MASSACHUSETTS INSTITUTE OF TECHNOLOGY
ARTIFICIAL INTELLIGENCE LABORATORY

A.I. Memo No. 1152 Aug., 1989

Recognition by Linear Combinations of Models

Shimon Ullman and Ronen Basri[1]

Abstract. Visual object recognition requires the matching of an image with a set of models stored in memory. In this paper we propose an approach to recognition in which a 3-D object is represented by the linear combination of 2-D images of the object. If $\mathcal{M} = \{M_1, ..., M_k\}$ is the set of pictures representing a given object, and P is the 2-D image of an object to be recognized, then P is considered an instance of M if $P = \sum_{i=1}^{k} \alpha_i M_i$ for some constants α_i. We show that this approach handles correctly rigid 3-D transformations of objects with sharp as well as smooth boundaries, and can also handle non-rigid transformations. The paper is divided into two parts. In the first part we show that the variety of views depicting the same object under different transformations can often be expressed as the linear combinations of a small number of views. In the second part we suggest how this linear combination property may be used in the recognition process.

Acknowledgments: 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 an Office of Naval Research University Research Initiative grant under contract N00014-86-K-0685, and 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 contract N00014-85-K-0124.

©Massachusetts Institute of Technology 1989.

[1]Department of Applied Math. The Weizmann Institute of Science, Rehovot, Israel

Recognition by Linear Combinations of Models

1 Modeling Objects by the Linear Combination of Images

1.1 Recognition by Alignment

Visual object recognition requires the matching of an image with a set of models stored in memory. Let $\mathcal{M} = \{M_1, ..., M_n\}$ be the set of stored models, and P be the image to be recognized. In general, the viewed object, depicted by P, may differ from all the previously seen images of the same object. It may be, for instance, the image of a three-dimensional object seen from a novel viewing position. To compensate for these variations, we may allow the models (or the viewed object) to undergo certain compensating transformations during the matching stage. If $\mathcal{T}$ is the set of allowable transformations, the matching stage requires the selection of a model $M_i \in \mathcal{M}$ and a transformation $T \in \mathcal{T}$, such that the viewed object P and the transformed model TM_i will be as close as possible. The general scheme is called the alignment approach, since an alignment transformation is applied to the model (or to the viewed object) prior to, or during the matching stage. Such an approach is used in [Chien & Aggarwal 1987, Faugeras & Hebert 1986, Fishler & Bolles 1981, Huttenlocher & Ullman 1987, Lowe 1985, Thompson & Mundy 1987, Ullman 1986]. Key problems that arise in any alignment scheme are how to represent the set of different models $\mathcal{M}$, what is the set of allowable transformations $\mathcal{T}$, and, for a given model $M_i \in \mathcal{M}$, how to determine the transformation $T \in \mathcal{T}$ so as to minimize the difference between P and TM_i. For example, in the scheme proposed by Basri and Ullman [1988] a model is represented by a set of 2-D contours, with associated depth and curvature values at each contour point. The set of allowed transformations includes 3-D rotation, translation and scaling, followed by an orthographic projection. The

transformation is determined as in [Huttenlocher & Ullman 1987, Ullman 1986, 1989] by identifying at least three corresponding features (points or lines) in the image and the object.

In this paper we suggest a different approach, in which each model is represented by the linear combination of 2-D images of the object. The new approach has several advantages. First, it handles all the rigid 3-D transformations, but it is not restricted to such transformations. Second, there is no need in this scheme to explicitly recover and represent the 3-D structure of objects. Third, the computations involved are often simpler than in previous schemes.

The paper is divided into two parts. In the first (section 1) we show that the variety of views depicting the same object under different transformations can often be expressed as the linear combinations of a small number of views. In the second part (section 2) we suggest how this linear combination property may be used in the recognition process.

1.2 Using Linear Combinations of Images to Model Objects and Their Transformations

The modeling of objects using linear combinations of images is based on the following observation. For many continuous transformations of interest in recognition, such as 3-D rotation, translation and scaling, all the possible views of the transforming object can be expressed simply as the linear combination of other views of the same object. The coefficients of these linear combinations often follow in addition certain functional restrictions. In the next two sections we show that the set of possible images of an object undergoing rigid 3-D transformations and scaling is embedded in a linear space, spanned by a small number of 2-D images.

The images we will consider are 2-D edge maps produced in the image by the (orthographic) projection of the bounding contours and other visible contours on 3-D objects. We will make use of the following definitions. Given an object and a viewing direction, the *rim* is the set of all the points on the object's surface, whose normal is perpendicular to the viewing direction [Koenderink & Van Doorn 1979]. This set is also called the *contour generator* [Marr 1977]. A *silhouette* is an image generated by the orthographic projection of the rim. In the analysis below we assume that every point along the silhouette is generated by a single rim point. An edge map of an object usually contains the silhouette, which is generated by its rim.

We will examine below two cases. The case of objects with sharp edges, and the case of objects with smooth boundary contours. The difference between these two cases is illustrated in Figure 1. For an object with sharp edges, such as the cube in Fig. 1 (a &

b), the rim is stable on the object as long as the edge is visible. In contrast, a rim that is generated by smooth bounding surfaces, such as in the ellipsoid in Fig. 1 (c & d), is not fixed on the object, but changes continuously with the viewpoint.

1.3 Objects with Sharp Edges

In the discussion below we examine the case of objects with sharp edges undergoing different transformations followed by an orthographic projection. In each case we show how the image of an object obtained by the transformation in question can be expressed as the linear combination of a small number of pictures. The coefficients of this combination may be different for the x- and y-coordinates. That is, the intermediate view of the object may be given by two linear combinations, one for the x-coordinates and the other for the y-coordinates. In addition, certain functional restrictions may hold among the different coefficients.

To introduce the scheme we first apply it to the restricted case of rotation about the vertical axis, then examine more general transformations.

