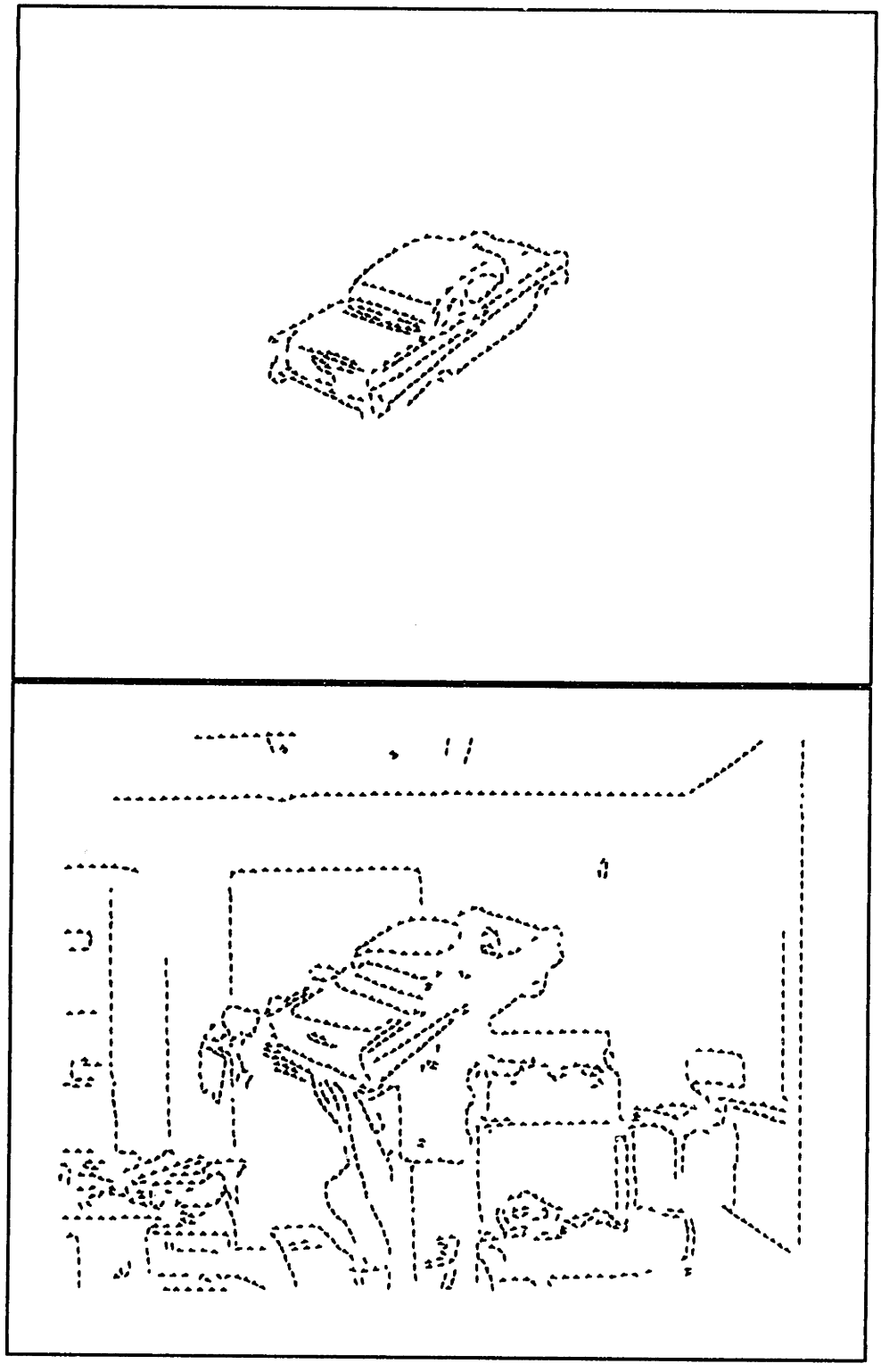


Figure 10-3: Fine Model and Image Features

The video image used for the recognition experiment appears in Figure 10-1. The model features were derived from Mean Edge Images, as described in Section 4.4. The standard deviation of the smoothing that was used in preparing the model and image edge maps was 3.97 for the coarse features, and 1.93 for the fine features. The edge curves were broken arbitrarily every 20 pixels for the coarse features, and every 10 pixels for the fine features. Point-radius features were fitted to the edge curve fragments, as described in Section 5.3. The coarse model and image features appear in Figure 10-2, the fine model and image features appear in Figure 10-3. There are 81 coarse model features, 334 coarse image features, 246 fine model features, and 1063 fine image features.

The oriented stationary statistics model of feature fluctuations was used (this is described in Section 3.3). The parameters (statistics) that appear in the PMPE objective function, the background probability and the covariance matrix for the oriented stationary statistics, were derived from matches that were done by hand. These training matches were also used in the empirical study of the goodness of the normal model for feature fluctuations discussed in Section 3.2.1, and they are described there.

10.1.1 Generating Alignments

Initial alignments were generated using Angle Pair Indexing (described in Chapter 9) on the coarse features. The angle pair table was constructed with 80 by 80 cells, and sparsification was used – 5 percent of the entries were randomly kept. The distance threshold was set at 50 pixels (the image size is 640 by 480). The resulting table contained 234 entries. With these values, uniformly generated random angle pairs have .0365 probability of "hitting" in the table.

When the image feature pairs were indexed into the table, 20574 candidate feature correspondence pairs were generated. This is considerably fewer that the 732 million possible pairs of correspondences in this situation. Figure 10-4 illustrates three of

the candidate alignments by superimposing the object in the images at the pose associated with the initial alignment implied by the pairs of feature correspondences. The indicated scores are the negative of the PMPE objective function computed with the coarse features.

10.1.2 Scoring Indexer Alignments

The initial alignments were evaluated in the following way. The indexing process produces hypotheses consisting of a pair of correspondences from image features to object features. These pairs of correspondences were converted into an initial weight matrix for the EM algorithm. The M step of the algorithm was run, producing a rough alignment pose. The pose was then evaluated using the E step of the EM algorithm, which computes the value of the objective function as a side effect (in addition to a new estimate of the weights). Thus, running one cycle of the EM algorithm, initialized by the pair of correspondences, generates a rough alignment, and evaluates the PMPE objective function for that alignment.

10.1.3 Refining Indexer Alignments

This section illustrates the method used to refine indexer alignments.

Figure 10-5 shows a closer view of the best scoring initial alignment from Angle Pair Indexing. The initial alignment was refined by running the EM algorithm to convergence using the coarse features and statistics. The result of this coarse refinement is displayed in Figure 10-6. The coarse refinement was refined further by running the EM algorithm to convergence with the fine features and statistics. The result of this fine refinement is shown in Figure 10-7, and over the video image in Figure 10-8.

Ground truth for the pose is available in this experiment, as the true pose is the null pose. The pose before refinement is

