MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY

A.I. Memo No. 1463 February, 1994

Object Recognition by Alignment using Invariant Projections of Planar Surfaces

Kenji NAGAO W. Eric. L. GRIMSON

This publication can be retrieved by anonymous ftp to publications.ai.mit.edu.
The pathname for this publication is: ai-publications/1994/AIM-1463.ps.Z

Abstract

In order to recognize an object in an image, we must determine the best-fit transformation which maps an object model into the image. In this paper, we first show that for features from coplanar surfaces which undergo linear transformations in space, there exists a class of transformations that yield projections invariant to the surface motions up to rotations in the image field. To use this property, we propose a new alignment approach to object recognition based on centroid alignment of corresponding feature groups built on these invariant projections of planar surfaces. This method uses only a single pair of 2D model and data pictures. Experimental results show that the proposed method can tolerate considerable errors in extracting features from images and can tolerate perturbations from coplanarity, as well as cases involving occlusions. As part of the method, we also present an operator for finding planar surfaces of an object using two model views and show its effectiveness by empirical results.

This report describes research done at the Artificial Intelligence Laboratory of the Massachusetts Institute of Technology. Support for the laboratory's artificial intelligence research is provided in part by the Advanced Research Projects Agency of the Department of Defense under Office of Naval Research contract N00014-91-J-4038. WELG was also supported by NSF grant IRI-8900267.

1 Introduction

A central problem in object recognition is finding the best transformation that maps an object model into the image data. Alignment approaches to object recognition [6] find this transformation by first searching over possible matches between image and model features, but only until sufficiently many matches are found to explicitly solve for the transformation. Given such an hypothesized transformation, it is applied directly to the other model features to align them with the image. Each such hypothesis can then be verified by search near each aligned model feature for supporting or refuting evidence in the image.

One of the advantages of Alignment approaches to recognition [6] is that they are guaranteed to have a worst case polynomial complexity. This is an improvement, for example, over correspondence space search methods such as Interpretation Trees [5], which in general can have an exponential expected case complexity. At the same time, the worst case complexity for alignment can still be expensive in practical terms. For example, to recognize an object with m features from an image with n features, where the projection model is weak perspective, we must search on the order of m^3n^3 possible correspondences [6], where m and n can easily be on the order of several hundred. One way to control this cost is to replace simple local features (such as vertices) used for defining the alignment with larger groups (thereby effectively reducing the size of m and n). In this paper, we examine one such method, by showing that for features from planar surfaces which undergo linear transformations in space, there exists a class of transformations that yield projections invariant to the surface motions up to rotations in the image field.

This allows us to derive a new alignment approach to object recognition based on centroid alignment of corresponding feature groups built on these invariant projections of the planar surface. This method uses only a single pair of 2D model and data pictures, and is quite fast; in our testing, it took no more than 15 msec (0.015sec) per sample model and data pair, each with 50 features.

As part of the method, we also present an operator for finding planar surfaces of an object using two model views and show its effectiveness by empirical results.

2 Problem definition

Our problem is to recognize an object which has planar portions on its surface, using a single pairing of 2D model and data views as features. Thus, we assume that at least one corresponding region (which is from a planar surface of the object) including a sufficient number of features exists in both the model and data 2D views. Although we do not explicitly address the issue of extracting such regions from the data, we note that several techniques exist for accomplishing this, including the use of color and texture cues [12, 14], as well as motion cues(e.g.[15, 10]). We devise a method for finding an alignment between features of these planar regions. It is important to stress that our method is not restricted to 2D objects. Rather it assumes that objects have planar sections, and that we are provided with 2D views of the object model that include such planar sections. Once we have solved with the transformation between model and image, we can apply it to all the features on a 3D object, either by using a full 3D model [6] or by using the Linear Combinations method on 2D views of the object [16].

The basis for our method is the consistency of an object's structure under some simple transformations. To see how this works, we first summarize the derivation of the constraint equation of the 2D affine transformations which describe the motion of the object in space (see, e.g.[11, 8]).

Let O, P_1, P_2, P_3 be four non-coplanar points on an object. Then, any point on the object can be represented by the vector sum:

$$OP = \sum_{i=1}^{3} \alpha_i \, OP_i \tag{1}$$

where the α_i's are real coefficients. When the object undergoes a linear transformation caused by its motion in space, this equation will be transformed as

$$O'P' = \sum_{i=1}^{3} \alpha_i \, O'P'_i \tag{2}$$

where the primes indicate the position of the features after the motion. Taking the orthographic projections of these points to the xy image plane yields

$$op = \sum_{i=1}^{3} \alpha_i op_i \tag{3}$$

$$o'p' = \sum_{i=1}^{3} \alpha_i o'p'_i \tag{4}$$

Since the op_i's and $o'p'_i$'s are independent of one another, there exists a unique 2D affine transformation L, ω, such that,

$$o'p'_i = Lop_i + w \tag{5}$$

where L is a 2×2 matrix and ω is a 2D vector. Then, combining (3), (4) and (5), for an arbitrary point we get,

$$o'p' = Lop + \omega + (\sum_{i=1}^{3} \alpha_i - 1)\omega \tag{6}$$

Hence, as a constraint equation for the motion of a plane, we obtain the well known result:

$$o'p' = Lop + \omega \tag{7}$$

Thus, the new position of any point (after the motion) is described by an affine transformation, and that transformation can be found by matching a small number of points across images. The direct use of 2D affine transformations in object recognition was made earlier by Huttenlocher[6]. The issue in which we are interested is whether there are properties of the affine transformation which we can use to efficiently and reliably find the parameters of that transformation.

3 A class of 2D projections of planar surfaces invariant to linear transformations

In this section, we show a class of transformations of 2D image features from planar surfaces which yield a unique projection up to rotations in the image field, regardless of the pose of the surface in space. First, the following useful observation is made.

[Definition]
Let H be a positive definite symmetric matrix, expressed as

$$H = U^T \Lambda U$$

where U is an orthogonal matrix and Λ is an eigenvalue matrix of H, specifically,

$$\Lambda = diag(\lambda_1, \lambda_2)$$

where λ_i's are the eigenvalues of H which are all positive.

The square root matrix $H^{\frac{1}{2}}$ of the matrix H is defined by,

$$H^{\frac{1}{2}} = U^T \Lambda^{\frac{1}{2}} U$$

where

$$\Lambda^{\frac{1}{2}} = diag(\lambda_1^{\frac{1}{2}}, \lambda_2^{\frac{1}{2}}) \qquad (8)$$

It is known that the positive definite symmetric square root matrix of a positive definite symmetric matrix is unique[7].
□

[Definition]
The covariance matrix of a feature distribution of vectors $\{X_i\}$ with a mean vector M and a probability density function $P(X)$ is given by,

$$\Sigma_X = \sum_{i=1}^{N} P(X_i)(X_i - M)(X_i - M)^T$$

where N is the number of features.
□

[Proposition 1]
Let X be a model feature position and X' be the corresponding data feature position. We can relate these by

$$X' = LX + \omega \qquad (9)$$

Now suppose both features are subjected to similar transformations

$$\begin{aligned} Y &= AX + B & (10)\\ Y' &= A'X' + B' & (11)\\ Y' &= TY + C & (12) \end{aligned}$$

Then a necessary and sufficient condition for these transformations to commute (i.e. to arrive at the same values for Y') for all X, X' is that (see Figure 1)

$$H^{\prime\frac{1}{2}} U H^{-\frac{1}{2}} = T \qquad (13)$$

for some orthogonal matrix U, where $H^{\frac{1}{2}}$ and $H^{\prime\frac{1}{2}}$ are square root matrices of H and H' respectively, and

$$\begin{aligned} H' &= A'\Sigma_{X'} A'^T & (14)\\ H &= A \Sigma_X A^T & (15) \end{aligned}$$

where Σ_X and $\Sigma_{X'}$ represent the covariance matrices of X and X' respectively.

$$\begin{tikzcd} X \arrow[r, "AX + B"] \arrow[d, "LX + w"'] & Y \arrow[d, "TY + C"] \\ X' \arrow[r, "A'X' + B'"'] & Y' \end{tikzcd}$$

Figure 1: Commutative Diagram of Transformations
Given model feature X and corresponding data feature X', we seek conditions on the transformations A, A' such that this diagram commutes.