1.3.1 3-D Rotation Around the Vertical Axis

Let P_1 and P_2 be two images of an object O rotating in depth around the vertical axis (Y-axis). P_2 is obtained from P_1 following a rotation by an angle α, ($\alpha \neq k\pi$). Let $\hat{P}$ be a third image of the same object obtained from P_1 by a rotation of an angle θ around the vertical axis. The projections of a point $p = (x, y, z) \in O$ in the three images are given by:

$$\begin{array}{lllll} p_1 & = & (x_1, y_1) & = & (x, y) \qquad\qquad\qquad\quad \in P_1 \\ p_2 & = & (x_2, y_2) & = & (x\cos\alpha + z\sin\alpha,\ y) \in P_2 \\ \hat{p} & = & (\hat{x}, \hat{y}) & = & (x\cos\theta + z\sin\theta,\ y) \in \hat{P} \end{array}$$

Claim: Two scalars a and b exist, such that for every point $p \in O$:

$$\hat{x} = ax_1 + bx_2$$

with:

$$a^2 + b^2 + 2ab\cos\alpha = 1$$

Proof: The scalars a and b are given explicitly by:

$$\begin{array}{rcl} a & = & \dfrac{\sin(\alpha - \theta)}{\sin\alpha} \\ b & = & \dfrac{\sin\theta}{\sin\alpha} \end{array}$$

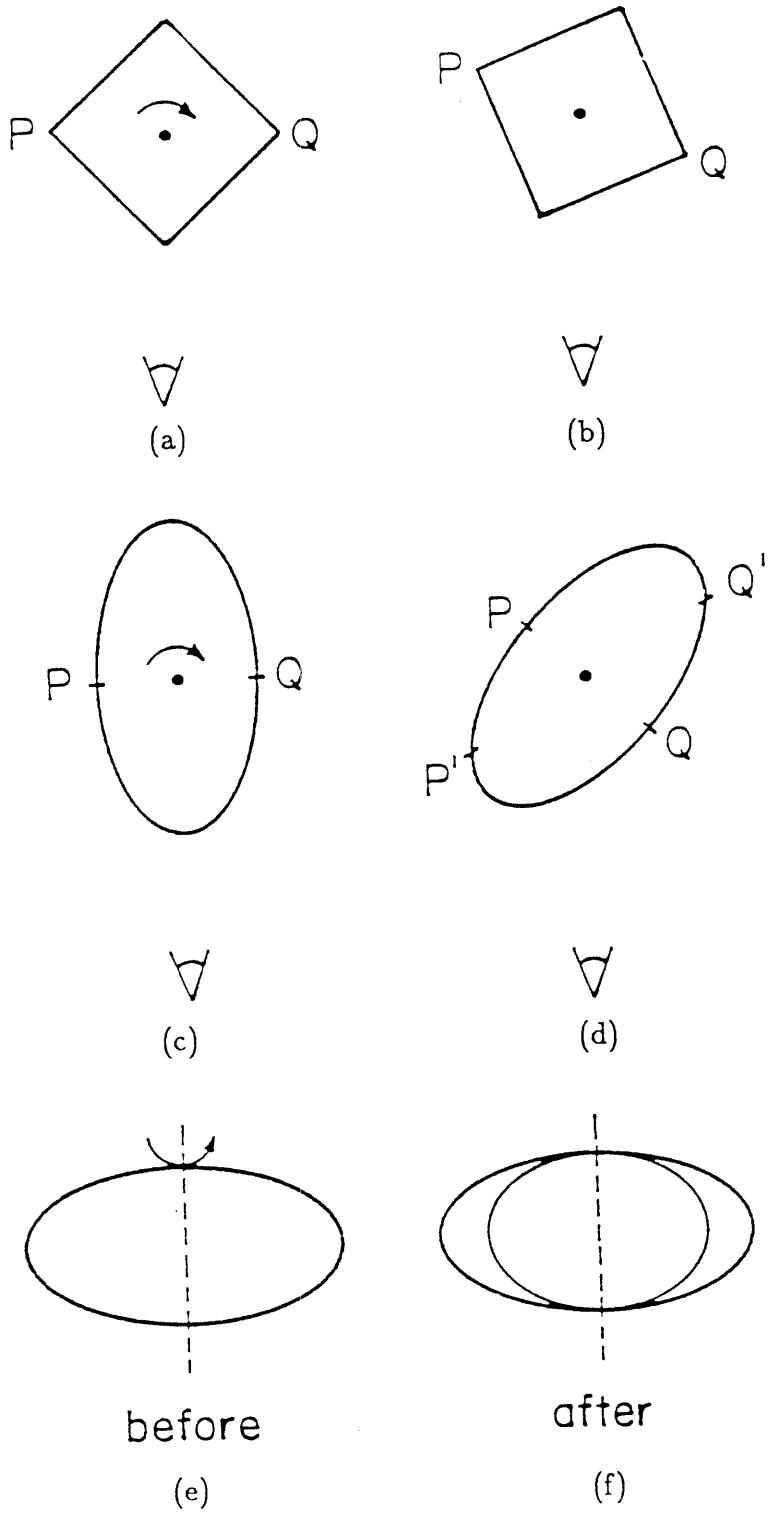

Figure 1: Changes in the rim during rotation. (a) A bird's eye view of a cube. (b) The cube after rotation. In both (a) and (b) points **p**, **q** lie on the rim. (c) A bird's eye view of an ellipsoid. (d) The ellipsoid after rotation. The rim points **p**, **q** in (c) are replaced by **p'**, **q'** in (d). (e) An ellipsoid in a frontal view. (f) The rotated ellipsoid (outer), superimposed on the appearance of the rim, as a planar space curve after rotation by the same amount (inner) (From [Basri & Ullman, 1988]).

Then:

$$ax_1 + bx_2 = \frac{\sin(\alpha - \theta)}{\sin \alpha} x + \frac{\sin \theta}{\sin \alpha}(x \cos \alpha + z \sin \alpha) = x \cos \theta + z \sin \theta = \hat{x}$$

Therefore, an image of an object rotating around the vertical axis is always a linear combination of two model images. It is straightforward to verify that the coefficients a and b satisfy the above constraint. It is worth noting that the new view $\hat{P}$ is not restricted to be an intermediate view (that is, the rotation angle θ may be larger than α). Finally, it should be noted that we do not deal at this stage with occlusion, we assume here that the same set of points is visible in the different views.