$$[.99595, -0.0084747, -0.37902, 5.0048]^T \ ,$$

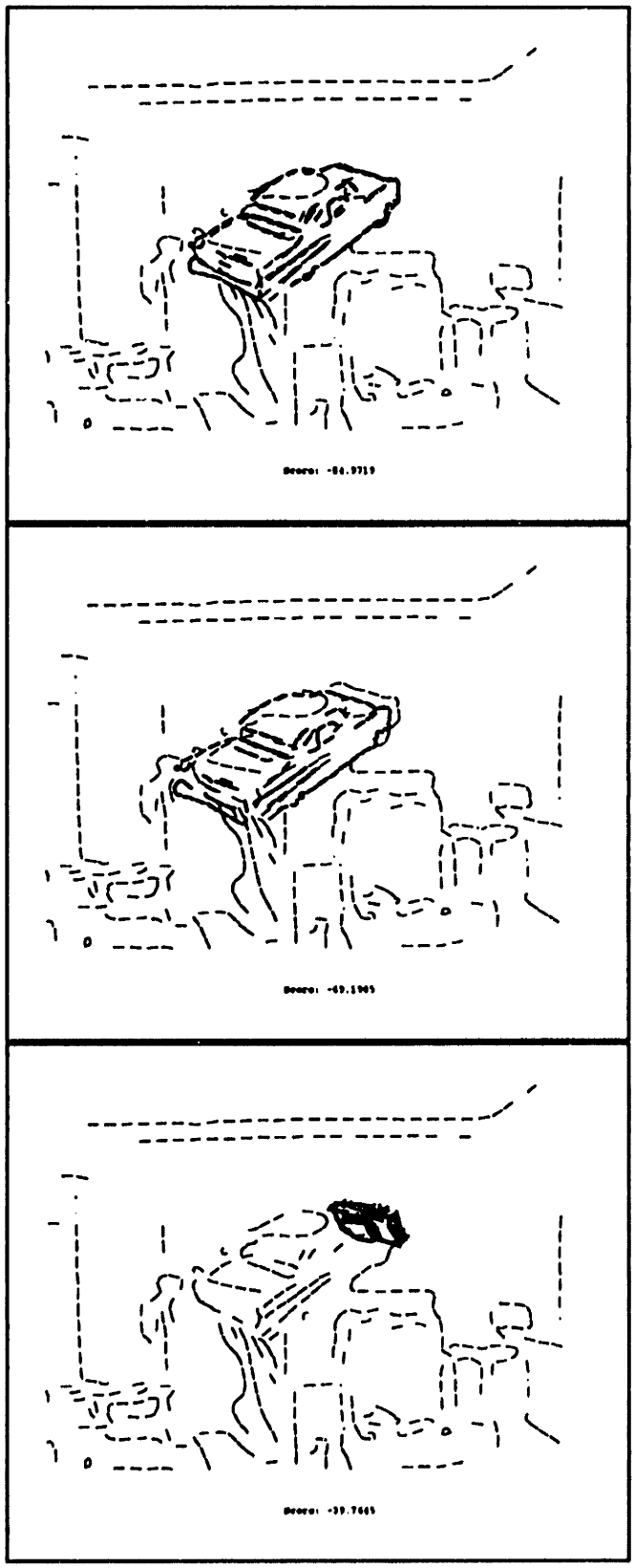


Figure 10-4: Poses and Scores of Some Indexed Hypotheses

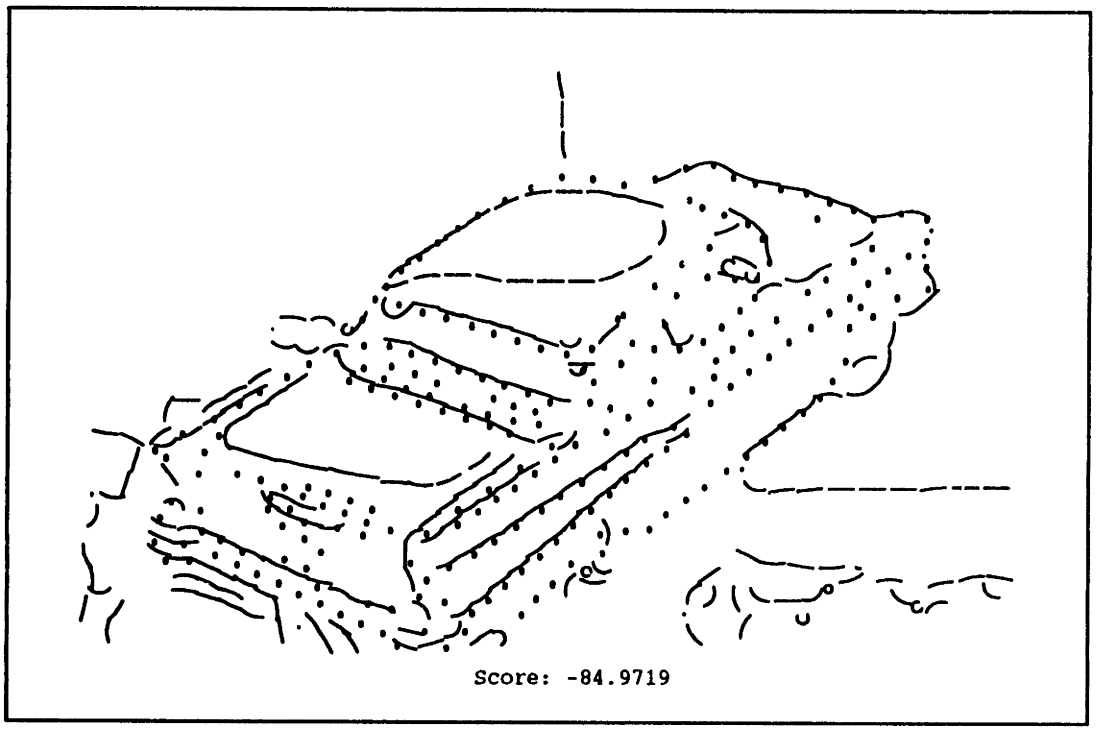


Figure 10-5: Best Alignment from Indexer, with Coarse Score

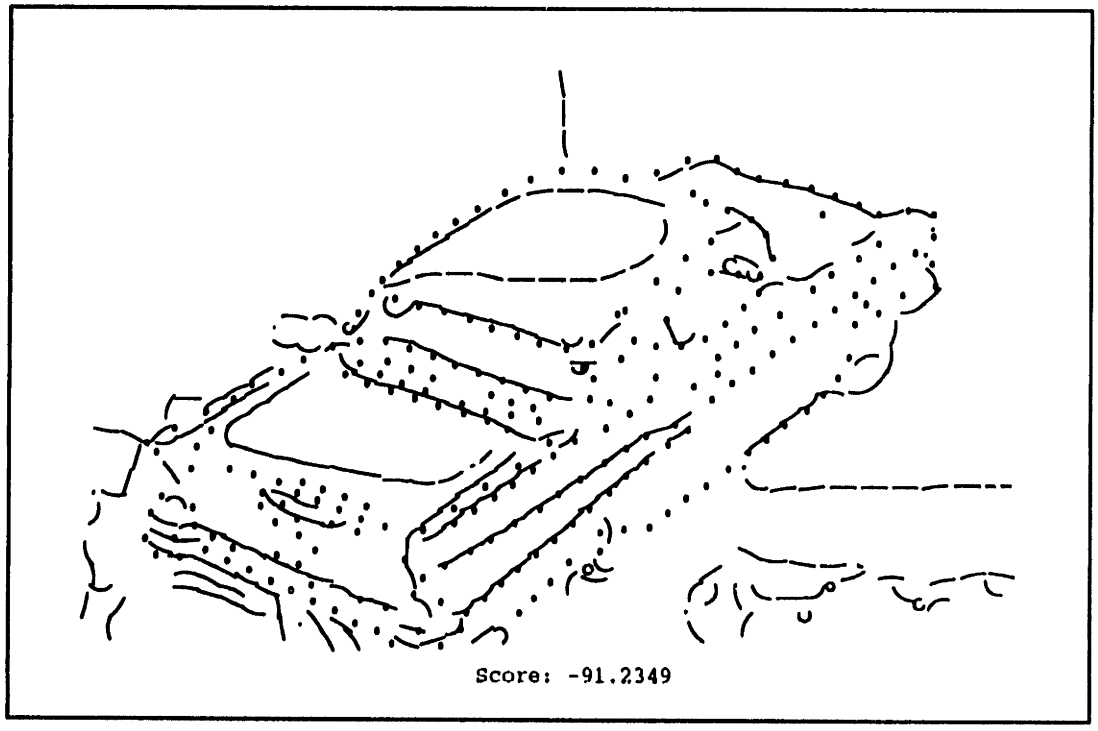


Figure 10-6: Coarse Refinement, with Coarse Score

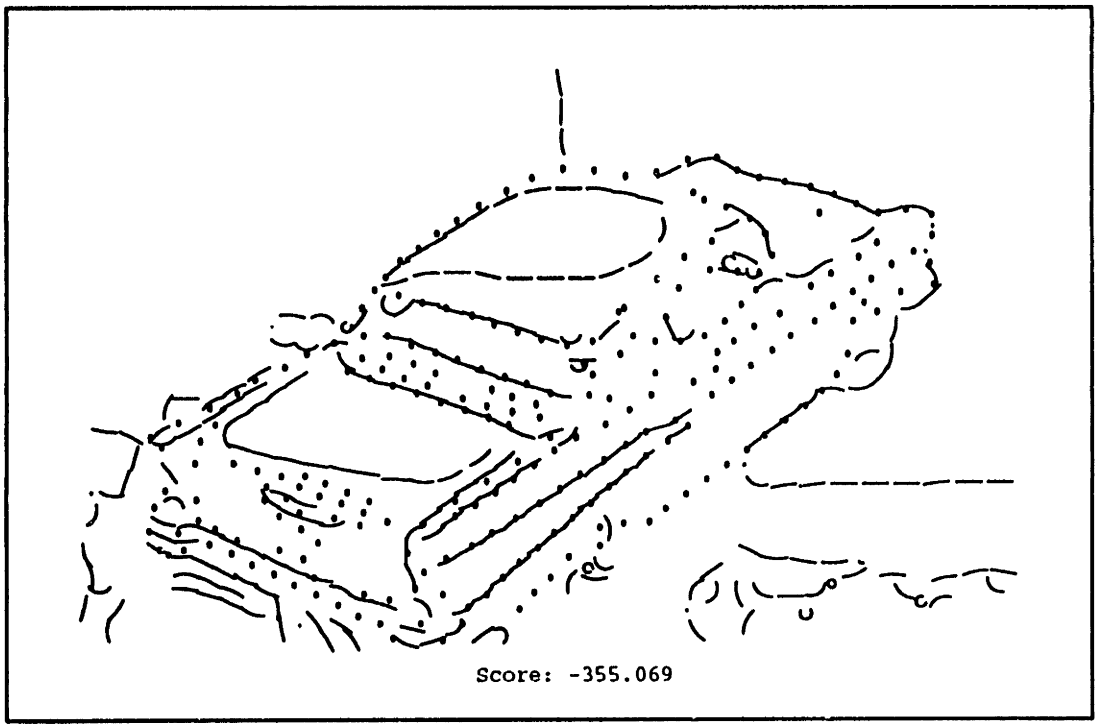


Figure 10-7: Fine Refinement, with Fine Score

Figure 10-8: Fine Refinement with Video Image

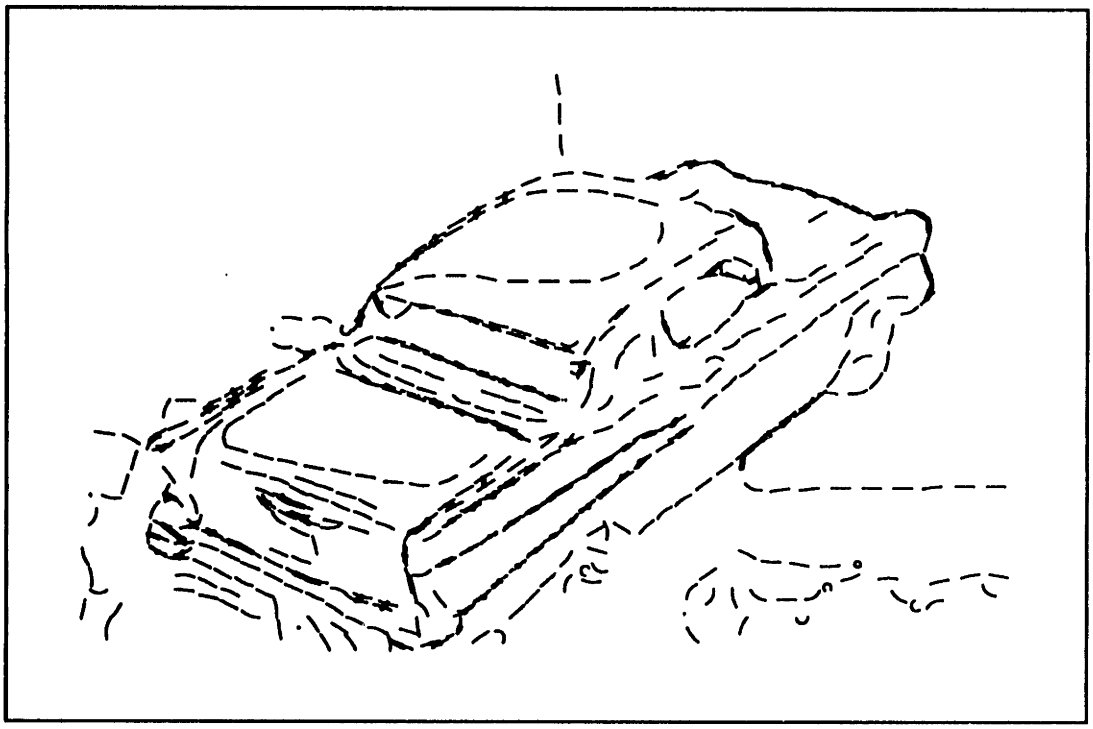


Figure 10-9: Correspondences with Weight Larger than .5

and after the refinement it is

$$[1.00166, 0.0051108, 0.68621, -1.7817]^T \ .$$

The encoding of these poses is described in Section 5.3 (the null pose is $[1,0,0,0]^T$.) The initial pose is in error by about .01 in scale and 5 pixels in position. The final pose errs by about .005 in scale and 1.8 pixels in position. Thus scale accuracy is improved by a factor of about two, and position accuracy is improved by factor of about three. An experiment showing more dramatic improvement is described below, in Section 10.4.1.

In these experiments, less that 15 iterations of the EM algorithm were needed for convergence.

10.1.4 Final EM Weights

As discussed in Section 8.1, a nice aspect of using the EM algorithm with PMPE is that estimates of feature correspondences are available in the weight matrix. Figure 10-9 displays the correspondences that have weight greater than .5, for the final convergence shown in Figure 10-7. Here, the image and model features are displayed as thin curves, and the correspondences between them are shown as heavy lines joining the features. Note the strong similarity between these correspondences, and those that the system was trained on, shown in Figure 3-2.

Table 10.1 displays the values of some of the weights. The weights show have value greater than .01. There are 299 weights this large among the 413,507 weights. The 39 weights shown are those belonging to 20 image features.

10.2 Evaluating Random Alignments

An experiment was performed to test the utility of PMPE in evaluating randomly generated alignments. Correspondences among the coarse features described in Section 10.1 having assignments from two image features to two model features were randomly generated, and evaluated as in Section 10.1.2. 19118 random alignments were generated, of which 1200 had coarse scores better than -30.0 (the negative of the PMPE objective function). Among these 1200, one was essentially correct. The first, second, third, fourth, fifth, and fifteenth best scoring alignments are shown in Figure 10-10.