Proof:
First, we show the necessity of the condition (13).
Substituting (9) to (11) into (12), we have,

$$(A'L - TA)X + A'\omega + B' - TB - C = 0. \qquad (16)$$

Since this must hold for any X, we have

$$A'L = TA. \qquad (17)$$

Applying (9) to the covariances of X' and X, we have

$$\Sigma_{X'} = L \Sigma_X L^T. \qquad (18)$$

Substituting (18) into (14) yields

$$A' L \Sigma_X L^T A'^T = H'. \qquad (19)$$

On the other hand from (15) we have

$$\Sigma_X = A^{-1} H (A^T)^{-1} = A^{-1} H (A^{-1})^T. \qquad (20)$$

Then, substituting (20) into (19) yields

$$(A'LA^{-1}) H (A'LA^{-1})^T = H'. \qquad (21)$$

Since H and H' are positive definite symmetric matrices, (21) can be rewritten as

$$(A'LA^{-1}H^{\frac{1}{2}})(A'LA^{-1}H^{\frac{1}{2}})^T = H^{\prime\frac{1}{2}}(H^{\prime\frac{1}{2}})^T \qquad (22)$$

where $H^{\frac{1}{2}}$, $H^{\prime\frac{1}{2}}$ are again positive definite symmetric matrices.
Then, from (22)

$$A'LA^{-1}H^{\frac{1}{2}} = H^{\prime\frac{1}{2}} U. \qquad (23)$$

Thus, we get

$$A'LA^{-1} = H^{\prime\frac{1}{2}} U H^{-\frac{1}{2}} \qquad (24)$$

where U is an orthogonal matrix.
Then, combining (17) and (24) finally we reach (13). Clearly, (13) is also a sufficient condition.
□

Note that this property is useful because it lets us relate properties of object and data together. In particular, if the projection of the object into the image can be approximated as a weak perspective projection, then we know that this defines a unique affine transformation of the planar object surface into the image[6]. The proposition gives us strong conditions on the relationship between linear transformations of the object, and the induced transformation of its projection into the image.

Now, if we limit T to orthogonal transformations, the following proposition holds.

[Proposition 2]
A necessary and sufficient condition that T in (13) is an orthogonal matrix for any U is

$$H' = H = c^2 I \tag{25}$$

where I is the identity matrix and c is an arbitrary scalar constant.

Proof:
Using the assumption that T is an orthogonal matrix, from (13), we have

$$\begin{aligned} I &= TT^T && (26)\\ &= \{H^{\frac{\prime 1}{2}} U H^{-\frac{1}{2}}\}\{H^{\frac{\prime 1}{2}} U H^{-\frac{1}{2}}\}^T && (27)\\ &= H^{\frac{\prime 1}{2}} U H^{-1} U^T H^{\frac{\prime 1}{2}}. && (28)\end{aligned}$$

Rearranging this, we get

$$U^T H' = H U^T \tag{29}$$

In order for any orthogonal matrix U to satisfy (29), as H and H' are positive definite,

$$H = H' = c^2 I \tag{30}$$

where c is an arbitrary scalar constant.
□

It should be noted that it is not possible that T in (13) is the identity matrix for any U. Thus, we are not allowed to align each model and data feature by just setting H and H' to some matrices, and solving for A and A'. This is because the distributions have been normalized, so that their second moments are already useless for determining the orientations of the distributions.

Proposition 2 allows us to provide the following useful proposition.

[Proposition 3]
Any solution for A and A' in (25), that is,

$$A' \Sigma_{X'} A'^T = A \Sigma_X A^T = c^2 I$$

can be expressed as

$$\begin{aligned} A &= cU\Lambda^{-\frac{1}{2}}\Phi^T && (31)\\ A' &= cU'\Lambda'^{-\frac{1}{2}}\Phi'^T && (32)\end{aligned}$$

where Φ and Φ' are eigenvector matrices and Λ and Λ' are eigenvalue matrices of the covariance matrices of X and X' respectively, U and U' are arbitrary orthogonal matrices, and c is an arbitrary scalar constant.

Proof:
Clearly,

$$\begin{aligned} \tilde{A} &= c\Lambda^{-\frac{1}{2}}\Phi^T && (33)\\ \tilde{A}' &= c\Lambda'^{-\frac{1}{2}}\Phi'^T && (34)\end{aligned}$$

are solutions for (25).
Let an arbitrary solution A of (25) be expressed as $A = U\tilde{A}$. Then,

$$\begin{aligned} A\Sigma_X A^T &= U\tilde{A}\Sigma_X \tilde{A}^T U^T && (35)\\ &= c^2 U U^T && (36)\\ &= c^2 I && (37)\end{aligned}$$

Therefore, A can be expressed as

$$A = U\tilde{A} \tag{38}$$

where U is an arbitrary orthogonal matrix and c is an arbitrary scalar constant.
In the same way,

$$A' = U'\tilde{A}' \tag{39}$$

where U' is an arbitrary orthogonal matrix.
□

By combining Proposition 2 and the following two properties, we can derive the major claim of this section.

[Lemma 1]
When U is an orthogonal matrix,
U is a rotation matrix $\Longleftrightarrow$ $det[U] > 0$
U is a reflection matrix $\Longleftrightarrow$ $det[U] < 0$

Proof :
When U is an orthogonal matrix, U can be expressed as

$$U = \begin{cases} \begin{pmatrix} c & -s \\ s & c \end{pmatrix} & \text{when U is a rotation matrix} \\ \begin{pmatrix} c & s \\ s & -c \end{pmatrix} & \text{when U is a reflection matrix} \end{cases} \tag{40}$$

where $c^2 + s^2 = 1$. Hence, the lemma is proved.
□
[Lemma 2]
When a planar surface is still visible after the motion in space, $det[L] > 0$.

Proof:
As is well known, any plane can be made parallel to the xy image plane by rotations around the x and y axes. The effect of these rotations in the xy plane can be expressed by a shear S and a subsequent dilation D. Specifically,

$$S = \begin{pmatrix} 1 & 0 \\ \alpha & \beta \end{pmatrix} \tag{41}$$

$$D = \begin{pmatrix} \gamma & 0 \\ 0 & 1 \end{pmatrix} \tag{42}$$

When this motion of the plane takes place so that it is always visible, clearly $\alpha > 0$, $\beta > 0$, $\gamma > 0$. Thus, we have $det[DS] > 0$. When we do this operation to the object planar surface both at the pose for the model and the data by respectively DS and $D'S'$, it is easy to see that the following relation holds,

$$RDS = D'S'L \tag{43}$$

for some rotation matrix R.
Then, from lemma 1 we get,

$$det[L] = det[S'^{-1}D'^{-1}RDS] > 0 \tag{44}$$

□

Finally, the following constructive property allows the claims presented above to become the basis of a practical tool for recognizing planar surfaces.

[Theorem 1]
When (9) represents the motion of a plane, and the transformation for model and data are respectively (33) and (34) such that both Φ and Φ' represent rotations/reflections, then T in (12) is a rotation matrix.

Proof:
From proposition 1,

$$A'L = TA \tag{45}$$

where A and A' are chosen as in (33) and (34) such that both Φ and Φ' represent rotations/reflections.
Then, from lemma 1 and 2, we have

$$det[T] = det[A'LA^{-1}] > 0. \tag{46}$$

□

What does this imply? If we have a set of model features and data features related by an affine transformation (either due to a weak perspective projection of the object into the image, or due to a linear motion of the object image between two image frames), then if we transform both sets of features linearly in a well defined way (via (33) and (34)), we derive two distributions of features that are identical up to a rotation in the image field. This implies that the transformed distributions are unique up to their shapes. More importantly, it also provides an easy method for finding the related transformation.

A physical explanation of this property is given using Figure 2 as follows. Suppose the upper pictures show the surfaces in space at the model and the data poses as well as the respective orthographic projections. Looking at the major and minor axes of the 2D model and the data, we can change the pose of the planes so that the major and minor axes have the same length in both the model and data, as depicted in the lower pictures. This is nothing but a normalization of the feature distributions, and the normalized distributions are unique up to a rotation, regardless of the pose of the plane, i.e., no matter whether it is from the pose for the model or for the data.

An example of applying the proposed transformation is shown in Figure 3.