1.3.2 Linear Transformations in 3-D Space

Let O be a set of object points. Let P_1, P_2 and P_3 be three images of O, obtained by applying 3×3 matrices R, S and T to O, respectively. (In particular, R can be the identity matrix, and R, S two rotations producing the second and third views.) Let $\hat{P}$ be a fourth image of the same object obtained by applying a different 3×3 matrix U to O. Let $\mathbf{r}_1$, $\mathbf{s}_1$, $\mathbf{t}_1$ and $\mathbf{u}_1$ be the first row vectors of R, S, T and U, respectively, and let $\mathbf{r}_2$, $\mathbf{s}_2$, $\mathbf{t}_2$ and $\mathbf{u}_2$ be the second row vectors of R, S, T and U respectively. The positions of a point $p \in O$ in the four images are given by:

$$\begin{array}{lcll} p_1 & = & (x_1, y_1) & = & (\mathbf{r}_1 p, \mathbf{r}_2 p) \\ p_2 & = & (x_2, y_2) & = & (\mathbf{s}_1 p, \mathbf{s}_2 p) \\ p_3 & = & (x_3, y_3) & = & (\mathbf{t}_1 p, \mathbf{t}_2 p) \\ \hat{p} & = & (\hat{x}, \hat{y}) & = & (\mathbf{u}_1 p, \mathbf{u}_2 p) \end{array}$$

Claim: If both sets $\{\mathbf{r}_1, \mathbf{s}_1, \mathbf{t}_1\}$ and $\{\mathbf{r}_2, \mathbf{s}_2, \mathbf{t}_2\}$ are linearly independent, then there exist scalars a_1, a_2, a_3 and b_1, b_2, b_3 such that for every point $p \in O$ it holds that:

$$\begin{array}{rcl} \hat{x} & = & a_1 x_1 + a_2 x_2 + a_3 x_3 \\ \hat{y} & = & b_1 y_1 + b_2 y_2 + b_3 y_3 \end{array}$$

Proof: $\{\mathbf{r}_1, \mathbf{s}_1, \mathbf{t}_1\}$ are linearly independent. Therefore, they span $\mathcal{R}^3$, and there exist scalars a_1, a_2 and a_3 such that:

$$\mathbf{u}_1 = a_1 \mathbf{r}_1 + a_2 \mathbf{s}_1 + a_3 \mathbf{t}_1$$

Since:

$$\hat{x} = \mathbf{u}_1 p$$

It follows that:

$$\hat{x} = a_1 \mathbf{r}_1 p + a_2 \mathbf{s}_1 p + a_3 \mathbf{t}_1 p$$

Therefore:

$$\hat{x} = a_1 x_1 + a_2 x_2 + a_3 x_3$$

In a similar way we obtain that:

$$\hat{y} = b_1 y_1 + b_2 y_2 + b_3 y_3$$

Therefore, an image of an object undergoing a linear transformation in 3-D space is a linear combination of three model images.

1.3.3 General Rotation in 3-D Space

Rotation is a nonlinear subgroup of the linear transformations. Therefore, an image of a rotating object is still a linear combination of three model images. However, not every point in this linear space represents a pure rotation of the object. Indeed, we can show that only points that satisfy the following three constraints represent images of a rotating object.

Claim: The coefficients of an image of a rotating object must satisfy the three following constraints:

$$\begin{aligned} \| \, a_1 \mathbf{r}_1 + a_2 \mathbf{s}_1 + a_3 \mathbf{t}_1 \, \| &= 1 \\ \| \, b_1 \mathbf{r}_2 + b_2 \mathbf{s}_2 + b_3 \mathbf{t}_2 \, \| &= 1 \\ (a_1 \mathbf{r}_1 + a_2 \mathbf{s}_1 + a_3 \mathbf{t}_1)\,(b_1 \mathbf{r}_2 + b_2 \mathbf{s}_2 + b_3 \mathbf{t}_2) &= 0 \end{aligned}$$

Proof: U is a rotation matrix. Therefore:

$$\begin{aligned} \| \, \mathbf{u}_1 \, \| &= 1 \\ \| \, \mathbf{u}_2 \, \| &= 1 \\ \mathbf{u}_1 \, \mathbf{u}_2 &= 0 \end{aligned}$$

And the required terms are obtained directly by substituting $\mathbf{u}_1$ and $\mathbf{u}_2$ with the appropriate linear combinations. It also follows immediately that if the constraints are met, then the new view represents a possible rotation of the object.

These functional constraints are second degree polynomials in the coefficients, and therefore span a nonlinear manifold within the linear subspace. In order to check whether a specific set of coefficients represents a rigid rotation, the values of the matrices R, S and T are required. These can be retrived by applying methods of "structure from motion"

to the model views. Ullman [1979] showed that in case of rigid transformations four corresponding points in three views are sufficient. A linear algorithm that can be used to recover the rotation matrices has been suggested by Huang & Lee [1989]. (The same method can be extended to deal with scale changes, in addition to the rotation.)

It should be noted that in some cases the explicit computation of the rotation matrices will not be necessary. First, if the set of allowable object transformations includes the entire set of linear 3-D transformations (including non-rigid stretch and shear), then no additional test of the coefficients is required. Second, if the transformations are constrained to be rigid, but the test of the coefficient is not performed, then the penalty may be some "false positives" misidentifications. If the image of one object happens to be identical to the projection of a (non-linear) rigid transformation applied to another object, then the two will be confuseable. If the objects contain a sufficient number of points (five or more), the likelihood of such an ambiguity becomes negligible. Finally, it is worth noting that it is also possible to determine the coefficient of the constraint equations above without computing the rotation matrices, by using a number of additional views (see also section 1.3.5).

Regarding the independence condition mentioned above, for many triplets of rotation matrices R, S and T both $\{\mathbf{r}_1, \mathbf{s}_1, \mathbf{t}_1\}$ and $\{\mathbf{r}_2, \mathbf{s}_2, \mathbf{t}_2\}$ will in fact be linearly independent. It will therefore be possible to select a non degenerate triplet of views (P_1, P_2 and P_3), in terms of which intermediate views are expressible as linear combinations. Note, however, that in the special case that R is the identity matrix, S is a pure rotation about the X-axis, and T about the Y-axis, the independence condition does not hold.

1.3.4 Rigid Transformations and Scaling in 3-D Space

Rotation, translation and scaling in 3-D space can be represented as linear transformations in 4-D space using homogenous coordinates. Therefore, an image of a rigid object can be expressed as the linear combination of four model images. In fact, only three different snapshots of the object are required, the fourth view can be derived from them.