With coarse – fine refinement, the best scoring random alignment converged to the same pose as the best refinement from the indexing experiment, shown in Figure 10-7, with fine score -355.069. The next best scoring random alignment converged to a grossly wrong pose, with fine score -149.064. This score provides some indication of the noise level in the fine PMPE objective function in pose space.

This test, though not exhaustive, produced no false positives, in the sense of a bad alignment with a coarse score better than that of the correct alignment. Additionally,

Image Index	Model Index	Weight
90	86	0.022738026840027032
90	101	0.014615921646994348
90	102	0.807966693444096
90	103	0.09581539482455806
91	103	0.9633441301926663
92	85	0.24166197059125494
92	103	0.19778274847425015
93	87	0.02784697957543993
93	88	0.37419218245379466
94	87	0.7478667723520142
95	87	0.44030413275215486
96	86	0.6127902576993082
97	85	0.9293665165549775
98	85	0.8621763443868999
99	84	0.9634827438267516
100	5	0.6499527214931725
100	84	0.19705210016850308
101	0	0.011400725934573982
101	67	0.9559675939354566
102	66	0.9194110795990801
102	67	0.0541643593533511
103	64	0.04765362703894284
103	65	0.8454128520499249
103	66	0.05787873660955701
104	63	0.05270908685541295
104	64	0.8854088356653954
104	65	0.014744194821866506
105	62	0.06158503222464117
105	63	0.9139939556525918
106	61	0.09270729594689026
106	62	0.8635739185353283
106	63	0.010447389024937672
107	61	0.9108899984969661
107	62	0.021204649868405194
108	60	0.861831671427887
108	61	0.049220125250993084
109	58	0.018077232316743887
109	59	0.9257311183042934
109	60	0.015434004217119308

Table 10.1: Some EM Weights

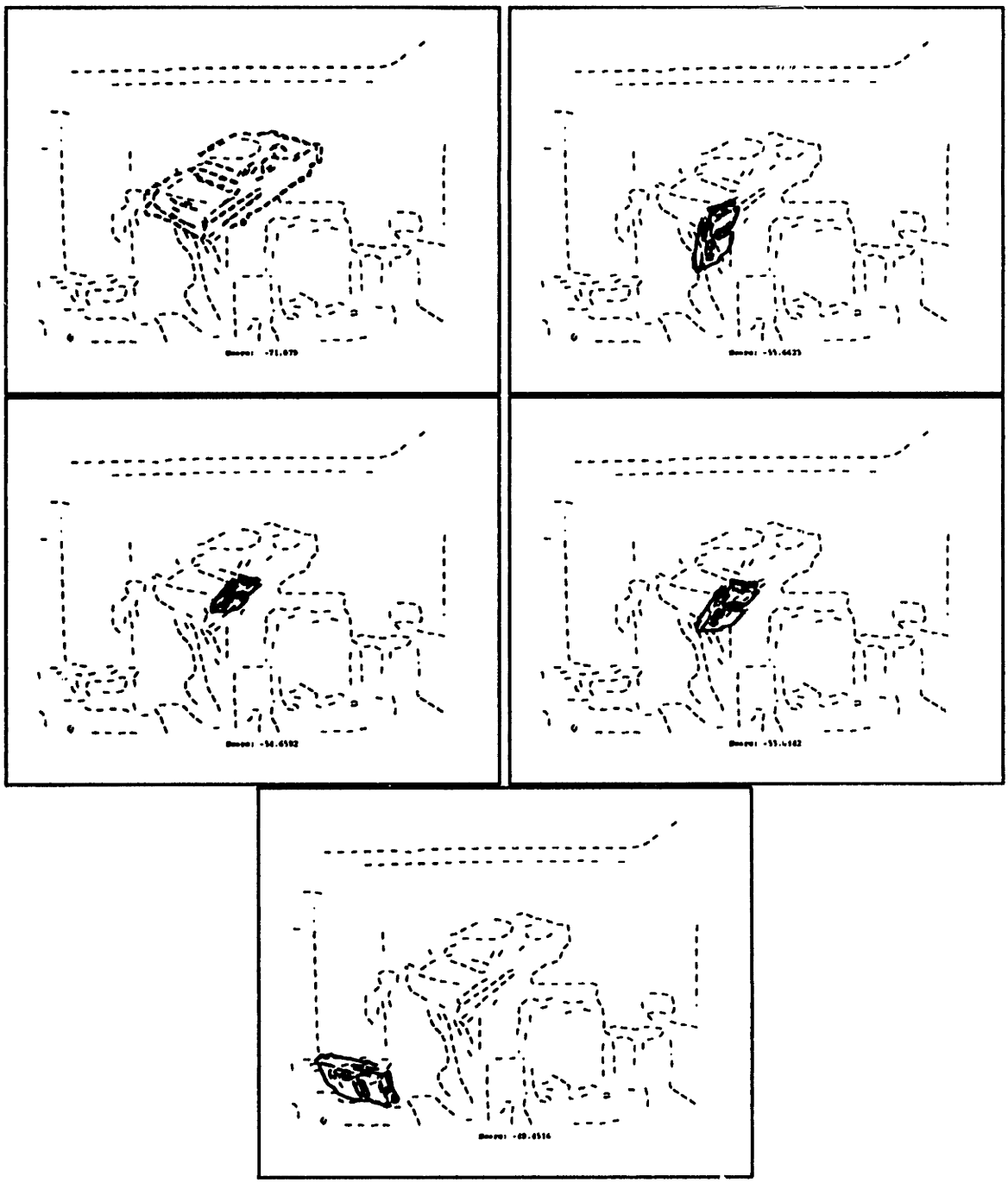


Figure 10-10: Random Alignments

the fine score of the refinement of the most promising incorrect random alignment was significantly lower than the fine score of the (correct) refined best alignment.

10.3 Convergence with Occlusion

The convergence behavior under occlusion of the EM algorithm with PMPE was evaluated using the coarse features described in Section 10.1. Images features simulating varying amounts of occlusion were prepared by shifting a varying portion of the image. These images, along with results of coarse – fine refinement using the EM algorithm are shown in Figure 10-11.

The starting value for the pose was the correct (null) pose. The refinements converged to good poses in all cases, demonstrating that the method can converge from good alignments with moderate amounts of occlusion.

The final fine score in the most occluded example is lower than the noise level observed in the experiment of Section 10.2. This indicates that as the amount of occlusion increases, a point will be reached where the method will fail to produce a good pose having a score above the noise level. In this experiment it happens before the method fails to converge properly.

10.4 3D Recognition Experiments

10.4.1 Refining 3D Alignments

This section demonstrates use of the EM algorithm with PMPE to refine alignments in 3D recognition. The linear combination of views method is used to accommodate a limited amount of out of plane rotation. A two-view variant of LCV, described in Section 5.5, is used.

A coarse – fine approach was used. Coarse PMPE scores were computed by smoothing the PMPE objective function, as described in Section 7.3.2. The smoothing