4 Alignment using a single 2D model view

In this section, we show how we can align the 2D model view of the planar surface with its 2D images using the tool derived in the last section.

4.1 Using the centroid of corresponding feature groups

If the model and data features can be extracted with no errors, and if the surface is completely planar, then applying the presented transformation to model and data features will yield new feature sets with identical shapes (up to an image plane rotation). Thus, in this case, our problem, i.e., recovering the affine parameters which generated the data from the model is quite straightforward. One way to do this is simply to take the most distant features from the centroid of the distribution both in the model and data, and then to do an alignment by rotating the model to yield a complete coincidence between each model and data feature. Then, we can compute the affine parameters which result in that correspondence.

However, the real world is not so cooperative. Errors will probably be introduced in extracting features from the raw image data, and, in general, the object surfaces may not be as planar as we expect. To overcome these complications, we propose a robust alignment algorithm that makes use of the correspondences of the centroid of corresponding feature groups in the model and data.

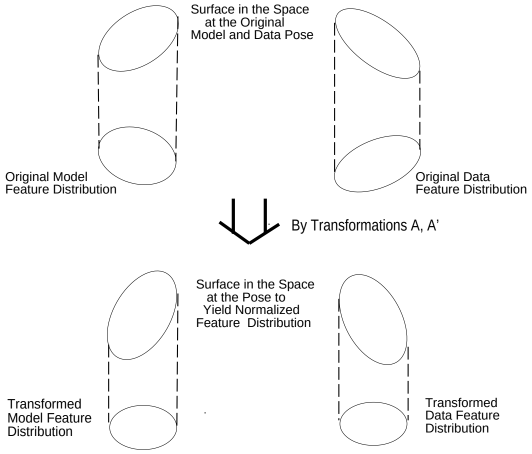

Figure 2: Physical explanation of the Invariant Projection

The upper pictures show the surfaces in space at the model and the data poses, as well as their orthographic projections to the image field. The lower pictures show the surfaces and their projections at the poses yielding normalized distributions.

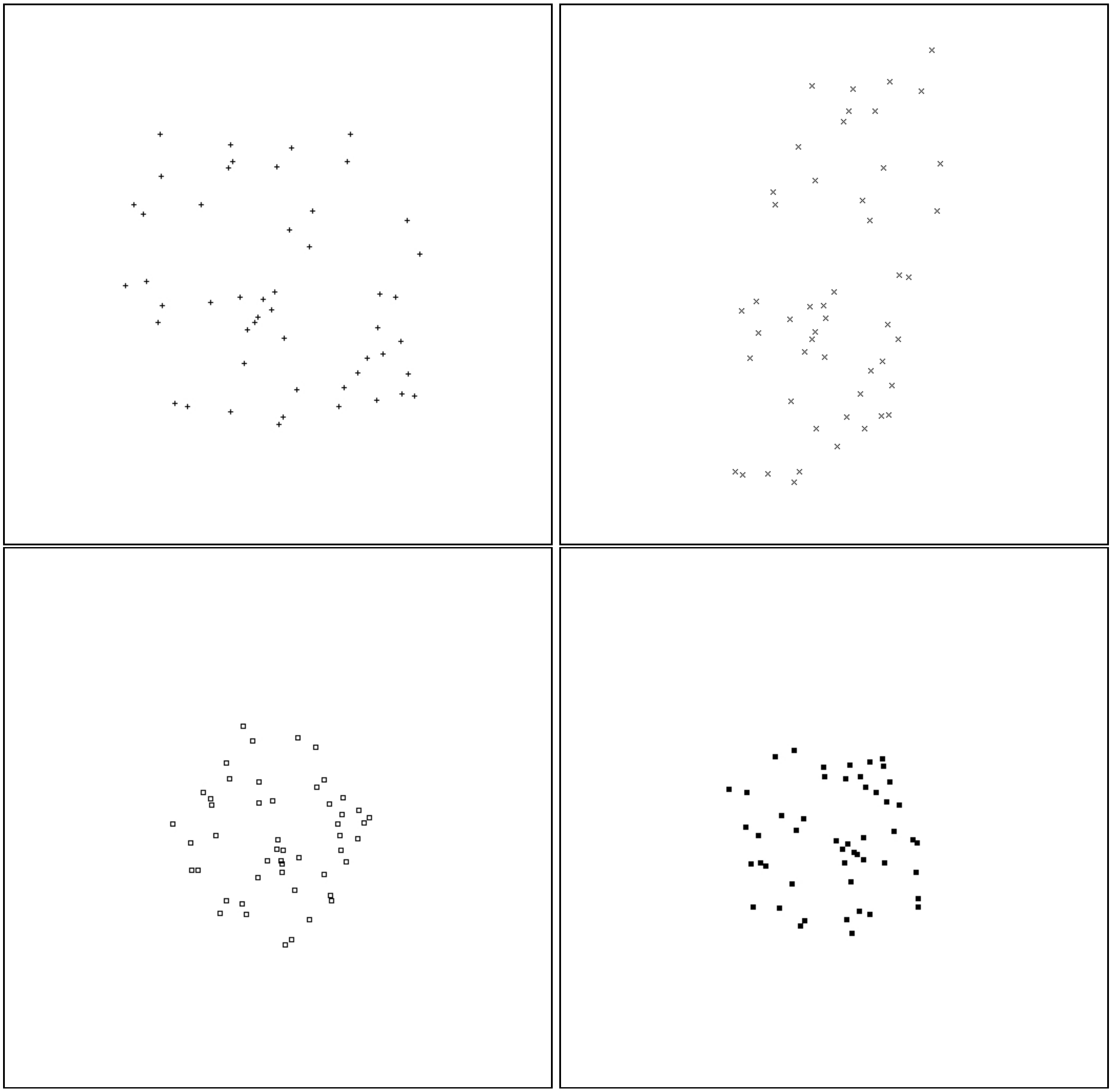

Figure 3: An Example of the Application of Invariant Projection

Upper left: the original model features, Upper right: the original data features, Lower left: transformed model features, Lower right: transformed data features. Transformed features from the model and the data have the same distribution up to a rotation in the image field.

Here we see an important property hold:

[Theorem 2]
When the motion of the object in space is limited to linear transformations, the centroid of its orthographic projection to a 2D image field, i.e., centroids of image feature positions, is transformed by the same transformation as that by which each image feature is transformed.

Proof:
When any point X_i on the object surface in space is transformed to X'_i by a 3D linear transformation $\mathcal{T}$, its orthographic projection x_i is transformed to x'_i by

$$x'_i = \Pi\,\mathcal{T}\,\Pi^{-1} x_i \qquad \text{for} \quad i = 1 \text{ to } N \tag{47}$$

where N is the number of points, Π represents the orthographic projection of object points, and Π^{-1} is the lifting operation. Specifically,

$$x_i = \Pi X_i \tag{48}$$
$$x'_i = \Pi X'_i \tag{49}$$

This is also true for any of the linear combinations of these points, because

$$\sum_{i=1}^{N} \alpha_i\, x'_i = \sum_{i=1}^{N} \alpha_i\, \Pi X'_i \tag{50}$$
$$= \sum_{i=1}^{N} \alpha_i\, \Pi\,\mathcal{T} X_i \tag{51}$$
$$= \sum_{i=1}^{N} \Pi\,\mathcal{T}\,\Pi^{-1} \alpha_i\, x_i \tag{52}$$
$$= \Pi\mathcal{T}\Pi^{-1}\Big(\sum_{i=1}^{N} \alpha_i\, x_i\,\Big) \tag{53}$$

where $\alpha_i{}'s$ are arbitrary real coefficient. Thus, the proposition is proved.
□

Moreover, we see that the following reliable property holds.

[Proposition 4]
When the errors in extracting features and/or the perturbation of their depth from coplanarity is zero-mean, the centroid is transformed by the same transformation, although each feature point is no longer guaranteed to be aligned by the same transformation.

The proof is straightforward, and is not given here. Note that these properties are generally true for any object surface and its motions. The coplanarity of the surface does not matter. In the case when the object happens to be planar, as the motion of the 2D image feature is described by an affine transformation, the centroid of the features is also transformed by the same affine transformation.

In [13], the use of region centroids was proposed in the recognition of planar surfaces. Unlike our approach for using feature group centroids, however, their method can only be applied to planar objects, as described in the paper.