Let O be a set of object points. Let P_1, P_2 and P_3 be three images of O, obtained by applying the 3×3 rotation matrices R, S and T to O, respectively. Let $\hat{P}$ be a fourth image of the same object obtained by applying a 3×3 rotation matrix U to O, scaling by a scale factor s, and translating by a vector (t_x, t_y). Let $\mathbf{r}_1$, $\mathbf{s}_1$, $\mathbf{t}_1$ and $\mathbf{u}_1$ be again the first row vectors of R, S, T and U, and $\mathbf{r}_2$, $\mathbf{s}_2$, $\mathbf{t}_2$ and $\mathbf{u}_2$ the second row vectors of R, S,

T and U, respectively. For any point $p \in O$, its positions in the four images are given by:

$$\begin{array}{rcccl} p_1 & = & (x_1, y_1) & = & (\mathbf{r}_1 p,\ \mathbf{r}_2 p) \\ p_2 & = & (x_2, y_2) & = & (\mathbf{s}_1 p,\ \mathbf{s}_2 p) \\ p_3 & = & (x_3, y_3) & = & (\mathbf{t}_1 p,\ \mathbf{t}_2 p) \\ \hat{p} & = & (\hat{x}, \hat{y}) & = & (s\mathbf{u}_1 p + t_x,\ s\mathbf{u}_2 p + t_y) \end{array}$$

Claim: If both sets $\{\mathbf{r}_1, \mathbf{s}_1, \mathbf{t}_1\}$ and $\{\mathbf{r}_2, \mathbf{s}_2, \mathbf{t}_2\}$ are linearly independent, then there exist scalars a_1, a_2, a_3, a_4, and b_1, b_2, b_3, b_4, such that for every point $p \in O$ it holds that:

$$\begin{aligned} \hat{x} &= a_1 x_1 + a_2 x_2 + a_3 x_3 + a_4 \\ \hat{y} &= b_1 y_1 + b_2 y_2 + b_3 y_3 + b_4 \end{aligned}$$

with the coefficient satisfying the two constraints:

$$\| a_1\mathbf{r}_1 + a_2\mathbf{s}_1 + a_3\mathbf{t}_1 \| = \| b_1\mathbf{r}_2 + b_2\mathbf{s}_2 + b_3\mathbf{t}_2 \|$$

$$(a_1\mathbf{r}_1 + a_2\mathbf{s}_1 + a_3\mathbf{t}_1)(b_1\mathbf{r}_2 + b_2\mathbf{s}_2 + b_3\mathbf{t}_2) = 0$$

Proof: $\{\mathbf{r}_1, \mathbf{s}_1, \mathbf{t}_1\}$ are linearly independent. Therefore, they span $\mathcal{R}^3$, and there exist scalars c_1, c_2 and c_3 such that:

$$\mathbf{u}_1 = c_1\mathbf{r}_1 + c_2\mathbf{s}_1 + c_3\mathbf{t}_1$$

Since:

$$\hat{x} = s(\mathbf{u}_1 p) + t_x$$

Then

$$\hat{x} = sc_1\mathbf{r}_1 p + sc_2\mathbf{s}_1 p + sc_3\mathbf{t}_1 p + t_x$$

Let:

$$\begin{aligned} a_1 &= sc_1 \\ a_2 &= sc_2 \\ a_3 &= sc_3 \\ a_4 &= t_x \end{aligned}$$

We obtain that:

$$\hat{x} = a_1 x_1 + a_2 x_2 + a_3 x_3 + a_4$$

In a similar way we obtain that:

$$\hat{y} = b_1 y_1 + b_2 y_2 + b_3 y_3 + b_4$$

U is rotation matrix, therefore:

$$\begin{aligned} \| \mathbf{u}_1 \| &= 1 \\ \| \mathbf{u}_2 \| &= 1 \\ \mathbf{u}_1 \mathbf{u}_2 &= 0 \end{aligned}$$

It follows that:

$$\begin{aligned} \| s\mathbf{u}_1 \| &= \| s\mathbf{u}_2 \| \\ (s\mathbf{u}_1)(s\mathbf{u}_2) &= 0 \end{aligned}$$

And the constraints are obtained directly by substituting the appropriate linear combinations for $s\mathbf{u}_1$ and $s\mathbf{u}_2$.

1.3.5 Using Two Views Only

In the scheme described above, any image of a given object (within a certain range of rotations) is expressed as the linear combination of three fixed views of the object. For general linear transformations, it is also possible to use instead just two views of the object. (This observation was made independently by T. Poggio and R. Basri.)

Let O be again a rigid object (a collection of 3-D points). P_1 is a 2-D image of O, and P_2 the image of O following a rotation by R (a 3×3 matrix). We will denote by $\mathbf{r}_1$, $\mathbf{r}_2$, $\mathbf{r}_3$, the three rows of R, and by $\mathbf{e}_1$, $\mathbf{e}_2$, $\mathbf{e}_3$, the three rows of the identity matrix. For a given 3-D point $\mathbf{p}$ in O, its coordinates (x_1, y_1) in the first image view are $x_1 = \mathbf{e}_1\mathbf{p}$, $y_1 = \mathbf{e}_2\mathbf{p}$. Its coordinates (x_2, y_2) in the second view are given by: $x_2 = \mathbf{r}_1\mathbf{p}$, $y_2 = \mathbf{r}_2\mathbf{p}$.

Consider now any other view obtained by applying another 3×3 matrix $\mathbf{U}$ to the points of O. The coordinates $(\hat{x}, \hat{y})$ of $\mathbf{p}$ in this new view will be:

$$\hat{x} = \mathbf{u}_1\mathbf{p}, \quad \hat{y} = \mathbf{u}_2\mathbf{p}$$

(where $\mathbf{u}_1$, $\mathbf{u}_2$, are the first and second rows of $\mathbf{U}$, respectively).