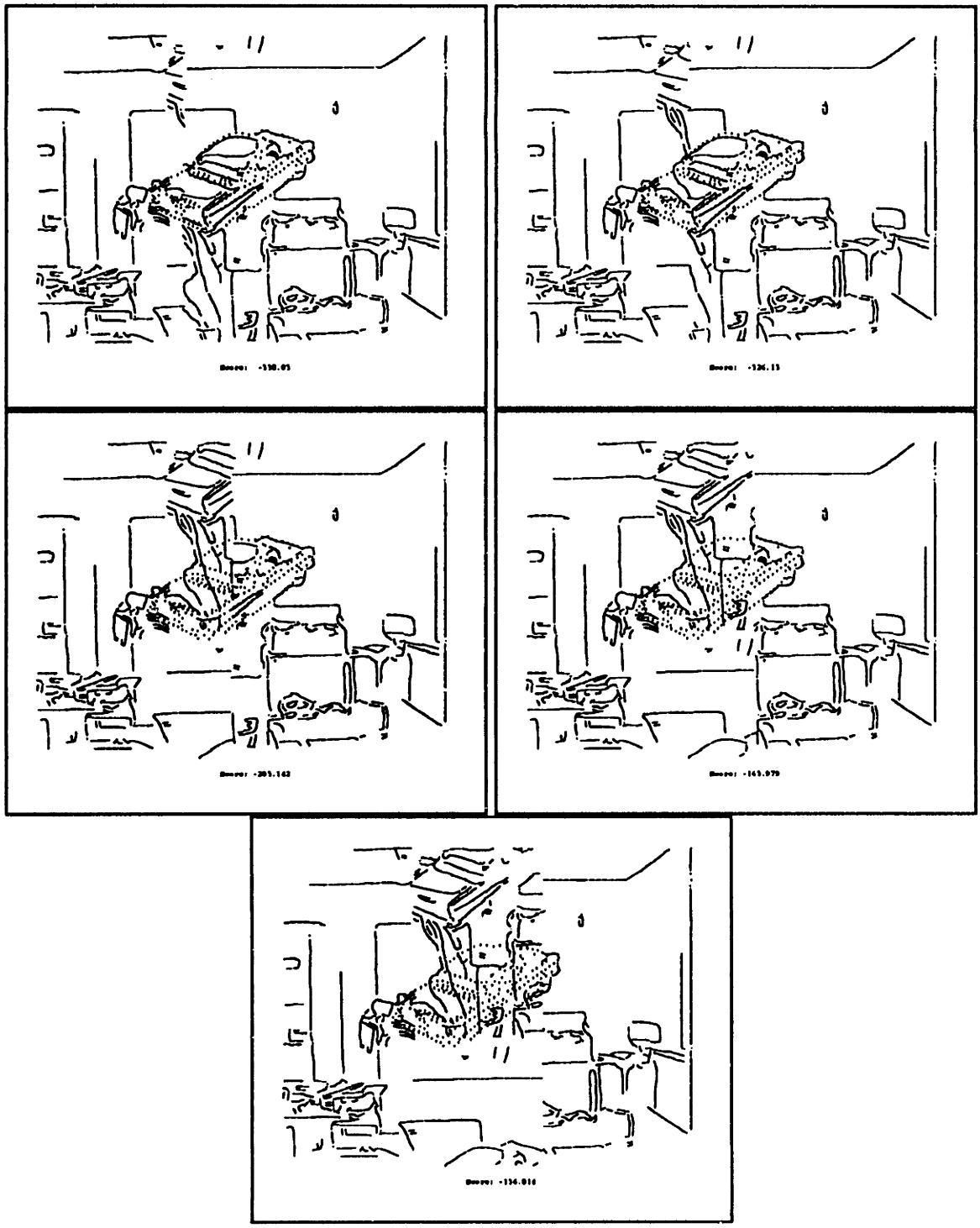


Figure 10-11: Fine Convergences with Occlusion

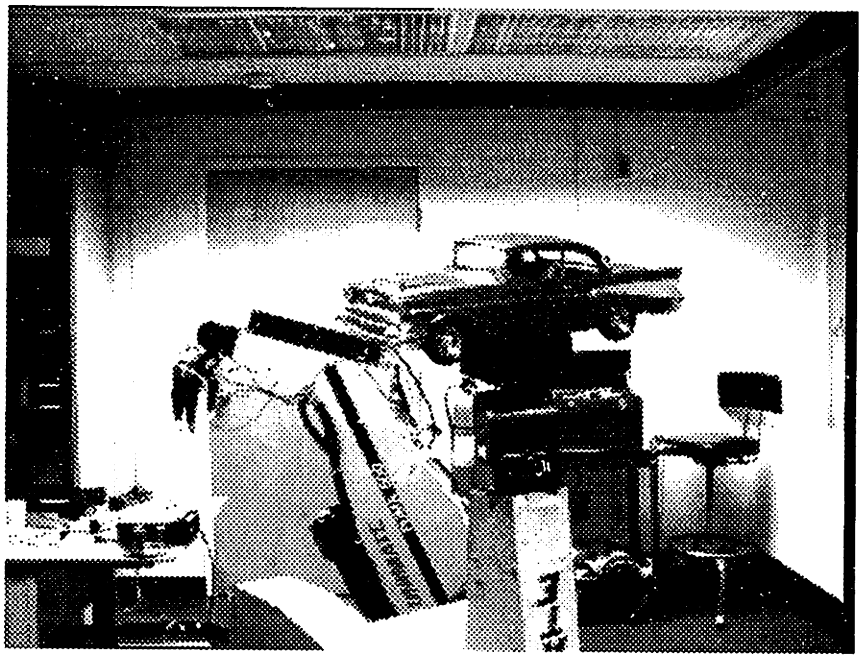


Figure 10-12: Grayscale Image

matrix was

$$\text{DIAG}((7.07)^2, (3.0)^2) \ .$$

These numbers are the amounts of additional (artificial) variance added for parallel and perpendicular deviations, respectively, in the oriented stationary statistics model.

The video test image is shown in Figure 10-12. It differs from the model images by a significant 3D translation and out of plane rotation. The test image edges are shown in Figure 10-13.

The object model was derived from the two Mean Edge Images shown in Figure 10-14. These were constructed as described in Section 4.4.

The smoothing used in preparation of the edge maps had 1.93 pixels standard deviation, and the edge curves were broken arbitrarily every 10 pixels. Point-radius features were fitted to the edge curve fragments, as described in Section 5.3, for purposes of display and for computing the oriented stationary statistics, although the features used with PMPE and the EM algorithm were simply the X and Y coordinates of the centroids of the curve fragments. Both views of the model features are shown

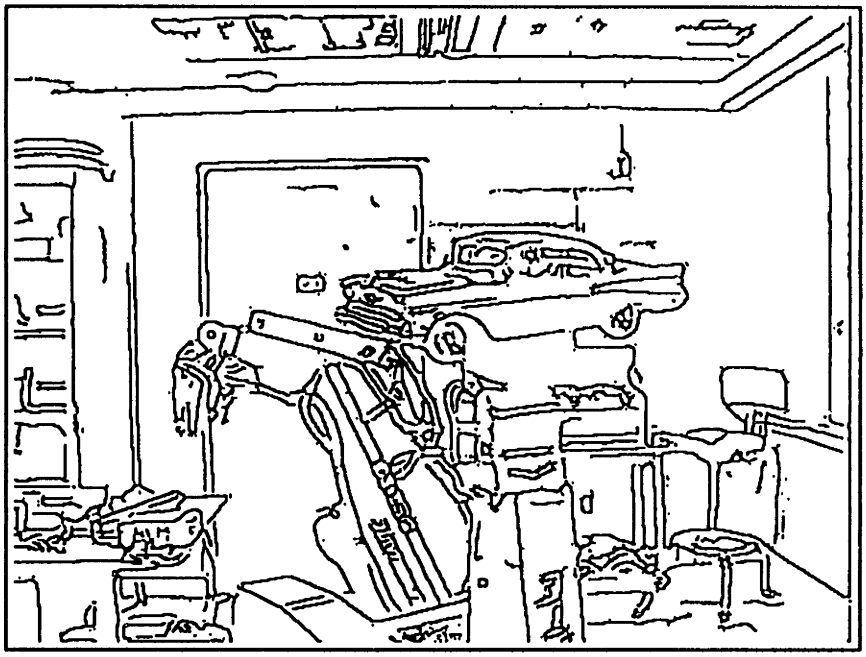


Figure 10-13: Image Edges

in Figure 10-15. The linear combination of views method requires correspondences among the model views. These were established by hand, and are displayed in Figure 10-16.

The relationship among the viewpoints in the model images and the test image is illustrated in Figure 10-17. This represents the region of the view sphere containing the viewpoints. Note that the test image is not on the line joining the two model views.

The oriented stationary statistics model of feature fluctuations was used (this is described in Section 3.3). As in Section 10.1, the parameters (statistics) that appear in the PMPE objective function, the background probability and the covariance matrix for the oriented stationary statistics, were derived from matches done by hand.

A set of four correspondences was established manually from the image features to the object features. These correspondences are intended to simulate an alignment generated by an indexing system. Indexing systems that are suitable for 3D recognition are described by Clemens and Jacobs [19] and Jacobs [45]. The rough alignment and score were obtained from the correspondences by one cycle of the EM algorithm,

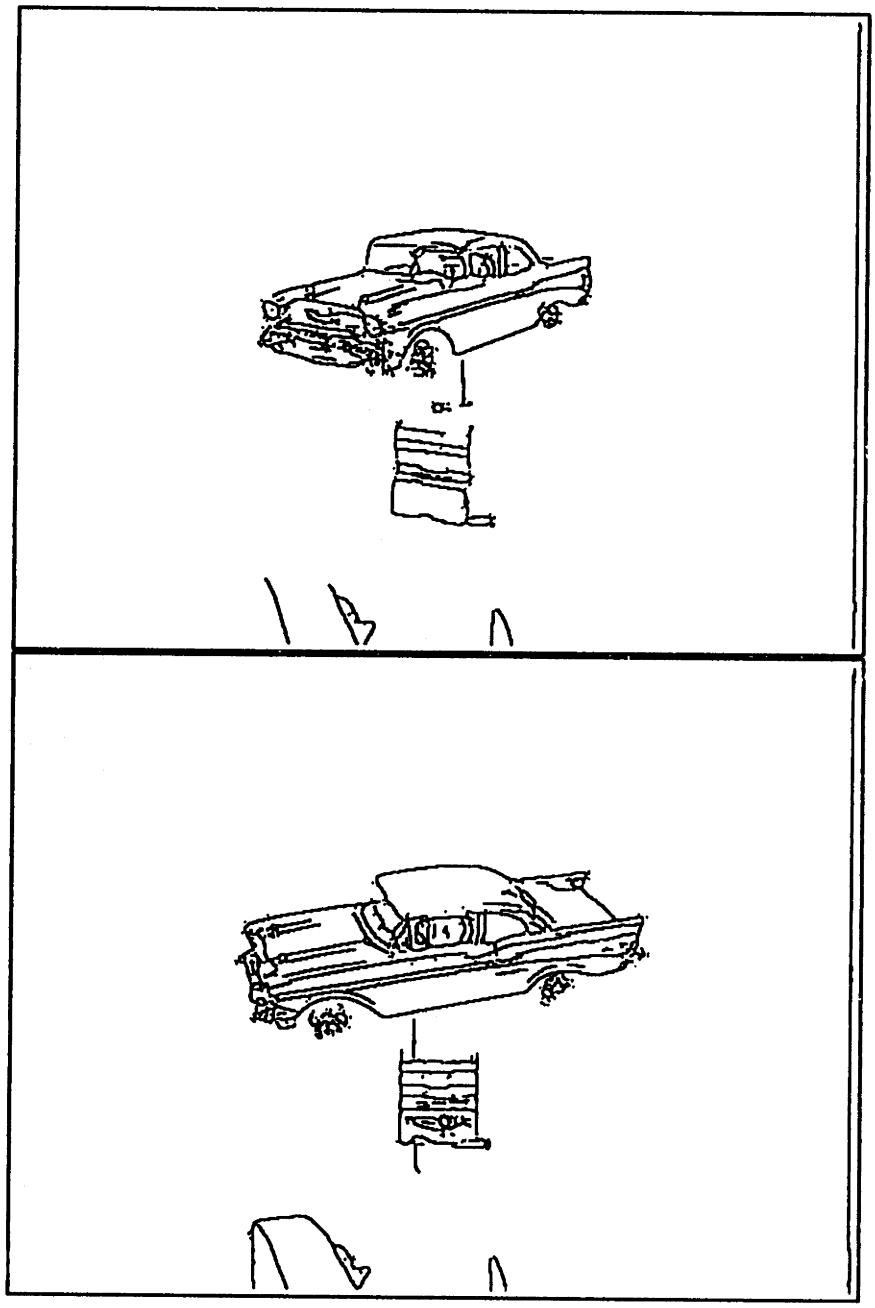


Figure 10-14: Model Mean Edge Images

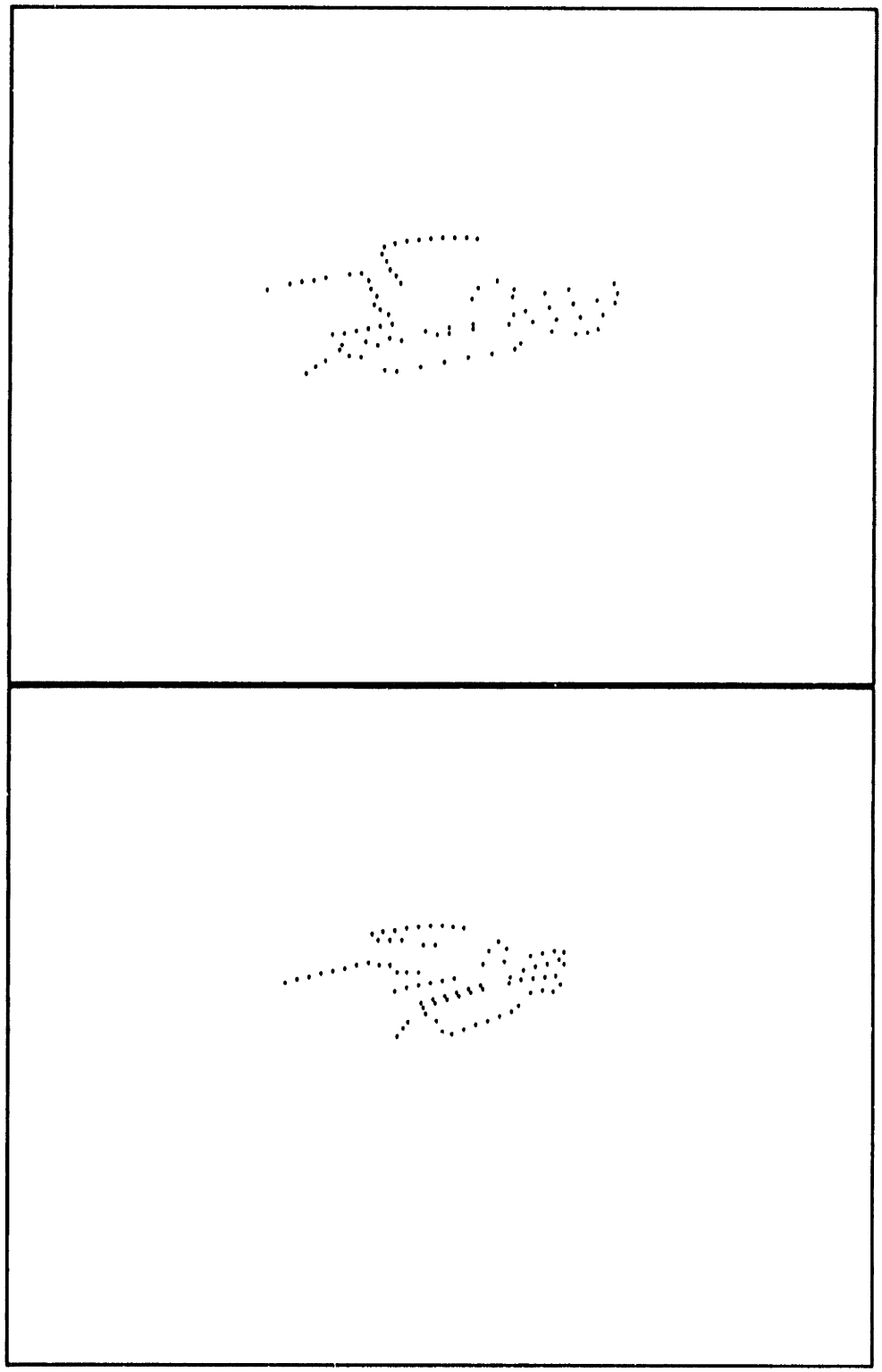


Figure 10-15: Model Features (Both Views)