4.2 Grouping by clustering of features

Since affine parameters can be determined from three point correspondences, our problem becomes one of obtaining three corresponding positions in model and data, in the presence of perturbations. Based on the observations made in the preceding sections, we propose to group the model and data features using their transformed coordinates, so that we can extract a single feature from each of a small number of groups. The goal is to use such groups to drastically reduce the complexity of alignment based approaches to recognition, by finding groups whose structure is reproducible in both the model and the data, and then only match distinctive features of such groups.

One way to group features is to employ clustering techniques. In the selection of clustering algorithm from the many choices, taking into account the use of the property we have derived in the last section, that is, the transformed model and data features are unique up to rotations and translations, we set the following two criteria: (a) invariance of the clustering criterion to rotations and translations of the x, y coordinate system, (b) low computational cost. The criterion (b) is also critical, because if the computational cost of clustering is similar to those of conventional feature correspondence approaches, the merit of our method will be greatly decreased.

We have opted to use Fukunaga's version of ISODATA algorithms [3, 4, 9] for the following reason. The criterion of this algorithm is to minimize the intraclass covariances of the normalized feature distribution instead of the original distribution. Specifically, let the criterion be:

$$J = trace[K_w] \tag{54}$$

where

$$K_w = \Sigma_{i=1}^{M} Q(\omega_i) K_i \tag{55}$$

where $Q(\omega_i)$ is the probability density function of the ith cluster, M is the number of clusters, and K_i is the intragroup covariance of the ith cluster for the normalized feature set. The normalization of an original features is performed using the same transformation as that presented in the last section. Therefore, applying ISODATA on our transformed coordinates is equivalent to adopting Fukunaga's method. It is clear that the criterion given in (54) is invariant to the rotation and translation of the x, y coordinate system.

Moreover, since the ISODATA algorithm, starting from the initial clustering, proceeds like a steepest descent method for ordered data, it is computationally very fast. It runs in $O(N)$ time in terms of the number of the features N to be classified, when we set the upper limit to the number of iteration as is often done. We should also note that, although it is not guaranteed that it can ever reach the real minimum of J, we know that our aim is not to minimize/maximize some criterion exactly, but to yield the same cluster configuration both in model and data clustering. Minimization of a criterion is nothing more than one attempt to this.

4.3 Aligning a model view with the data

Now we can describe an algorithm for aligning a 2D view of a model with its novel view, which is assumed to be nearly planar. Note that, however, to determine the best affine transformation, finally we must examine all the feature groups isolated from the data, as we do not know which group in the data actually corresponds to the planar surface which has been found in the model.

- Step 0: For a feature set from a 2D view of a model, compute the matrices given in (33) where U may be set to I and generate the normalized distribution. Cluster based on ISODATA to yield at least three clusters. Compute the centroid of each cluster reproduced in the original coordinate. This process can be done off-line.
- Step 1: Given a 2D image data feature set, do the same thing as step 0 for the data features.
- Step 2: Compute the affine transformation for each of the possible combinations of triples of the cluster centroids in model and data.
- Step 3: Do the alignment on the original coordinates and select the best-fit affine transformation.

Step 1 is clearly $O(N)$. In Step 2, computation of affine parameters must be done for only a small number of combinations of clusters of model and data features. So, it runs in constant time. Step 3 is, like all other alignment approaches, of the order of the image size. Thus, this alignment algorithm is computationally an improvement over the conventional ones for object recognition.

We stress again that our method is not restricted to planar objects. We simply require a planar surface on an object to extract the alignment transformation. This transform can then be applied to a full 3D model or used as part of a Linear Combinations approach to sets of views of a 3D model to execute 3D recognition.

5 Finding planar portions on the object surface using two 2D model views

In this section, we derive an operator for detecting the planar portions on the object surface without the direct use of depth information. This operator uses two 2D model views with a sufficient number of correspondences between features. The basic underlying idea in its derivation is the same as those used for motion/accretion region detection [1, 10], and for smooth/singular segment detection along a curve [2].

5.1 Evaluating the planarity of a surface

Suppose that we have the correspondences between model feature set $\{X\}$ and data feature set $\{X'\}$. From the expansion of (7) to x, y components, we have

$$\hat{x'} = L_{11}\hat{x} + L_{12}\hat{y} \qquad (56)$$

$$\hat{y'} = L_{21}\hat{x} + L_{22}\hat{y} \qquad (57)$$

where $\hat{a} = a - \bar{a}$, and $(\bar{x'}, \bar{y'})^T$ and $(\bar{x}, \bar{y})^T$ are the respective mean vectors of the model and data feature distributions. Clearly, the existence of $L_{ij}{}'s$ which satisfy (56) and (57) is the necessary and sufficient condition that the feature set is distributed coplanarly.

Let the covariance matrices of $U = (x', x, y)$ and $V = (y', x, y)$ respectively be C_U and C_V. Then, we see that the following lemma holds.

[Lemma 3]

$$det[C_U] = 0 \iff \hat{x'} = L_{11}\hat{x} + L_{12}\hat{y} \qquad (58)$$
$$\text{for some real } (L_{11}, L_{12}) \neq (0,0)$$

$$det[C_V] = 0 \iff \hat{y'} = L_{21}\hat{x} + L_{22}\hat{y} \qquad (59)$$
$$\text{for some real } (L_{21}, L_{22}) \neq (0,0)$$

This is basically the same result as that presented by Ando[1]. A proof is given in the Appendix. By using this property, we can evaluate to what extent a feature set is distributed coplanarly in space, without estimating the best-fit affine parameters L_{ij}, by some method, say, least square errors. In the following part, we concentrate the discussion on (58). The same argument holds for (59).

In [10], claims were made for the necessity of normalization of the measure. We support that argument here, because clearly $det[C_U]$ depends on the resolution of the image, so we can not use $det[C_U]$ directly to evaluate the coplanarity. In addition, in order to remove the effect of linearity of the (x, y) distribution itself from $det[C_U]$, we transform U to yield a normalized distribution.

$$\check{U} = AU \qquad (60)$$

where,

$$A = \begin{pmatrix} \Lambda^{-\frac{1}{2}}\Phi^T & 0 \\ 0 & \sigma^{-\frac{1}{2}} \end{pmatrix} \qquad (61)$$

where Λ and Φ are respectively eigenvalue and eigenvector matrices of the covariance matrix of (x, y), and σ is the variance of x'.

Let $\check{C_U}$ be the covariance matrix of $\check{U}$. Then, guided by the Schwarz Inequality for the eigenvalues α, β, γ of $\check{C_U}$, which are all positive, we get a normalized measure

$$1 - \frac{\alpha\beta\gamma}{(\frac{\alpha+\beta+\gamma}{3})^3} = \frac{det[C^1]C_{x'x'} - det[C_U]}{det[C^1]C_{x'x'}} \qquad (62)$$

where C^1 is the covariance matrix of (x, y) and $C_{x'x'}$ is the variance of x'.

Note that, since $det[C_U] = \alpha\beta\gamma$ indicates the square of the volume of the distribution of U, the numerator of (62) reflects the relation in (56), while the denominator has no direct connection to it.

In the same way, for V we get,

$$\frac{det[C^1]C_{y'y'} - det[C_V]}{det[C^1]C_{y'y'}} \qquad (63)$$

where $C_{y'y'}$ is the variance of y'.

Then, combining these two, finally we get an operator P

$$P = \frac{det[C^1](C_{x'x'} + C_{y'y'}) - (det[C_U] + det[C_V])}{det[C^1](C_{x'x'} + C_{y'y'})} \qquad (64)$$

Note that P is a normalized measure which is free from any physical dimensions, with the following important property that is easily shown by a simple calculation.

[Lemma 4]
P is invariant to rotations and translations in the xy image plane.