Assuming that $\mathbf{e}_1$, $\mathbf{e}_2$ and $\mathbf{r}_1$ span $\mathcal{R}^3$ (see below), then:

$$\mathbf{u}_1 = a_1\mathbf{e}_1 + a_2\mathbf{e}_2 + a_3\mathbf{r}_1$$

for some scalars a_1, a_2, a_3. Therefore:

$$\hat{x} = \mathbf{u}_1\mathbf{p} = (a_1\mathbf{e}_1 + a_2\mathbf{e}_2 + a_3\mathbf{r}_1)\mathbf{p} = a_1x_1 + a_2y_1 + a_3x_2$$

This equality holds for every point $\mathbf{p}$ in O, Let $\mathbf{x}_1$ be the vector of all the x-coordinates of the points in the first view, $\mathbf{x}_2$ in the second, $\hat{\mathbf{x}}$ in the third, and $\mathbf{y}_1$ the vector of y-coordinates in the first view. Then:

$$\hat{\mathbf{x}} = a_1\mathbf{x}_1 + a_2\mathbf{y}_1 + a_3\mathbf{x}_2$$

Here $\mathbf{x}_1$, $\mathbf{y}_1$ and $\mathbf{x}_2$ are used as a basis for all of the views. For any other image of the same object, its vector $\hat{\mathbf{x}}$ of x-coordinates is the linear combination of these basis vectors.

Similarly, for the y-coordinates:

$$\hat{\mathbf{y}} = b_1\mathbf{x}_1 + b_2\mathbf{y}_1 + b_3\mathbf{x}_2$$

The vector $\hat{\mathbf{y}}$ of y-coordinates in the new image is therefore also the linear combination of the same three basis vectors. In this version the basis vectors are the same for the x- and y-coordinates, and they are obtained from two rather then three views. One can view the situation as follows. Within an n-dimensional space, the vectors $\mathbf{x}_1$, $\mathbf{y}_1$, $\mathbf{x}_2$ span a 3-dimensional subspace. For all the images of the object in question, the vectors of both the x- and y-coordinates must reside within this 3-dimensional subspace.

Instead of using $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{r}_1)$ as the basis for $\mathcal{R}^3$ we could also use $(\mathbf{e}_1, \mathbf{e}_2, \mathbf{r}_2)$. One of these bases spans $\mathcal{R}^3$, unless the rotation R is a pure rotation around the line of sight.

The use of two views described above is applicable to general linear transformations of the object, and, without additional constraints, it is impossible to distinguish between rigid and linear but not rigid transformations of the object. To impose rigidity (with possible scaling) the coefficients $(a_1, a_2, a_3, b_1, b_2, b_3)$ must meet two simple constraints. Since $\mathbf{U}$ is now a rotation matrix (with possible scaling),

$$\mathbf{u}_1\,\mathbf{u}_2 \;=\; 0$$

$$\|\,\mathbf{u}_1\,\| = \|\,\mathbf{u}_2\,\|$$

In terms of the coefficients a_i, b_i, $\mathbf{u}_1\,\mathbf{u}_2 \;=\; 0$ implies:

$$a_1b_1 + a_2b_2 + a_3b_3 + (a_1b_3 + a_3b_1)r_{11} + (a_2b_3 + a_3b_2)r_{12} = 0$$

The second constraint implies:

$${a_1}^2 + {a_2}^2 + {a_3}^2 - {b_1}^2 - {b_2}^2 - {b_3}^2 = 2(b_1b_3 - a_1a_3)r_{11} + 2(b_2b_3 - a_2a_3)r_{12}$$

A third view can therefore be used to recover, using two linear equations, the values of r_{11} and r_{12}. (r_{11} and r_{12} can in fact be determined to within a scale factor from the first two views, only one additional equation is required.) The full scheme for rigid objects is then the following. Given an image, determine whether the vectors $\hat{\mathbf{x}}$, $\hat{\mathbf{y}}$, are linear combinations of $\mathbf{x}_1$, $\mathbf{y}_1$ and $\mathbf{x}_2$. Only two views are required for this stage. Using the values of r_{11} and r_{12}, test whether the coefficients a_i, b_i, $(i = 1, 2, 3)$ satisfy the two constraints above.

It is of interest to compare this use of two views to structure-from-motion (SFM) techniques for recovering 3-D structure from orthographic projections. It is well known

that three distinct views are required, two are insufficient [Ullman 1979]. Given only two views and an infinitesmal rotation (the velocity field), the 3-D structure can be recovered to within depth-scaling [Ullman 1983]. It is also straightforward to establish that if the two views are separated by a general affine transformation of the 3-D object (rather than a rigid one), then the structure of the object can be recovered to within an affine transformation.

Our use of two views above for the purpose of recognition is thus related to known results regarding the recovery of structure from motion. Two views are sufficient to determine the object's structure to within an affine transformation, and three are required to recover the full 3-D structure of a rigidly moving object. It can also be observed that an extension of the scheme above can be used to recover structure from motion. It was shown how the scheme can be used to recover r_{11} and r_{12}. r_{21} and r_{22} can be recovered in a similar manner. Consequently, it becomes possible to recover 3-D structure and motion in space based on three orthographic views, using linear equations.

1.3.6 Summary

In this section we have shown that an object with sharp contours, undergoing rigid transformations and scaling in 3-D space followed by an orthographic projection, can be expressed as the linear combination of four images of the same object. In this scheme, the model of a 3-D object consists of a number of 2-D pictures of it. The pictures are in correspondence, in the sense that it is known which are the corresponding points in the different pictures. Two images are sufficient to represent general linear transformations of the object. Three images are required to represent rotations in 3-D space, and one additional image is required to represent translations. The scaling does not require any additional image, since it is represented by a scaling of the coefficients. As mentioned above, the fourth picture can be generated internally, therefore only three different snapshots of the object are required.

The linear combination scheme assumes that the same object points are visible in the different views. When the views are sufficiently different, this will no longer hold, due to self-occlusion. To represent an object from all possible viewing directions (e.g. both "front" and "back"), a number of different models of this type will be required. This notion is similar to the use of different object aspects suggested by Koenderink & Van Doorn [1979]. (Other aspects of occlusion are examined in the final discussion and Appendix D.)

The linear combination scheme described above was implemented and applied first to artificially created images. Figure 2 shows examples of object models and their linear combinations. The figure shows how 3-D similarity transformations can be represented by the linear combinations of four images.