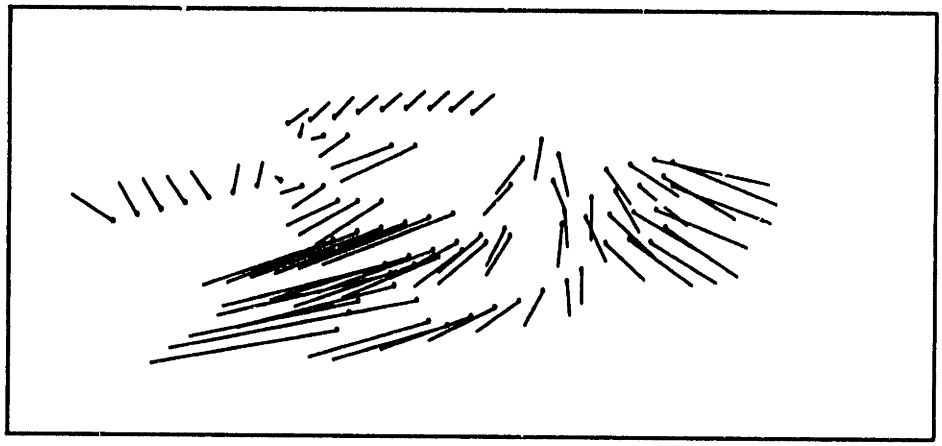


Figure 10-16: Model Correspondences

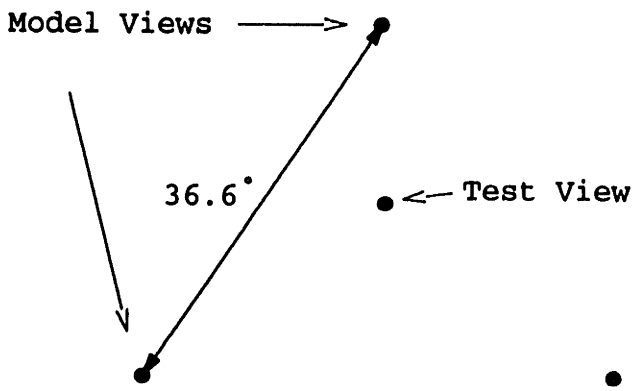


Figure 10-17: Model and Test Image View Points

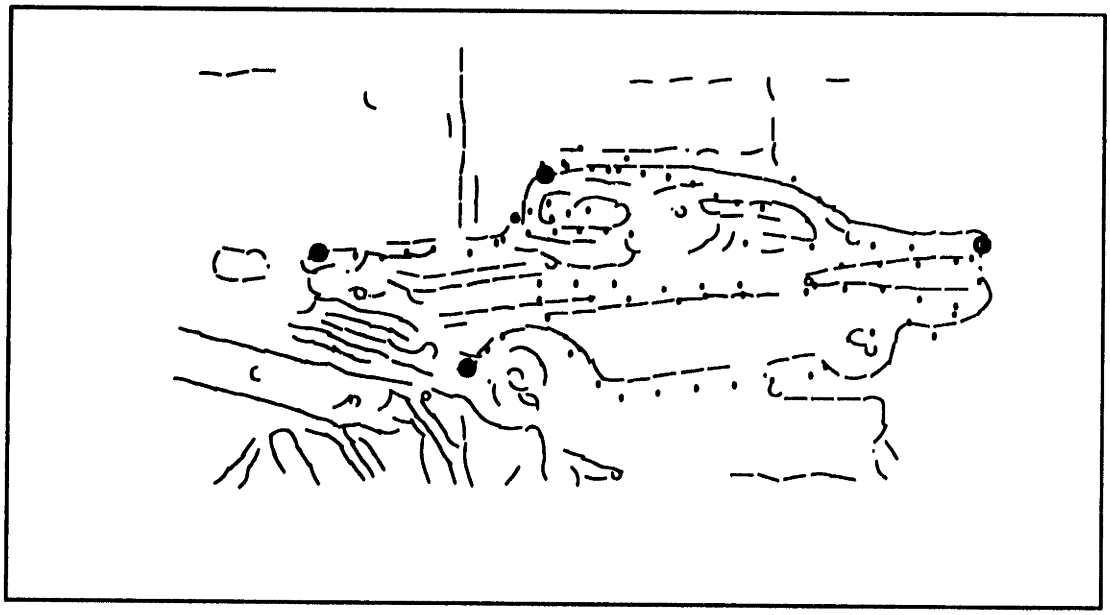


Figure 10-18: Initial Alignment

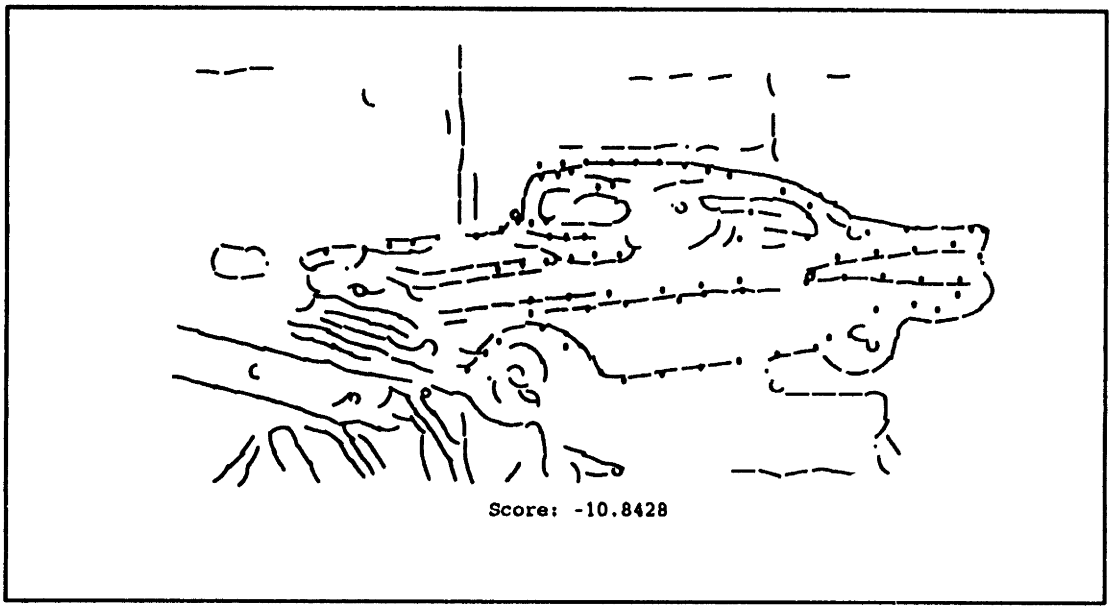


Figure 10-19: Coarse Refined Alignment and Coarse Score

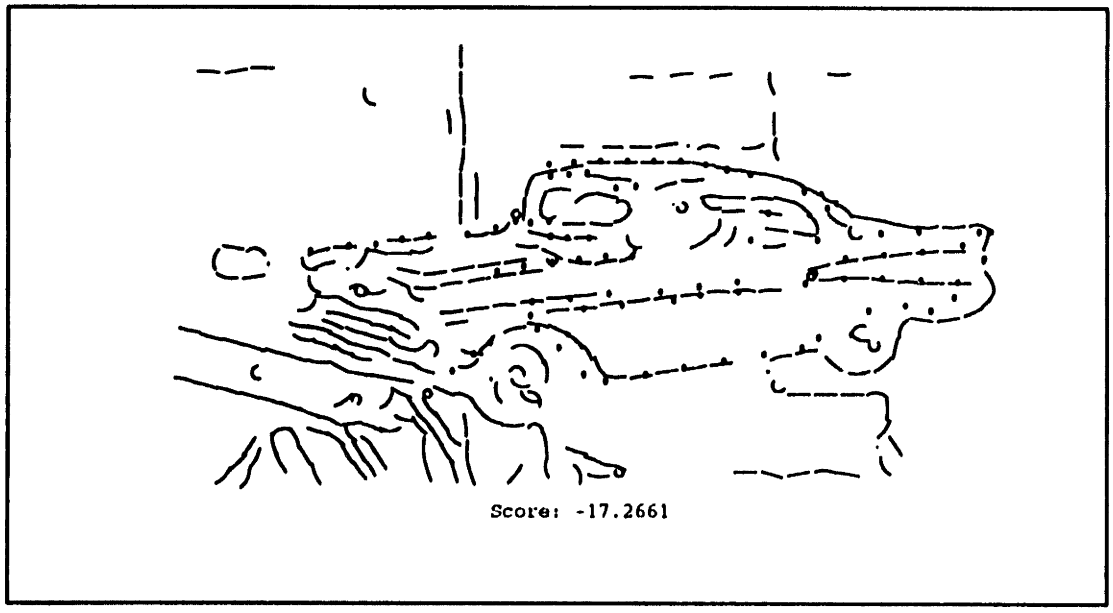


Figure 10-20: Fine Refined Alignment and Fine Score

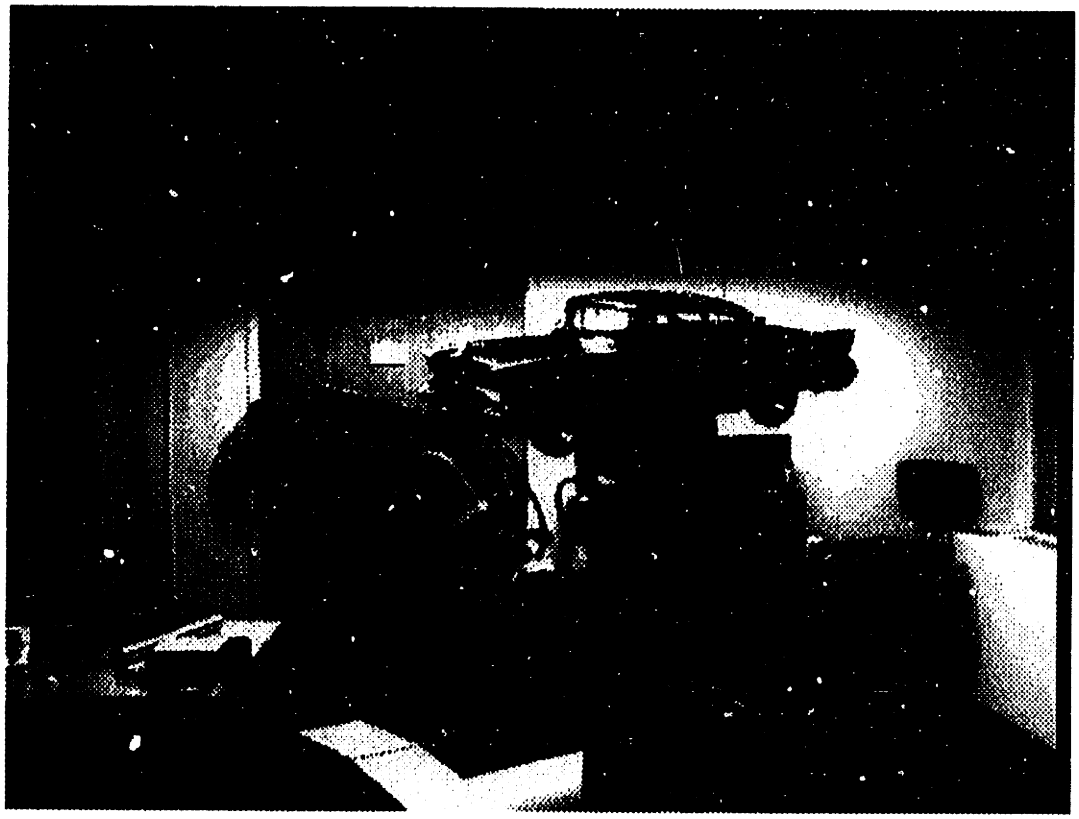


Figure 10-21: Fine Refined Alignment with Video Image

as described above in Section 10.1.2. They are displayed in Figure 10-18, where the four corresponding features appear circled. A coarse alignment was then obtained by running the EM algorithm to convergence with smoothing, the result appears in Figure 10-19. This alignment was refined further by running the EM algorithm again, without smoothing. The resulting alignment and score are shown in Figure 10-20. In these figures, the image features are shown as curve fragments for clarity, although only the point locations were used in the experiment. The image features used are a subset taken from a rectangular region of the larger image.

Figure 10-21 displays the final alignment superimposed over the original video image. Most of the model features have aligned well. The discrepancy in the forward wheel well may be due to inaccuracies in the LCV modeling process, perhaps in the feature correspondences. This figure demonstrates good results for aligning a smooth 3D object having six degrees of freedom of motion, without the use privileged features.

10.4.2 Refining Perturbed Poses

This section describes an additional demonstration of local search in pose space using PMPE in 3D.

The pose corresponding to the refined alignment displayed in Figure 10-20 was perturbed by adding a displacement by 4.0 pixels in Y. This pose was then refined by running the EM algorithm to convergence. The perturbed alignment and the resulting coarse – fine refinement is shown in Figure 10-22. The result is very close to the pose prior to perturbation.

A similar experiment was carried out with a larger perturbation, 12.0 pixels in Y. The results of this appear in Figure 10-23. This time the convergence is to a clearly wrong alignment. The model has been stretched to a thin configuration, and mismatched to the image. The resulting fine score is lower than that of the good alignment in Figure 10-21. This illustrates a potential drawback of the linear combination of views method. In addition to correct views, LCV can synthesize