5.2 Using the operator in detecting planar surfaces

When we set the tolerable perturbation of the surface at the rate $P \geq r$, then we can introduce a coefficient to adjust the measure P so that it ranges from 1 down to 0 within the range $P \geq r$. This is done by choosing the scalar coefficient k such that,

$$det[\tilde{C}^1](\tilde{C}_{x'x'} + \tilde{C}_{y'y'}) - k \cdot (det[\tilde{C}_U] + det[\tilde{C}_V]) = 0 \tag{65}$$

where $E\{P(\tilde{C_U}, \tilde{C_V})\} = r$, $E\{\cdot\}$ denotes an expectation of the P obtained through experimental results. Thus, we have

$$P(k) = \frac{det[C^1](C_{x'x'} + C_{y'y'}) - k \cdot (det[C_U] + det[C_V])}{det[C^1](C_{x'x'} + C_{y'y'})} \tag{66}$$

So, we have derived a pseudo-normalized measure for the specific range of surface coplanarity with which we are concerned. It is easy to see that $P(k)$ is again invariant to rotations and translations in the xy image plane.

5.3 Empirical results on the sensitivity of P

We show empirical results on the sensitivity of P to the perturbations of feature positions caused by their depth perturbations in space. Examinations were performed on two sets of model features produced by canonical statistical methods. First, a set of model features were generated randomly. Then, generating random affine parameters, in our case L_{ij}, each model feature was transformed by this transformation to yield another model feature set. Finally, we added perturbations to the second set of features according to a Gaussian model. Since the effect of depth perturbations appears only in the direction of the translational component of the affine transformation, in proportion to the dislocation of the point from the plane[11], we added perturbations only in the direction of the x axis. Perturbations along other directions yielded similar results.

Figure 4 shows the values of the operator P versus the deviation of the Gaussian perturbation. The horizontal axis shows the Gaussian deviation and the vertical axis shows the value of the operator P. Twenty model pairs were used for each of the Gaussian perturbation, and 50 features were included in each model. In the Figure, the average value of P from the 20 pairs is plotted versus the Gaussian deviation. The value of the operator P decreases monotonically as the deviation increases.

6 Experimental results

In this section, experimental results show the effectiveness of the proposed algorithm for recognizing planar surfaces.

As in the last section, we used random patterns for model features, random values for affine parameters, and additive Gaussian perturbations to simulate the feature extraction errors and the depth perturbations of the object surface in space from planarity. We also simulate the case including occlusions.

[Algorithm Implementation]
In order to obtain three clusters in model and data, we adopted a hierarchical application of ISODATA. This is because through some tests of ISODATA, we learned that the accuracies for generating three clusters severely declined from those for generating two clusters. Therefore, the actual method we took for feature clustering was: (1) first do clustering on the original complete feature set to yield two clusters for model and data, (2) then, do clustering again for each of the clusters generated in the first clustering to yield two subclusters from each cluster. To find the best affine parameters L_{ij}, all the possible combinations of the centroid correspondences between model and data clusters and subclusters were examined. Initial clusters were produced by selecting the initial separating line as the one that passes through the centroid of the distributions to be classified and is perpendicular to the line passing through the centroid and the most distant feature position from the centroid.

In Figure 5, intermediate results of the hierarchical procedures described above are shown.

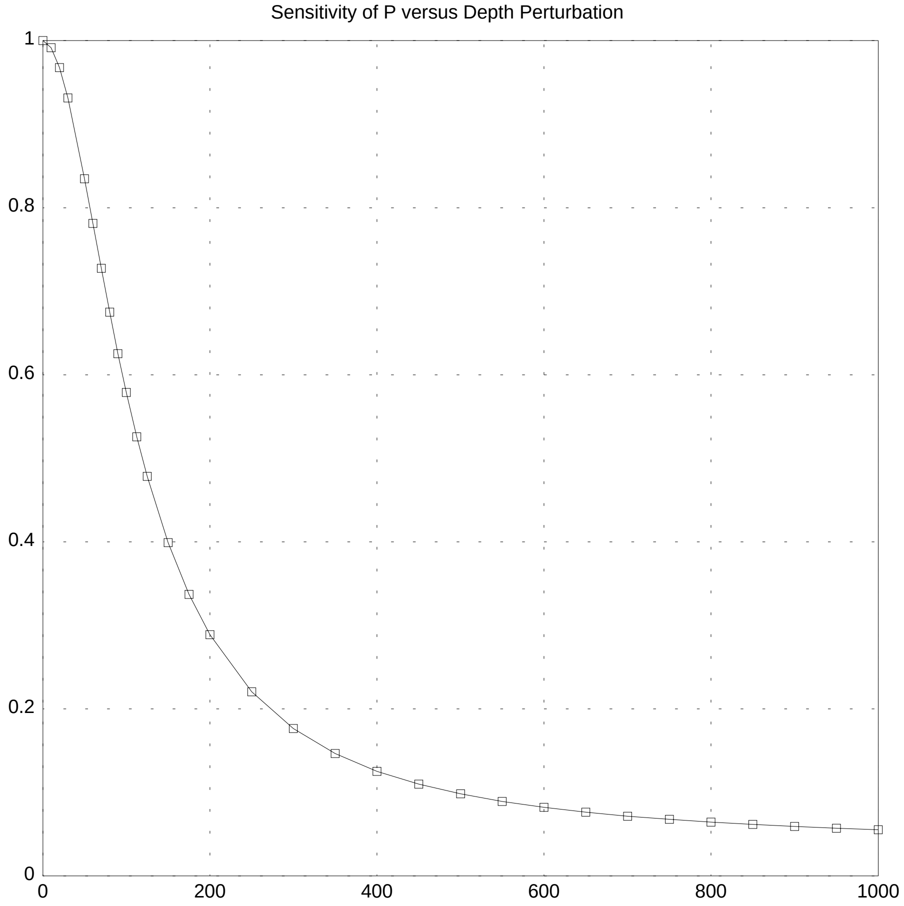

Figure 4: Sensitivity of the operator P to perturbations of the depth from planarity in space.
The values of the operator P are plotted versus the Gaussian deviations of the perturbations in data feature. The horizontal axis shows the Gaussian deviation and the vertical axis shows the value of the operator P. Twenty model pairs were used for each of the Gaussian perturbations, and 50 features were included in each model.