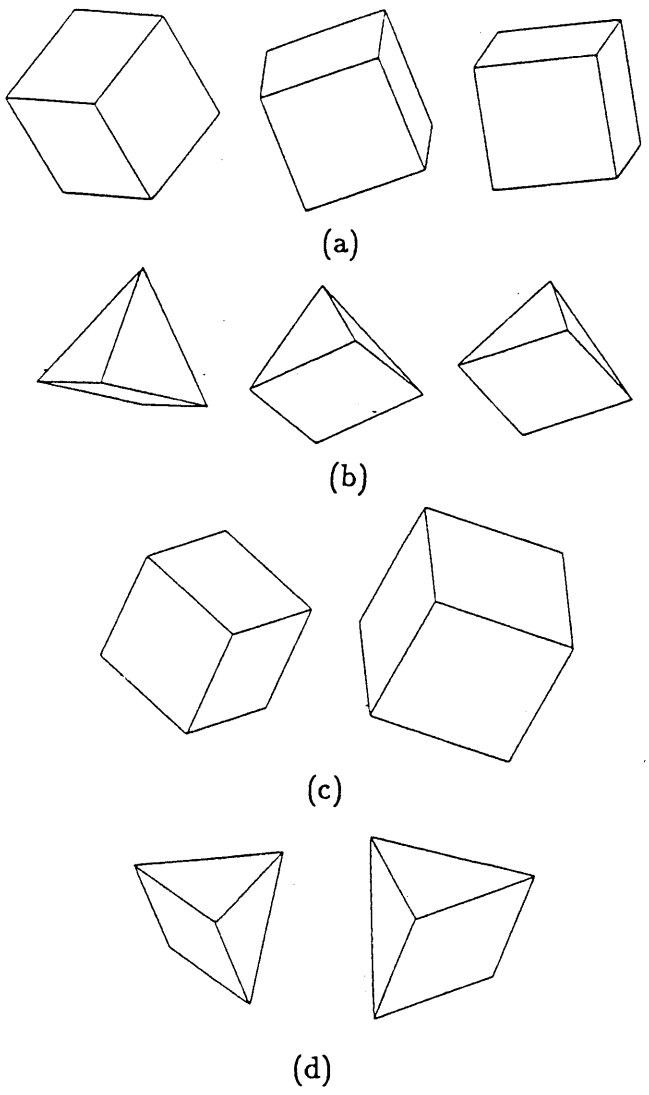

Figure 2: (a) Three model pictures of a cube. The second picture was obtained by rotating the cube by 30° around the X-axis, then by 30° around the Y-axis. The third picture was obtained by rotating the cube by 30° around the Y-axis, then by 30° around the X-axis. (b) Three model pictures of a pyramid taken with the same transformations as the pictures in (a). (c) Two linear combinations of the cube model. The left picture was obtained using the following parameters: the x-coefficients are $(0.343, -2.618, 2.989, 0)$, and the y-coefficients are $(0.630, -2.533, 2.658, 0)$, which correspond to a rotation of the cube by 10°, 20° and 45° around the X-, Y- and Z-axes respectively. The right picture was obtained using the following parameters: x-coefficients $(0.455, 3.392, -3.241, 0.25)$, y-coefficients $(0.542, 3.753, -3.343, -0.15)$. These coefficients correspond to a rotation of the cube by 20°, 10° and $-45°$ around the X-, Y- and Z-axes respectively, followed by a scaling of factor 1.2, and a translation of $(25, -15)$ pixels. (d) Two linear combinations of the pyramid model taken with the same parameters as the pictures in (c).

1.4 Objects with Smooth Boundaries

The case of objects with smooth boundaries is identical to the case of objects with sharp edges as long as we deal with translation, scaling and image rotation. The difference arises when the object rotates in 3-D space. This case is discussed in [Basri & Ullman, 1988], where we have suggested a method for predicting the appearance of such objects following 3-D rotations. This method, called "the curvature method", is summarized briefly below.

A model is represented by a set of 2-D contours. Each point $p = (x, y)$ along the contours is labeled with its depth value z, and a curvature value r. The curvature value is the length of a curvature vector at p, $r = \| (r_x, r_y) \|$. (r_x is the surface's radius of curvature at p in a planar section in the X direction, r_y in the Y direction.) This vector is normal to the contour at p. Let V_ϕ be an axis lying in the image plane and forming an angle ϕ with the positive X direction, and $\mathbf{r}_\phi$ be a vector of length $r_\phi = r_y \cos\phi - r_x \sin\phi$ and perpendicular to V_ϕ. When the object is rotated around V_ϕ we approximate the new position of the point p in the image by:

$$p' = R(p - \mathbf{r}_\phi) + \mathbf{r}_\phi \tag{1}$$

where R is the rotation matrix. The equation has the following meaning. When viewed in a cross section perpendicular to the rotation axis V_ϕ, the surface at p can be approximated by a circular arc with radius r_ϕ and center at $p - \mathbf{r}_\phi$. The new rim point p' is obtained by first applying R to this center of curvature $(p - \mathbf{r}_\phi)$, then adding the radius of curvature $\mathbf{r}_\phi$. This expression is precise for circular arcs, and gives a good approximation for other surfaces provided that the angle of rotation is not too large (see [Basri & Ullman 1988] for details). The depth and the curvature values were estimated in [Basri & Ullman 1988] using three pictures of the object, and the results were improved using five pictures. In this section we show how the curvature method can also be replaced by linear combinations of a small number of pictures. In particular, we use three images to represent rotations around the vertical axis, and five images for general rotations in 3-D space.

1.4.1 3-D Rotation Around the Vertical Axis

When an object rotates around the vertical (Y) axis by an angle θ, $\mathbf{r}_\phi$ in equation (1) above becomes $\mathbf{r}_{\frac{\pi}{2}}$, which is a horizontal vector of length $r_{\frac{\pi}{2}} = r_x$. Therefore, the new position of a point $p = (x, y)$ is given by $p' = (x', y')$ where:

$$\begin{aligned} x' &= (x - r_x)\cos\theta + z\sin\theta + r_x = x\cos\theta + z\sin\theta + r_x(1 - \cos\theta) \\ y' &= y \end{aligned}$$

This expression gives the new coordinates (x', y') in terms of the original coordinates (x, y), the rotation angel θ, the local depth z and the radius of curvature r_x. Next we show that the new image can be expressed instead as the linear combination of three 2-D images.