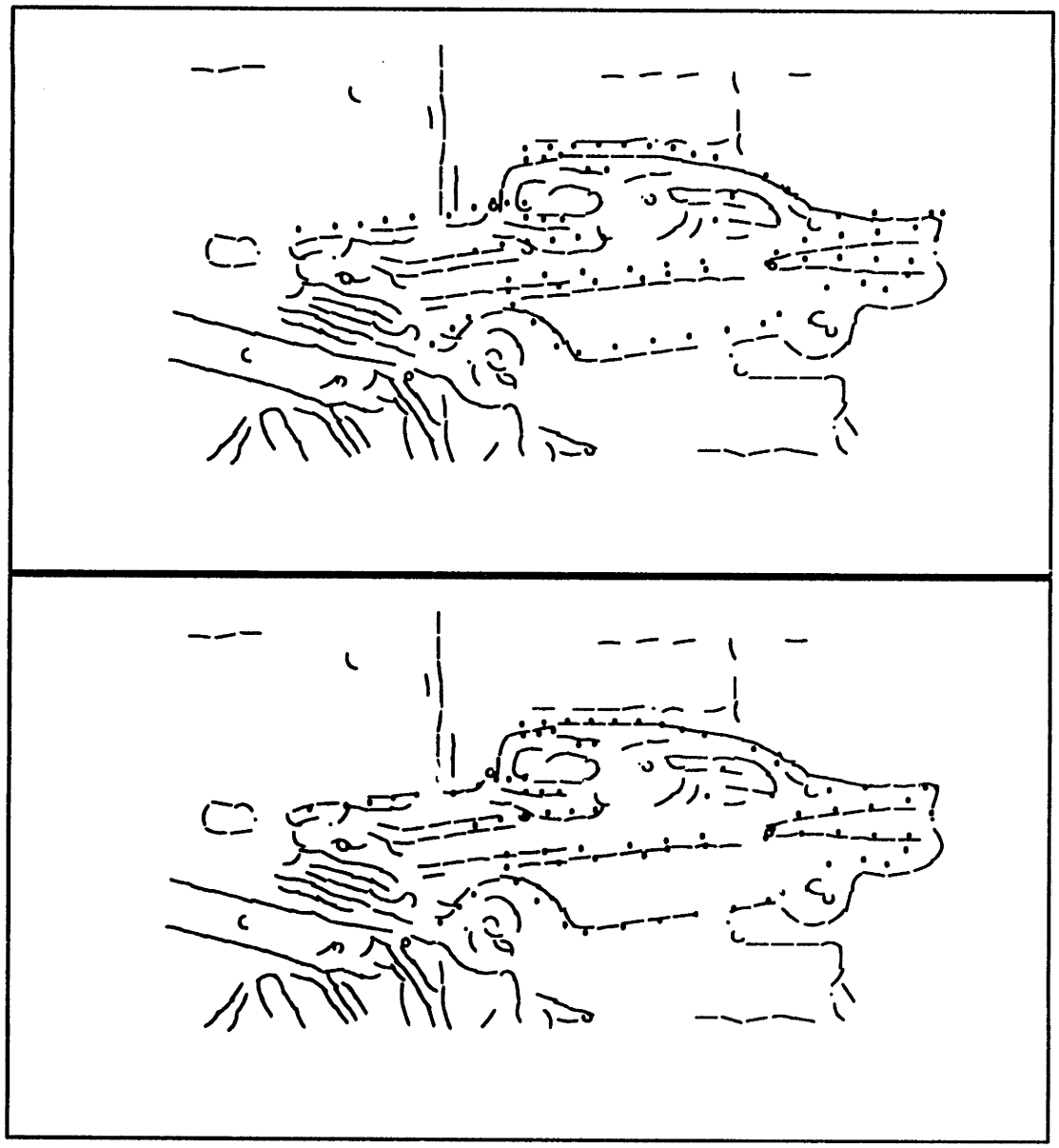


Figure 10-22: Mildly Perturbed Alignment and Resulting Refinement

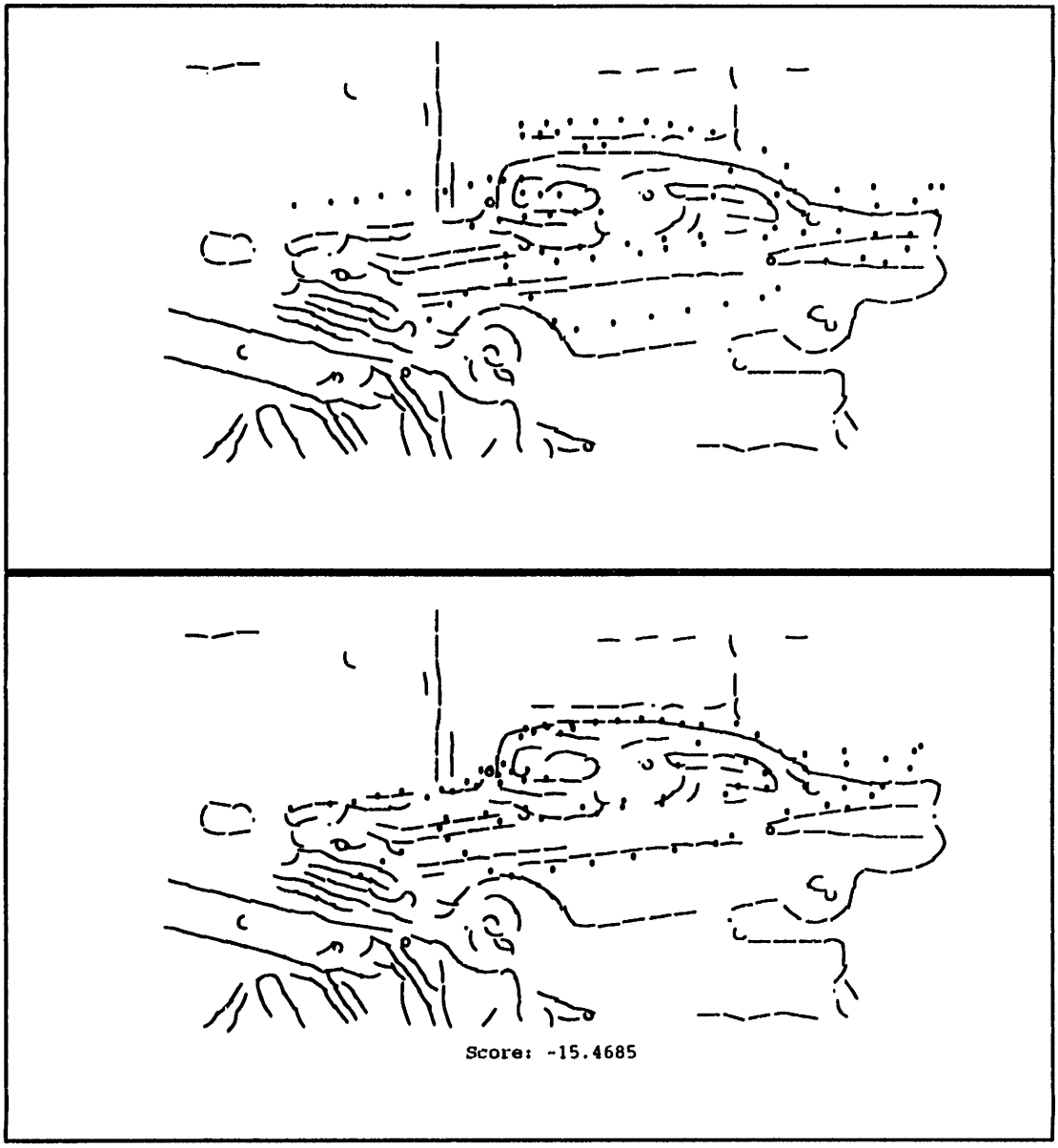


Figure 10-23: Perturbed Alignment and Resulting Refinement with Fine Score

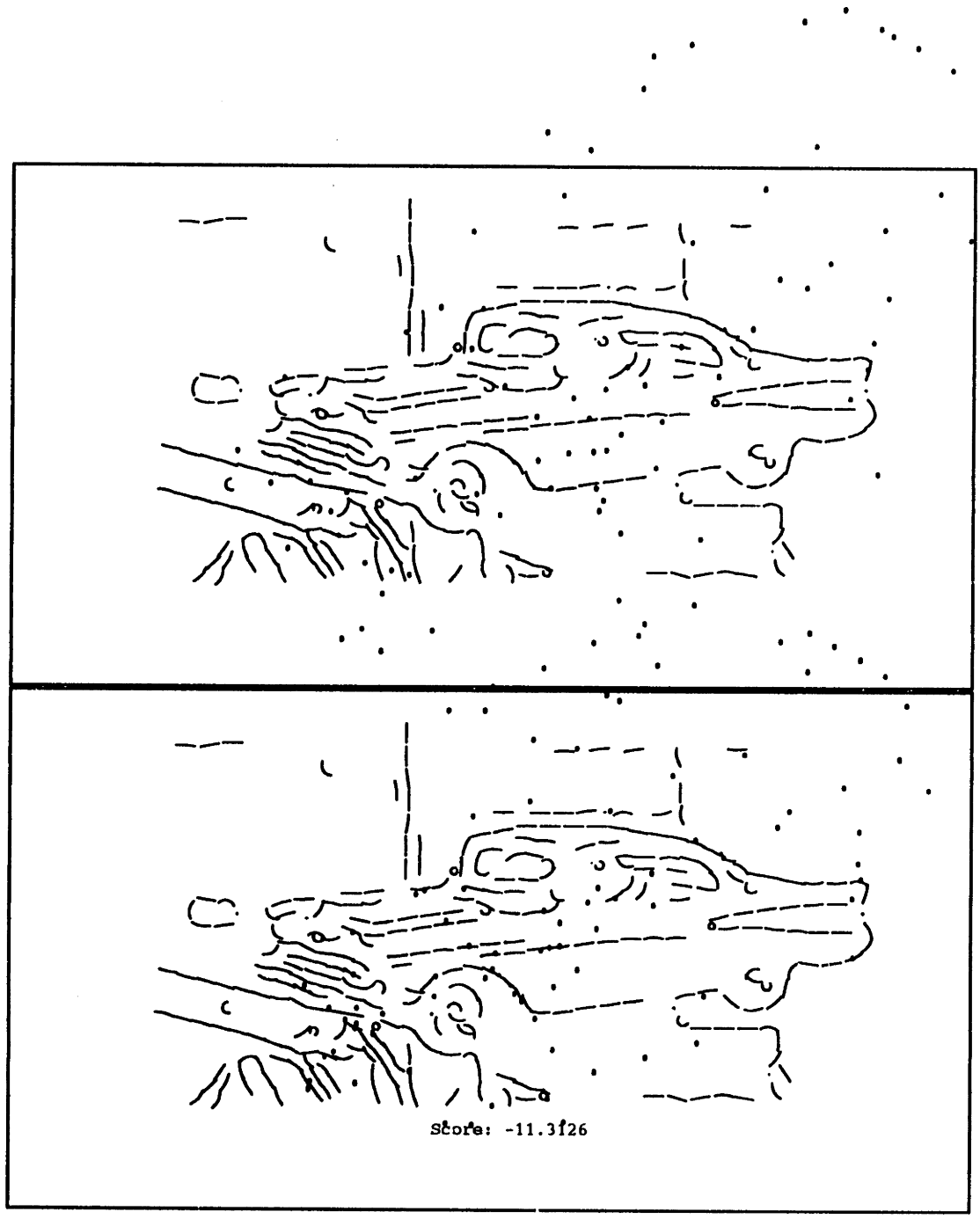


Figure 10-24: Bad Alignment and Resulting Refinement with Fine Score

views where the model is stretched. LCV, as used here, has 8 parameters, rather than the 6 of rigid motion. The two extra parameters determine the stretching part of the transformation. This problem can be addressed by checking, or enforcing, a quadratic constraint on the parameters. This is discussed in [71].

Another similar experiment was performed starting with a very bad alignment. The results appear in Figure 10-24. The algorithm was able to bring some features into alignment, but the score remained low.

Chapter 11

Conclusions

Visual object recognition – finding a known object in scenes, where the object is smooth, is viewed under varying illumination conditions, has six degrees of freedom of position, is subject to occlusions and appears against varying backgrounds – still presents problems. In this thesis, progress has been made by applying methods of statistical inference to recognition. Ever-present uncertainties are accommodated by statistical characterizations of the recognition problem: MAP Model Matching (MMM) and Posterior Marginal Pose Estimation (PMPE). MMM was shown to be effective for searching among feature correspondences and PMPE was shown effective for searches in pose space. The issue of acquiring salient object features under varying illumination was addressed by using Mean Edge Images.

The alignment approach, which leverages fast indexing methods of hypothesis generation, is utilized. Angle Pair Indexing is introduced as an efficient 2D indexing method that does not depend on extended or special features that can be hard to detect. An extension to the alignment approach that may be summarized as *align refine verify* is advocated. The EM algorithm is employed for refining the estimate of the object's pose while simultaneously identifying and incorporating the constraints of all supporting image features.

Areas for future research include the following:

- Indexing was not used in the 3D recognition experiments. Identifying a suitable mechanism for this purpose that meshes well with the type of features used here, would be an improvement.

- Too few views were used in model construction. Fully automating the model acquisition process, as described in Chapter 4, and acquiring models from more views would help.

- Extending the formulations of recognition to handle multiple objects is straightforward, but identifying suitable search strategies is an important and non-trivial task.

- Incorporating non-linear models of projection into the formulation would allow robust performance in domains having serious perspective distortions.

- Using image-like tables could speed the evaluation of the PMPE objective function.

- Investigating the use of PMPE in object tracking or in other active vision domains might prove fruitful.

More work in these areas will lead to practical and robust object recognition systems.

Appendix A

Notation

Symbol	*Meaning*	*Defining Section*
$Y = \{Y_1, Y_2, \ldots, Y_n\}$	the image	2.1
n	number of image features	
$Y_i \in R^v$	image feature	2.1
$M = \{M_1, M_2, \ldots, M_m\}$	the object model	2.1
m	number of object features	
M_j	model feature, frequently $M_j \in R^{v \times z}$	2.1
$\perp$	the background feature	2.1
$\Gamma = \{\Gamma_1, \Gamma_2, \ldots, \Gamma_n\}$	correspondences	2.1
$\Gamma_i \in M \cup \{\perp\}$	assignment of image feature i	2.1