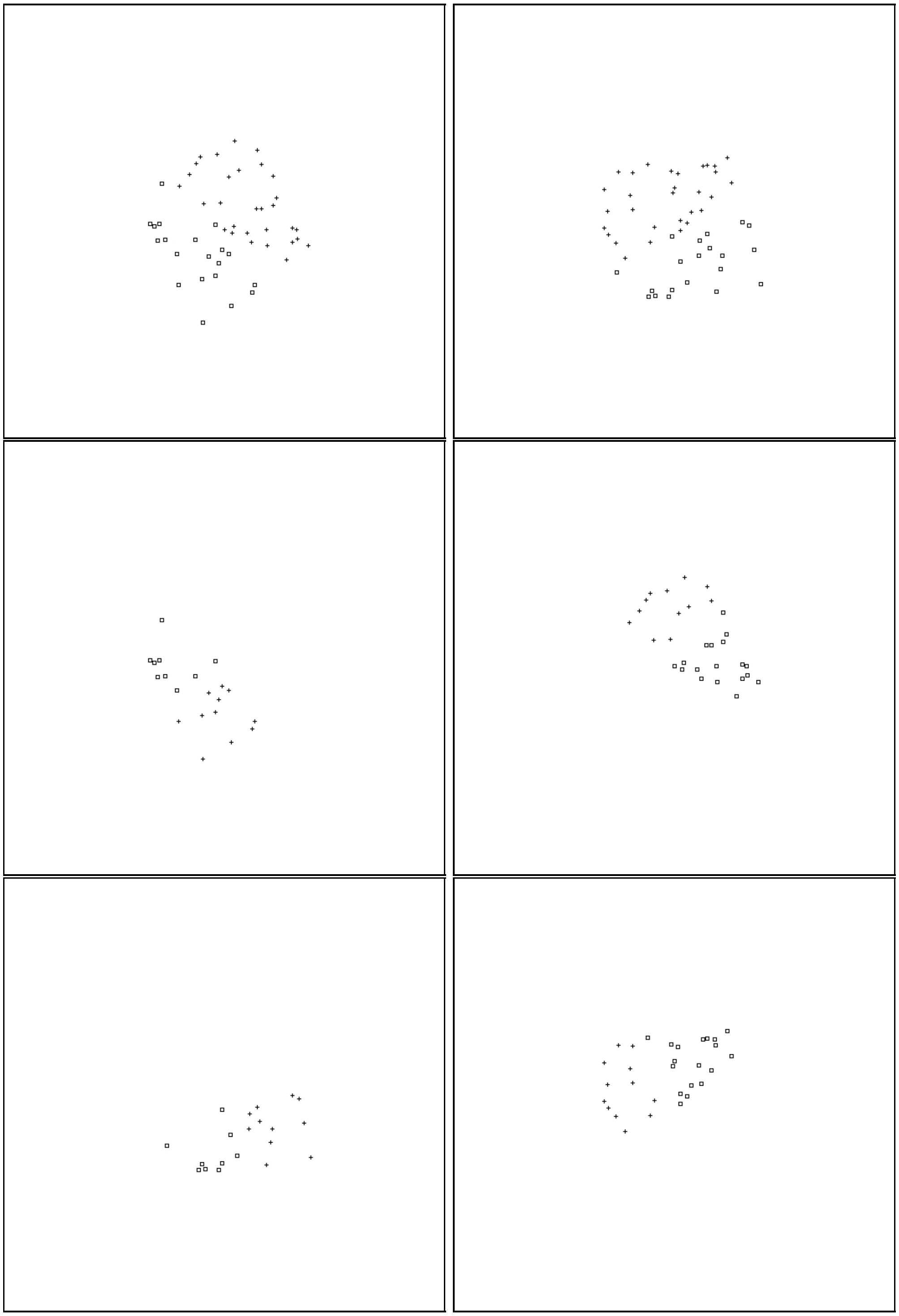

Figure 5: An example of hierarchical clustering.
Upper left: results of the first clustering of the transformed model features, Upper right: results of the first clustering of the transformed data features, Middle: subclusters yielded by the second clustering of the first clustering results of the model, Lower: subclusters yielded by the second clustering of the first clustering results of the data.

In each of the following experiment 100 sample model and data with 50 features were used, and the average of their results were taken.

[With errors in extracting features]
In Figure 6, errors in recovering the affine parameters L_{ij}, which are estimated by the following measure, are plotted versus the rate of the Gaussian deviation to the average distance between closest features of the data.

$$\text{error} = \sqrt{\frac{\Sigma_{i,j}(\hat{L}_{i,j} - L_{ij})^2}{\Sigma_{i,j} L_{ij}^2}} \qquad (67)$$

where $\hat{L_{ij}}$ is the recovered values for affine parameters. The average distance between closest feature points was estimated by

$$\text{average distance} = \sqrt{\frac{det[L]A}{\pi N}} \qquad (68)$$

where A is the area occupied by the model distribution, and N is the number of the features included. The perturbation rate used to generate Gaussian deviation were taken to be the same in both the x and y coordinates to simulate the errors in feature extraction. In Figure 6 we note that errors are almost proportional to the perturbation rate. In Figure 7, examples of the reconstructed data distributions, with different errors in recovering the affine parameters, were superimposed on the data with no perturbations. The average errors in recovering affine parameters increased, as perturbations in the data features grew larger. However, even in such cases, errors are still small for most samples as we can see in Table 1. In almost all cases when the recovering of L_{ij} results in large errors, the first clustering failed due to the change of the most distant features in model and data. The ratio of this kind of failure increased as the perturbation percentage grew. That is the reason for the error elevations in such samples. But, by combining properties other than positions of the features in giving initial clusters, such as colors, this will be considerably improved.

From Figures 6 and 7, our algorithm is found to be quite robust against considerable perturbations caused by the errors in feature extractions.

	5	10	15	20	25	30	35
— 0.01	73	52	30	21	7	3	0
0.01 - 0.05	12	17	27	31	36	37	31
0.05 - 0.1	8	10	14	16	15	14	14
0.1 - 0.2	2	3	5	4	8	10	11
0.2 - 0.3	2	2	3	6	7	5	7
0.3 - 0.4	0	2	3	5	3	5	5
0.4 —	3	14	18	17	24	26	32

Table 1: Number of Samples with Errors vs. Perturbation.
The number of the samples with errors out of 100 model and data pairs are shown versus perturbation rate. The first column shows the recovery errors, and the first row shows the perturbation percentages included in the data features.

	5	10	15	20	25
— 0.01	7	0	0	0	0
0.01 - 0.05	24	29	21	15	1
0.05 - 0.1	19	24	34	33	28
0.1 - 0.2	4	17	2	8	18
0.2 - 0.3	6	17	8	11	8
0.3 - 0.4	11	4	1	14	10
0.4 —	29	9	34	19	35

Table 2: Number of Samples with Errors vs. Occlusion. The number of the samples with errors out of 100 model and data pairs are shown versus the rate of missing features in the data. Each model has 50 features. The first column shows the recovery errors, and the first row shows the percentages of missing features.

[Depth perturbation from planarity]
In the same way, in Figure 8 estimation errors are shown to simulate the case where the surface has depth perturbations from planarity. As described previously, perturbations in the image field caused by depth variation occur in the direction of the translational component of the affine transformation. Therefore, the perturbation rate was taken only for the x coordinate. Similar results were obtained from other directions of perturbations.
From Figure 8, again, we can see that our algorithm is quite stable against perturbations caused by the depth variations of the points from planarity. Thus, our method can be used to obtain approximate affine parameters for object surfaces with small perturbations from planarity.

[With Occlusion]
In Figure 9, the errors in recovering affine parameters are plotted versus the rate of the number of the missing features in the data, which is to simulate the case including occlusions.

Roughly speaking, the errors increase as the missing features increase. The perturbations from the monotonous elevation of the errors are caused by the unstable initial clusterings. Actually, we note in Table 2 that even in the cases with high average errors, many of the samples result in a good recovery, while some result in large errors. This is because the accuracy of the initial clustering in our algorithm depends on how much the most distant feature from the centroid remain identical in model and data. So, when it changes critically due to the missing of features, it becomes unstable. However, again this can probably be fixed by combining other cues in obtaining initial clustering.

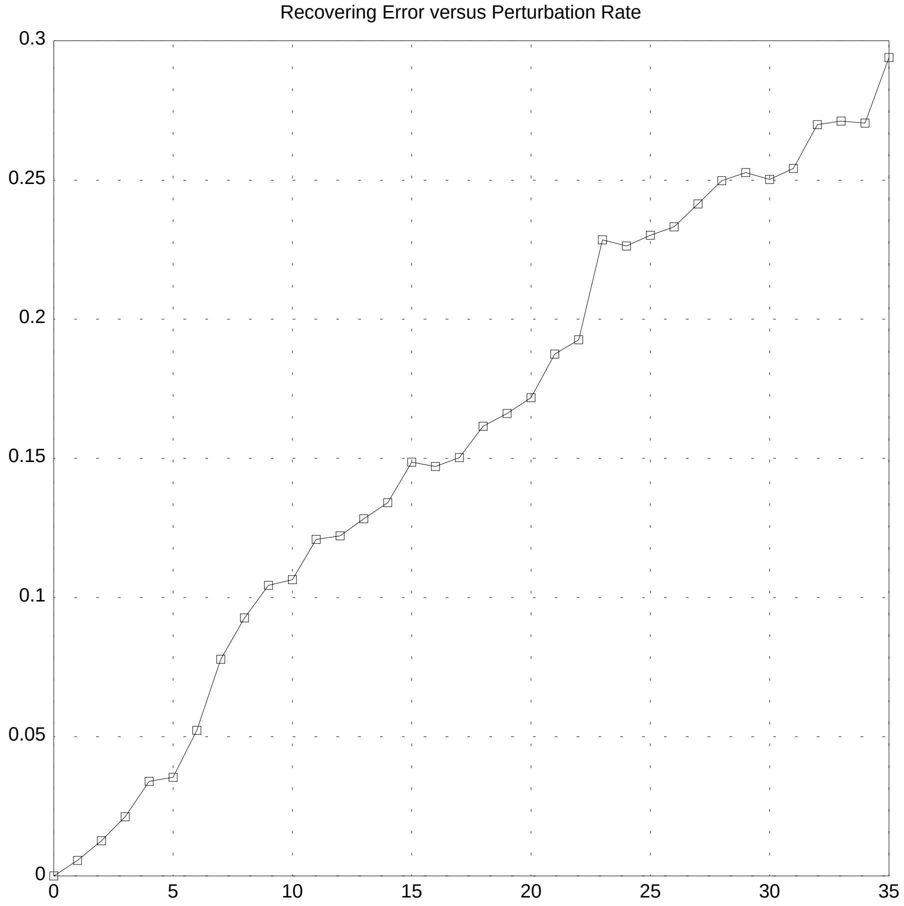

Figure 6: Errors in recovering affine parameters L_{ij} from the data extracted with errors. The horizontal axis shows the percentage of the Gaussian deviation to the average distance between closest features and the vertical axis shows the error in recovering L_{ij}. One hundred model and data pairs were used for each of the perturbation ratio, and 50 features were included in the model and data. Errors are almost proportional to the the perturbation rate.