Let P_1, P_2 and P_3 be three images of an object O rotating around the vertical (Y) axis. P_2 is obtained from P_1 by a rotation by an angle α, and P_3 by a rotation by an angle β ($\alpha \neq \beta$, $\alpha, \beta \neq k\pi$). Let $\hat{P}$ be another image of the same object obtained from P_1 by a rotation by an angle θ around the vertical axis. We assume that the curvature scheme gives sufficiently close approximation to the images. Under this assumption, the positions of a point $p = (x, y, z) \in O$ can be expressed in the following manner:

$$\begin{array}{lllll} p_1 = (x_1, y_1) & = & (x, y) & \in & P_1 \\ p_2 = (x_2, y_2) & = & (x\cos\alpha + z\sin\alpha + r_x(1-\cos\alpha),\ y) & \in & P_2 \\ p_3 = (x_3, y_3) & = & (x\cos\beta + z\sin\beta + r_x(1-\cos\beta),\ y) & \in & P_3 \\ \hat{p} = (\hat{x}, \hat{y}) & = & (x\cos\theta + z\sin\theta + r_x(1-\cos\theta),\ y) & \in & \hat{P} \end{array}$$

Claim: $\hat{P}$ is a linear combination of P_1, P_2, P_3. That is, there exist scalars a, b and c such that for every four corresponding points $p_1, p_2, p_3, \hat{p}$:

$$\hat{x} = ax_1 + bx_2 + cx_3$$

with:

$$a + b + c = 1$$

and:

$$a^2 + b^2 + c^2 + 2ab\cos\alpha + 2ac\cos\beta + 2bc\cos(\beta - \alpha) = 1$$

Proof: We construct a, b and c explicitly. Let:

$$\begin{aligned} a &= \frac{\sin(\alpha-\theta) - \sin(\beta-\theta) - \sin(\alpha-\beta)}{\sin\alpha - \sin\beta - \sin(\alpha-\beta)} \\ b &= \frac{-\sin\beta + \sin\theta + \sin(\beta-\theta)}{\sin\alpha - \sin\beta - \sin(\alpha-\beta)} \\ c &= \frac{\sin\alpha - \sin\theta - \sin(\alpha-\theta)}{\sin\alpha - \sin\beta - \sin(\alpha-\beta)} \end{aligned}$$

($\alpha \neq \beta$ and $\alpha, \beta \neq k\pi$ implies that $\sin\alpha - \sin\beta - \sin(\alpha-\beta) \neq 0$). It follows that:

$$ax_1 + bx_2 + cx_3 =$$

$$= \frac{\sin(\alpha-\theta) - \sin(\beta-\theta) - \sin(\alpha-\beta)}{\sin\alpha - \sin\beta - \sin(\alpha-\beta)} x +$$

$$+\frac{-\sin\beta+\sin\theta+\sin(\beta-\theta)}{\sin\alpha-\sin\beta-\sin(\alpha-\beta)}(x\cos\alpha+z\sin\alpha+r_x(1-\cos\alpha))+$$
$$+\frac{\sin\alpha-\sin\theta-\sin(\alpha-\theta)}{\sin\alpha-\sin\beta-\sin(\alpha-\beta)}(x\cos\beta+z\sin\beta+r_x(1-\cos\beta))=$$
$$=(x\cos\theta+z\sin\theta+r_x(1-\cos\theta))=\hat{x}$$

Therefore, an image of an object rotating around the vertical axis and described accurately by the curvature method is always a linear combination of three model images. In addition, if we substitute the values above for a, b and c in the two functional constraints we obtain that:

$$a+b+c=1$$
$$a^2+b^2+c^2+2ab\cos\alpha+2ac\cos\beta+2bc\cos(\beta-\alpha)=1$$

1.4.2 General Rotation in 3-D Space

In this section we first derive an expression for the image deformation of an object with smooth boundries under general 3-D rotation. We then use this expression to show that the deformed image can be expressed as the linear combination of five images.

Computing the transformed image.

Using the curvature method we can predict the appearance of an object undergoing a general rotation in 3-D space as follows. A rotation in 3-D space can be decomposed into the following three successive rotations: a rotation around the Z-axis, a subsequent rotation around the X axis, and a final rotation around the Z-axis, by angles α, β and γ respectively. Since the Z-axis coincides with the line of sight, a rotation around the Z-axis is simply an image rotation. Therefore, only the second rotation deforms the object, and the curvature method must be applied to it. Suppose that the curvature vector at a given point $p=(x,y)$ before the first Z-rotation is (r_x,r_y). Following the rotation by α it becomes $r'_x=r_x\cos\alpha-r_y\sin\alpha$ and $r'_y=r_x\sin\alpha+r_y\cos\alpha$. The second rotation is around the X-axis, and therefore the appropriate r_ϕ to be used in eq. (1) becomes $r'_y=r_x\sin\alpha+r_y\cos\alpha$. The complete rotation (all three rotations) therefore takes a point $p=(x,y)$ through the following sequence of transformations:

$$(x,y)\longrightarrow(x\cos\alpha-y\sin\alpha,\ x\sin\alpha+y\cos\alpha)\longrightarrow$$
$$(x\cos\alpha-y\sin\alpha,\ (x\sin\alpha+y\cos\alpha)\cos\beta-z\sin\beta+(r_x\sin\alpha+r_y\cos\alpha)(1-\cos\beta))\longrightarrow$$
$$((x\cos\alpha-y\sin\alpha)\cos\gamma+((x\sin\alpha+y\cos\alpha)\cos\beta-z\sin\beta+(r_x\sin\alpha+r_y\cos\alpha)(1-\cos\beta))\sin\gamma,$$
$$(x\cos\alpha-y\sin\alpha)\sin\gamma+((x\sin\alpha+y\cos\alpha)\cos\beta-z\sin\beta+(r_x\sin\alpha+r_y\cos\alpha)(1-\cos\beta))\cos\gamma)$$

(The first of these transformations is the first Z-rotation, the second is the deformation caused by the X-rotation, and the third is the final Z-rotation).