$\beta \in R^z$	pose of object	5.1
$\mathcal{P}(M_j, \beta)$	projection into image	5.1
$\mathrm{G}_\psi(x)$	Gaussian probability density	3.2 6.1
ψ_{ij}	covariance matrix of feature pair	3.3
$\hat{\psi}$	stationary feature covariance matrix	3.3
ψ_β	covariance matrix of pose prior	6.1
B, B_i	background probability	2.2 2.4
W_k	extent of image feature dimension k	3.1
λ_{ij}, λ	correspondence reward	6.1
$\hat{x}$	estimate of x	
$p(\cdot)$	probability	(see below)

Probability notation is somewhat abused in this work, in the interest of brevity. $p(x)$ may stand for either a probability mass function of a discrete variable x, *or* for a probability density function of a continuous variable x. The meaning will be clear in

context based on the type of the variable argument. Additionally, mixed probabilities are described with the same notation. For example $p(\Gamma, \beta \mid Y)$ stands for the mixed probability function that is a probability mass function of Γ (the discrete variable describing correspondences), and a probability density function of β (the pose vector) – both conditioned on Y (the image feature coordinates).

Bibliography

[1] N. Ayache and O.D. Faugeras. HYPER: A New Approach for the Recognition and Positioning of Two-Dimensional Objects. *IEEE Transactions PAMI*, PAMI-8(1):44–54, January 1986.

[2] S.T. Barnard. Stereo Matching by Hierarchical, Microcanonical Annealing. In *Proceedings of the Tenth International Joint Conference on Artificial Intelligence*, pages 832–835, August 1987.

[3] H.G. Barrow, J.M. Tenenbaum, R.C. Bolles, and H.C. Wolf. Parametric Correspondence and Chamfer Matching: Two New Techniques for Image Matching. In *Proc. 5th Int. Joint Conf. Artificial Intelligence*, pages 659–663, 1977.

[4] J.V. Beck and K.J. Arnold. *Parameter Estimation in Science and Engineering.* John Wiley & Sons, 1977.

[5] P.J. Besl and R.C. Jain. Three-Dimensional Object Recognition. *Computing Surveys*, 17:75–145, 1985.

[6] J. Beveridge, R. Weiss, and E. Riseman. Optimization of 2-Dimensional Model Matching. In *Proceedings: Image Understanding Workshop*, pages 815–830. Morgan Kaufmann, 1989.

[7] A. Blake and A. Zisserman. *Visual Reconstruction.* MIT Press, 1987.

[8] G. Borgefors. Hierarchical Chamfer Matching: A Parametric Edge Matching Algorithm. *IEEE Transactions PAMI*, 10(6):849–865, November 1988.

[9] T.M. Breuel. *Geometric Aspects of Visual Object Recognition.* PhD thesis, MIT Department of Brain and Cognitive Sciences, 1992.

[10] R.A. Brooks. Model-Based Three-Dimensional Interpretations of Two-Dimensional Images. *IEEE Transactions PAMI*, PAMI-5(2):140 - 150, March 1983.

[11] J.B. Burns. *Matching 2D Images to Multiple 3D Objects Using View Description Networks.* PhD thesis, University of Massachusetts at Amherst, Dept. of Computer and Information Science, 1992.

[12] J.B. Burns and E.M. Riseman. Matching Complex Images to Multiple 3D Objects Using View Description Networks. In *Proceedings: Image Understanding Workshop*, pages 675–682. Morgan Kaufmann, January 1992.

[13] J.F. Canny. A Computational Approach to Edge Detection. *IEEE Transactions PAMI*, PAMI-8(6):679–698, November 1986.

[14] T.A. Cass. A Robust Parallel Implementation of 2D Model-Based Recognition. In *Proceedings of the Computer Society Conference on Computer Vision and Pattern Recognition*, pages 879–884. IEEE, June 1988.

[15] T.A. Cass. *Polynomial-Time Geometric Matching for Object Recognition.* PhD thesis, MIT Department Electrical Engineering and Computer Science, 1992.

[16] T.A. Cass. Polynomial-Time Object Recognition in the Presence of Clutter, Occlusion, and Uncertainty. In G. Sandini, editor, *Computer Vision - ECCV '92*, pages 834–851. Springer Verlag, 1992.