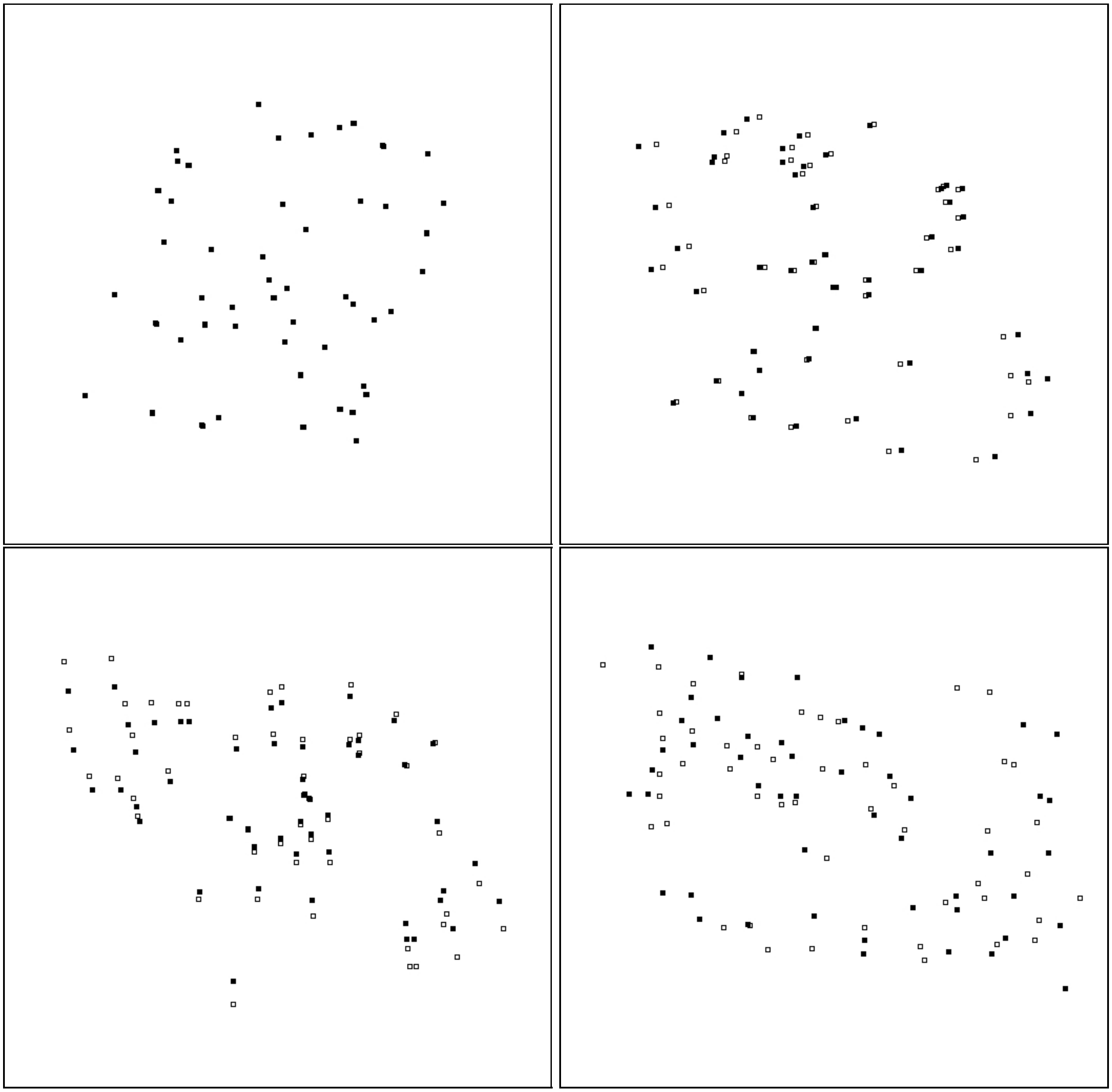

Figure 7: Reconstructed data features by the recovered affine parameters

Reconstructed data features are superimposed on the data generated with no errors: with the error in recovering L_{ij} Upper left: 0.0027, Upper right: 0.069, Lower left: 0.11, Lower right: 0.27. White boxes shows the data features without errors, while the black boxes show the reconstructed features.

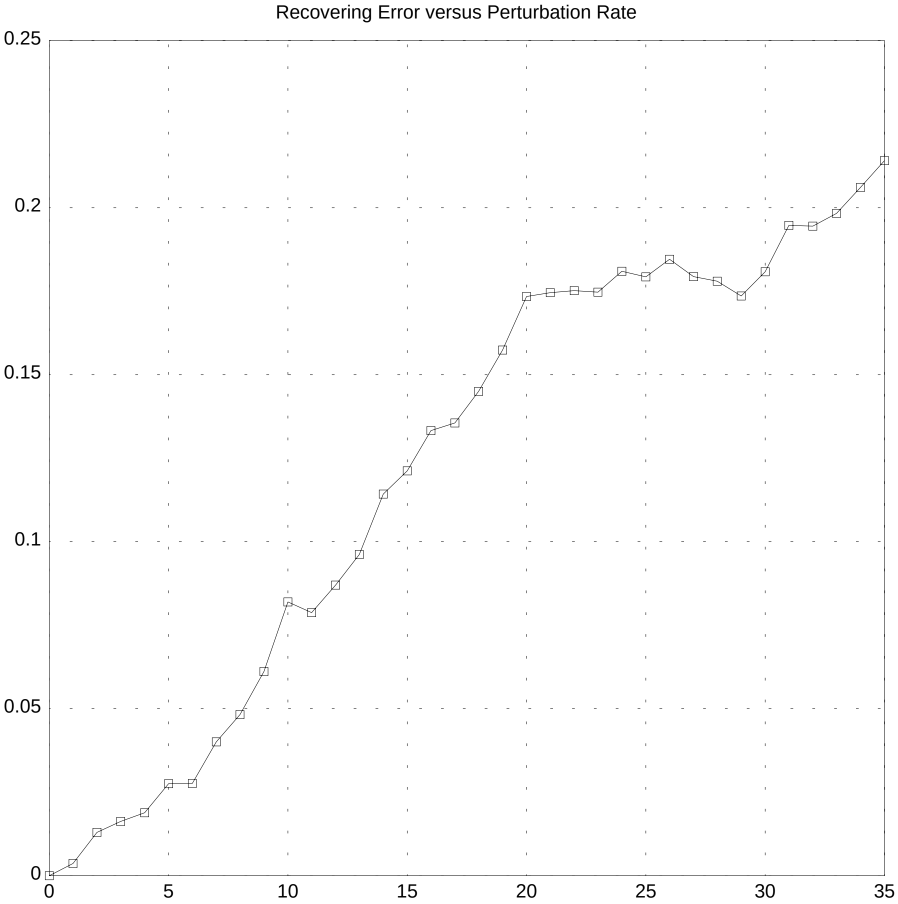

Figure 8: Errors in recovering affine parameters L_{ij} from datum with depth perturbations. The horizontal axis shows the percentage of the Gaussian deviation to the average distance between closest features and the vertical axis shows the error in recovering L_{ij}. One hundred model and data pairs were used for each of the perturbation ratio, and 50 features were included in each model and data. For small depth perturbations, the recovered affine parameters can work as a good approximate.

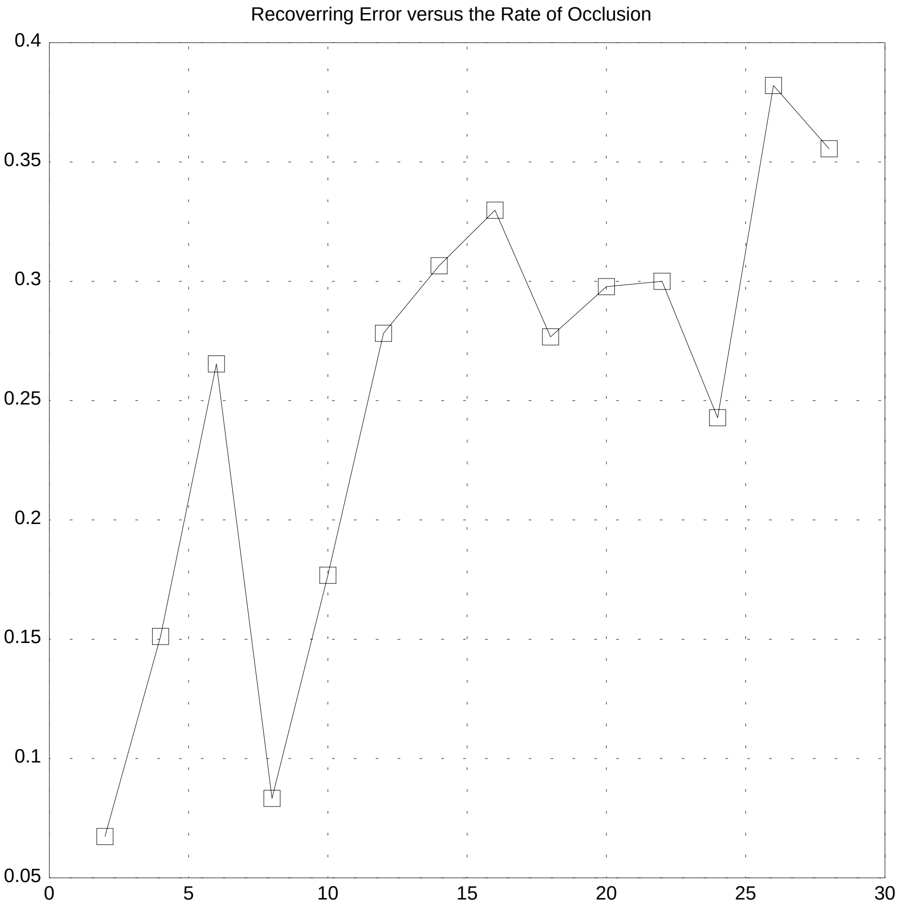

Figure 9: Errors in recovering affine parameters L_{ij} in case with occlusion.
The horizontal axis shows the percentages of the missing features and the vertical axis shows the error in recovering L_{ij}. The number of model features was 50. One hundred model and data pairs were used for each of the rate of missing features in the data.

[**Computational cost**]
The run time computational cost for recovering affine parameters was in average less than 15 msec on SPARCstation IPX. Compared with the conventional approaches to object recognition, this is a noticeable improvement.

7 Conclusion

It was shown that for sets of 2D image features from a planar surface, there exists a class of transformations that yield a unique distribution up to rotations. Also, the use of centroid correspondences between corresponding feature groups was proposed in the recognition of objects. Then, we proposed an approach to the alignment of the model of a planar object with its novel view as a combination of these two convenient tools. An algorithm was presented using clustering techniques for forming the feature groups. Then, experimental results demonstrated the robustness and computational merit of this approach. We also proposed an operator to detect planar portions of the object surface using two object images and showed its effectiveness through experiments.

Acknowledgments

Kenji Nagao is thankful to Henry Minsky for his help in getting accustomed to the computer environment at the MIT AI Lab. He is thankful to Dr. Amnon Shashua for his useful comments to this paper. He also thanks Marina Meila, Greg Klanderman, Aparna Lakshmi Ratan, and Kah Kay Sung for discussions on his research.

References

[1] S. Ando, "Gradient-Based Feature Extraction Operators for the Classification of Dynamical Images", Transactions of Society of Instrument and Control Engineers, vol.25, No.4, pp.496-503, 1989 (in Japanese).

[2] S. Ando, K. Nagao, "Gradient-Based Feature Extraction Operators for the Segmentation of Image Curves", Transactions of Society of Instrument and Control Engineers, vol.26, No.7, pp.826-832, 1990 (in Japanese).

[3] K. Fukunaga, *Introduction to Statistical Pattern Recognition*, Academic Press 1972.

[4] K. Fukunaga, W. L. G. Koontz, "A Criterion and an Algorithm for Grouping Data", IEEE Transactions on Computers, vol. c-19, No.10, pp.917-923, October 1970.

[5] W. E. L. Grimson, *Object Recognition by Computer*, MIT Press, 1991.

[6] Daniel P. Huttenlocher, Shimon Ullman, "Recognizing Solid Objects by Alignment with an Image", Inter. Journ. Comp. Vision, 5:2, pp.195-212, 1990.

[7] M. Iri, T. Kan, *Linear Algebra*, Kyouiku-Syuppan, pp.120-147, 1985 (in Japanese).

[8] Jan J. Koenderink, Andrea J. Van Doorn, "Affine structure form motion", Journ. Opt. Soc. Am., 8:377-385, 1991

[9] J. MacQueen, "Some methods for classification and analysis of multivariate observations", In Proc. 5th Berkeley Symp. on Probability and Statistics, pp.281-297, 1967.

[10] K. Nagao, M. Sohma, K. Kawakami, S. Ando, "Detecting Contours in Image Sequences", Transactions of the Institute of Electronics, Information and Communication Engineers in Japan on Information and Systems, vol. E76-D, No.10, pp. 1162-1173, 1993 (in English)

[11] A. Shashua, "Correspondence and Affine Shape from two Orthographic Views: Motion and Recognition", A.I. Memo No. 1327, Artificial Intelligence Laboratory, Massachusetts Institute of Technology, December 1991.

[12] Michael J. Swain, Color Indexing, PhD Thesis, Chapter 3, University of Rochester Technical Report No. 360, November 1990.

[13] S. K. Nayar and R. M. Bolle, "Reflectance Ratio: A Photometric Invariant for Object Recognition" In Proc. Fourth International Conference on Computer Vision, pp.280–285, 1993.

[14] T. F. Syeda-Mahmood, "Data and Model-driven Selection using Color Regions", In Proc. European Conference on Computer Vision, pp.321-327, 1992.

[15] W. B. Thompson, K. M. Mutch and V. A. Berzins, "Dynamic Occlusion Analysis in Optical Flow Fields", IEEE Transaction on Pattern Analysis and Machine Intelligence, vol. PAMI-7, pp.374–383, 1985.

[16] S. Ullman and R. Basri, "Recognition by Linear Combinations of Models", IEEE Transactions on Pattern Analysis and Machine Intelligence, **13**(10),pp.992–1006, 1991.

Appendix

In this Appendix, we show the validity of lemma 3. That is

$det[C_U] = 0 \Longleftrightarrow \hat{x'} = L_{11}\hat{x} + L_{12}\hat{y}$ for some real constant $(L_{11}, L_{12}) \neq (0, 0)$.

Proof)
$det[C_U] = 0$ is equivalent to that the column vectors C_{U1}, C_{U2}, C_{U3} of C_U are linearly dependent.
Specifically, for some constant α, β, γ

$$\alpha C_{U1} + \beta C_{U2} + \gamma C_{U3} = 0 \qquad (69)$$

This is equivalent to,

$$\sum \alpha\hat{x}\hat{U} + \beta\hat{y}\hat{U} + \gamma\hat{x'}\hat{U} = 0 \qquad (70)$$

where $\hat{U} = (\hat{x}, \hat{y}, \hat{x'})$, and the summation is taken over all the features concerned.

Premultiplying (α, β, γ) to (70) yields,

$$\sum (\alpha\hat{x} + \beta\hat{y} + \gamma\hat{x'})^2 = 0 \qquad (71)$$

When the number of features is sufficiently large, this is equivalent to,

$$\alpha\hat{x} + \beta\hat{y} + \gamma\hat{x'} = 0 \tag{72}$$

Ignoring the case where β/α, γ/α have infinite values, we obtain

$$\hat{x} + (\beta/\alpha)\hat{y} + (\gamma/\alpha)\hat{x'} = 0 \tag{73}$$

Then by setting $L_{11} = \beta/\alpha$ and $L_{12} = \gamma/\alpha$, we have the lemma.
□