[17] P. Cheeseman. A Method of Computing Generalized Bayesian Probability Values for Expert Systems. In *Proc. Eighth Int. Joint Conf. Artificial Intelligence*, pages 198–202, 1983.

[18] R.T. Chin and C.R. Dyer. Model-Based Recognition in Robot Vision. *Computing Surveys*, 18:67–108, 1986.

[19] D.T. Clemens and D.W. Jacobs. Model Group Indexing for Recognition. In *Symposium on Advances in Intelligent Systems*. SPIE, 1990.

[20] D.T. Clemens and D.W. Jacobs. Space and Time Bounds on Indexing 3-D Models from 2-D Images. *IEEE Transactions PAMI*, 13(10):1007–1017, October 1991.

[21] A.P. Dempster, N.M. Laird, and D.B. Rubin. Maximum Likelihood from Incomplete Data via the EM Algorithm. *J. Roy. Statist. Soc.*, 39:1 - 38, 1977.

[22] R.O. Duda and P.E. Hart. *Pattern Classification and Scene Analysis*. John Wiley and Sons, 1973.

[23] P. Dykstra and M.J. Muuss. The BRL CAD Package: an Overview. In *Proceedings of the Fourth USENIX Computer Graphics Workshop*, pages 73–80, Cambridge MA, 1987.

[24] S. Edelman and T. Poggio. Bringing the Grandmother Back Into the Picture: a Memory-Based View of Object Recognition. A.I. Memo 1181, Massachusetts Institute of Technology, April 1990.

[25] D.I. Perrett et al. Visual Cells in the Temporal Cortex Sensitive to Face View and Gaze Control. *Proc. Roy. Soc. London B*, 223:293 - 317, 1985.

[26] P. Fua and A.J. Hanson. Objective Functions for Feature Discrimination: Applications to Semiautomated and Automated Feature Extraction. In *Proceedings: Image Understanding Workshop*, pages 676–694. Morgan Kaufmann, 1989.

[27] P. Fua and A.J. Hanson. Objective Functions for Feature Discrimination: Theory. In *Proceedings: Image Understanding Workshop*, pages 443–459. Morgan Kaufmann, 1989.

[28] S. Geman and D. Geman. Stochastic Relaxation, Gibbs Distributions, and the Bayesian Restoration of Images. *IEEE Transactions PAMI*, PAMI-6(6):721–741, November 1984.

[29] C. Goad. Fast 3-D Model Based Vision. In Alex P. Pentland, editor, *From Pixels to Predicates*, pages 371–391. Ablex Publishing Co., 1986.

[30] S. A. Goldman. Efficient Methods for Calculating Maximum Entropy Distributions. Technical Report TR-391, MIT Laboratory for Computer Science, 1987.

[31] T.J. Green, Jr. *Three-Dimensional Object Recognition Using Laser Radar.* PhD thesis, MIT Department Electrical Engineering and Computer Science, 1992.

[32] T.J. Green, Jr. and J.H. Shapiro. Maximum-Likelihood Laser Radar Range Profiling with the Expectation – Maximization Algorithm. *Opt. Eng.*, November 1992.

[33] W.E.L. Grimson. Computational Experiments with a Feature Based Stereo Algorithm. *IEEE Transactions PAMI*, PAMI-7(1):17–34, January 1985.

[34] W.E.L. Grimson. *Object Recognition by Computer: The Role of Geometric Constraints.* MIT Press, 1990.

[35] W.E.L. Grimson and D.P. Huttenlocher. On the Verification of Hypothesized Matches in Model-Based Recognition. In *First Europ. Conf. Comp. Vision*, pages 489–498, 1990.

[36] W.E.L. Grimson and T. Lozano-Pérez. Localizing Overlapping Parts by Searching the Interpretation Tree. *IEEE Transactions PAMI*, PAMI-9(4):469–482, July 1987.

[37] R.M. Haralick. Digital Step Edges from Zero Crossing of Second Directional Derivatives. *IEEE Transactions PAMI*, PAMI-6(1):58 - 129, January 1984.

[38] R.M. Haralick, H.Joo, C.N.Lee, X.Zhuang, V.G.Vaidya, and M.B.Kim. Pose Estimation from Corresponding Point Data. *IEEE Trans. on Systems Man and Cybernetics*, 19(6):1426 - 1445, December 1989.

[39] B.K.P. Horn. *Robot Vision.* McGraw-Hill, New York, 1986.

[40] P.V.C. Hough. Methods and Means for Recognizing Complex Patterns. U.S. Patent 3069654, 1962.

[41] P.J. Huber. *Robust Statistics.* J. Wiley & Sons, 1981.

[42] D.P. Huttenlocher, K. Kedem, K. Sharir, and M. Sharir. The Upper Envelope of Voronoi Surfaces and its Applications. In *Proceedings of the Seventh ACM Symposium on Computational Geometry*, pages 194–293, 1991.

[43] D.P. Huttenlocher and S. Ullman. Recognizing Solid Objects by Alignment. In *Proceedings: Image Understanding Workshop*, pages 1114–1124. Morgan Kaufmann, April 1988.

[44] J. Illingworth and J. Kittler. A Survey of the Hough Transform. *Computer Vision, Graphics, and Image Processing*, 44:87–116, 1988.

[45] D.W. Jacobs. *Recognizing 3D Objects Using 2D Images.* PhD thesis, MIT Department Electrical Engineering and Computer Science, 1992.

[46] E. T. Jaynes. Where Do We Go From Here? In C. R. Smith and W. T. Grandy, Jr., editors, *Maximum-Entropy and Bayesian Methods in Inverse Problems*, pages 21 - 58. MIT Press, 1982.

[47] H. Jiang, R.A. Robb, and K.S. Holton. A New Approach to 3-D Registration of Multimodality Medical Images by Surface Matching. In *Visualization in Biomedical Computing*, pages 196–213. SPIE, 1992.

[48] R. Kumar and A.R. Hanson. Robust Estimation of Camera Location and Orientation from Noisy Data Having Outliers. In *Proc. of the Workshop on Interpretation of 3D Scenes*, pages 52–60. IEEE Computer Society, 1989.

[49] Y. Lamdan and H.J. Wolfson. Geometric Hashing: A General and Efficient Model-Based Recognition Scheme. In *Second Int. Conf. Comp. Vision*, 1988.

[50] P. Lipson. Model Guided Correspondence. Master's thesis, MIT Department Electrical Engineering and Computer Science, 1992.

[51] D.G. Lowe. *Perceptual Organization and Visual Recognition.* Kluwer Academic Publishers, 1985.

[52] D. Marr. *Vision.* Freeman, 1982.

[53] D. Marr and E. Hildreth. Theory of Edge Detection. *Proc. Roy. Soc. London B*, 207:187 - 217, 1980.

[54] J.L. Marroquin. *Probabilistic Solution of Inverse Problems.* PhD thesis, MIT Department Electrical Engineering and Computer Science, 1985.

[55] J.L. Marroquin, S. Mitter, and T. Poggio. Probabilistic Solution of Ill-posed Problems in Computational Vision. *Journal of the Am. Stat. Assoc.*, 82(397):76–89, 1987.

[56] M. Menon and W.M. Wells III. Massively Parallel Image Restoration. In *Proceedings of the International Joint Conference on Neural Networks*, San Diego, CA., 1990. IEEE.

[57] D. Mercer. *Optimization of a Curvature-Gradient Dot-Product Line and Edge Detector.* Bachelor's Thesis, MIT Department of Electrical Engineering and Computer Science, 1991.

[58] J. Ponce and D.J. Kriegman. On Recognizing and Positioning Curved 3D Objects From Image Contours. In *Image Understanding Workshop (Palo Alto, CA, May 23-26, 1989)*, pages 461–470, 1989.

[59] W.H. Press, B.P. Flannery, S.A. Teukolsky, and W.T. Vetterling. *Numerical Recipes: the Art of Scientific Computing.* Cambridge University Press, 1986.

[60] E. Rivlin and R. Basri. Localization and Positioning Using Combinations of Model Views. Technical Report CAR-TR-631, Center for Automation Research, University of Maryland, 1992.

[61] T.J. Sejnowski and C.R. Rosenberg. Parallel Networks that Learn to Pronounce English Text. *Complex Systems*, 1:145–168, 1987.

[62] J.H. Shapiro, R.W. Reinhold, and D. Park. Performance Analyses for Peak-Detecting Laser Radars. *Proceedings of the SPIE*, 663:38–56, 1986.

[63] J.H. Shapiro and A.L. Willsky. Stochastic Processes, Detection, and Estimation. MIT Course 6.432 Supplementary Notes.

[64] S.C. Shapiro, editor. *Encyclopedia of Artificial Intelligence.* John Wiley & Sons, 1987.

[65] A. Shashua. Correspondence and Affine Shape from two Orthographic Views: Motion and Recognition. A.I. Memo 1327, Massachusetts Institute of Technology, December 1991.

[66] A. Shashua and S. Ullman. Structural Saliency: The Detection of Globally Salient Structures Using a Locally Connected Network. In *Proceedings of the 2nd International Conference on Computer Vision*, pages 321–327, 1988.

[67] R.S. Stephens. A Probabilistic Approach to the Hough Transform. In *Proceedings of the British Machine Vision Conference*, pages 55–60, Univ. of Oxford, September 1990.

[68] R.S. Stephens. *The Hough Transform: A Probabilistic Approach.* PhD thesis, Cambridge University, Department of Engineering, 1990.

[69] R.W. Taylor and A.P. Reeves. Classification Quality Assessment for a Generalized Model-Based Object Identification System. *IEEE Transactions on Systems, Man, and Cybernetics*, 19:861–866, 1989.

[70] D.W. Thompson and J.L. Mundy. Three Dimensional Model Matching From an Unconstrained Viewpoint. In *Proceedings IEEE International Conference on Robotics and Automation*, pages 208 – 219. IEEE, 1987.

[71] S. Ullman and R. Basri. Recognition by Linear Combinations of Models. A.I. Memo 1152, Massachusetts Institute of Technology, August 1989.

[72] H.L. Van Trees. *Detection, Estimation, and Modulation Theory, part 1.* John Wiley and Sons, 1968.

[73] G.H. Wannier. *Elements of Solid State Theory.* Cambridge University Press, 1959.

[74] W.M. Wells III. A Statistical Approach to Model Matching. In *Symposium on Advances in Intelligent Systems.* SPIE, 1990.

[75] W.M. Wells III. MAP Model Matching. In *Proceedings of the Computer Society Conference on Computer Vision and Pattern Recognition*, pages 486–492, Lahaina, Maui, Hawaii, June 1991. IEEE.

[76] W.M. Wells III. Posterior Marginal Pose Estimation. In *Proceedings: Image Understanding Workshop*, pages 745 – 751. Morgan Kaufmann, January 1992.

[77] C.F.J. Wu. On the Convergence Properties of the EM Algorighm. *The Annals of Statistics*, 11(1):95 – 103, 1983.

[78] A. Yuille, D. Geiger, and H. Bülthoff. Stereo Integration, Mean Field Theory and Psychophysics. In *Computer Vision - ECCV 90*, pages 73–82. Springer Verlag, 1990.

[79] A.L. Yuille. Generalized Deformable Models, Statistical Physics, and Matching Problems. *Neural Computation*, 2:1 - 24, 1990.