This is an explicit expression of the final coordinates of a point on the object's contour. This can also be expressed more compactly as follows. Let $R = \{r_{ij}\}$ be a 3×3 rotation matrix. Let α, β and γ be the angles of the Z-X-Z rotations represented by R. We construct a new matrix $R' = \{r'_{ij}\}$ of size 2×5 as follows:

$$R' = \begin{pmatrix} r_{11} & r_{12} & r_{13} & \sin\alpha(1-\cos\beta)\sin\gamma & \cos\alpha(1-\cos\beta)\sin\gamma \\ r_{21} & r_{22} & r_{23} & \sin\alpha(1-\cos\beta)\cos\gamma & \cos\alpha(1-\cos\beta)\cos\gamma \end{pmatrix}$$

Let $p = (x, y)$ be a contour point with depth z and curvature vector (r_x, r_y), and let $\tilde{p} = (x, y, z, r_x, r_y)$. Then, the new appearance of $\mathbf{p}$ after a rotation R is applied to the object is described by:

$$p' = R'\tilde{p} \tag{2}$$

This is true because eq. (2) is equivalent to eq. (1) in section 1.4 with the appropriate values for r_ϕ.

Expressing the transformed image as a linear combination.

Let O be a set of points of an object rotating in 3-D space. Let P_1, P_2, P_3, P_4 and P_5 be five images of O, obtained by applying a rotation matrix $R_1, ..., R_5$ respectively. $\hat{P}$ is an image of the same object obtained by applying a rotation matrix $\hat{R}$ to O. Let $R'_1, ..., R'_5, \hat{R}'$ be the corresponding 2×5 matrices representing the transformations applied to the contour points according to the curvature method. Finally, let $\mathbf{r}_1, ..., \mathbf{r}_5, \hat{\mathbf{r}}$ denote the first row vectors of $R'_1, ..., R'_5, \hat{R}'$, and $\mathbf{s}_1, ..., \mathbf{s}_5, \hat{\mathbf{s}}$ the second row vectors $R'_1, ..., R'_5, \hat{R}'$ respectively. The positions of a point $p = (x, y) \in O$, $\tilde{p} = (x, y, z, r_x, r_y)$, in the six pictures is then given by:

$$\begin{array}{rcccl} p_i & = & (x_i, y_i) & = & (\mathbf{r}_i\tilde{p},\ \mathbf{s}_i\tilde{p}) \in P_i,\ 1 \le i \le 5 \\ \hat{p} & = & (\hat{x}, \hat{y}) & = & (\hat{\mathbf{r}}\tilde{p},\ \hat{\mathbf{s}}\tilde{p}) \in \hat{P} \end{array}$$

Claim: If both sets $\{\mathbf{r}_1, ..., \mathbf{r}_5\}$ and $\{\mathbf{s}_1, ..., \mathbf{s}_5\}$ are linearly independent vectors then there exist scalars $a_1, ..., a_5$ and $b_1, ..., b_5$ such that for every point $p \in O$ it holds that:

$$\hat{x} = \sum_{i=1}^{5} a_i x_i$$

$$\hat{y} = \sum_{i=1}^{5} b_i y_i$$

Proof: $\{\mathbf{r}_1, ..., \mathbf{r}_5\}$ are linearly independent. Therefore, they span $\mathcal{R}^5$, and there exist scalars $a_1, ..., a_5$ such that:

$$\hat{\mathbf{r}} = \sum_{i=1}^{5} a_i \mathbf{r}_i$$

Since:

$$\hat{x} = \hat{\mathbf{r}}\tilde{p}$$

Then:

$$\hat{x} = \sum_{i=1}^{5} a_i \mathbf{r}_i \tilde{p}$$

That is:

$$\hat{x} = \sum_{i=1}^{5} a_i x_i$$

In a similar way we obtain that:

$$\hat{y} = \sum_{i=1}^{5} b_i y_i$$

In addition, for pure rotation, the coefficients of this linear combinations satisfy seven functional constraints. These constraints, which are second degree polynomials, are given in Appendix A.

Again, one may or may not actually test for these additional constraints. If the test is ommitted, the probablity of a false-positive misidentification is slightly increased.

As in the case of sharp boundaries, it is possible to use mixed x- and y-coordinates to reduce the number of basic views for genral linear transformations (Section 1.3.5). For example, one can use five basis vectors $(\mathbf{x}_1, \mathbf{x}_2, \mathbf{x}_3, \mathbf{y}_1, \mathbf{y}_2)$ taken from three distict views as the basis for the x- and y-coordinates in all other views.

1.4.3 Rigid Transformation and Scaling in 3-D Space

So far we have shown that an object with smooth boundaries, represented by the curvature scheme, and undergoing a rotation in 3-D space, can be represented as a linear combination of 2-D views. The method can be easily extended to handle translation by taking, as before, an additional image of the object. The linear combination scheme for objects with smooth bounding contours is thus a direct extension of the scheme in section 1.3 for objects with sharp boundaries. In both cases, object views are expressed as the linear combination of a small number of pictures. The scheme for objects with sharp boundaries can be viewed as a special case of the more general one, when r, the radius of curvature, vanishes. In practice, we found that it is also possible to use the scheme for sharp boundaries, that uses a smaller number of views in each model, for general objects, provided that r is not too large (and at the price of increasing the number of models).

1.4.4 Summary

In this section we have shown that an object with smooth boundaries undergoing rigid transformations and scaling in 3-D space followed by an orthographic projection, can be expressed (within the approximation of the curvature method) as the linear combination of six images of the object. Five images are used to represent rotations in 3-D space, and one additional image is required to represent translations. (In fact, although the coordinates are expressed in terms of five basis vectors, only three distinct views are needed for a general linear transformation.) The scaling does not require any additional image since it is represented by a scaling of the coefficients. This scheme was implemented and applied to images of 3-D objects.

Figures 3 and 4 show the application of the LC (linear combination) method to complex objects with smooth bounding contours. Since the rotation was about the vertical axis, three 2-D views were used for each model. The figure shows a good agreement between the actual image and the appropriate linear combination. Although the objects are similar, they are easily discriminable by the LC method within the entire 60° rotation range.

Finally, it is worth noting that the modeling of objects by linear combinations of stored pictures is not limited only to rigid objects. The method can also be used to deal with various types of non-rigid transformations, such as articulations and non-rigid stretching. For example, in the case of an articulated object, the object is composed of a number of rigid parts linked together by joints that constraint the relative movement of the parts. We saw that the x- and y-coordinates of a rigid part are constrained to a 4-D subspace. Two rigid parts reside within an 8-D subspace, but, because of the constraints at the joints, they usually occupy a smaller subspace (e.g., 6-D for a planar joint).

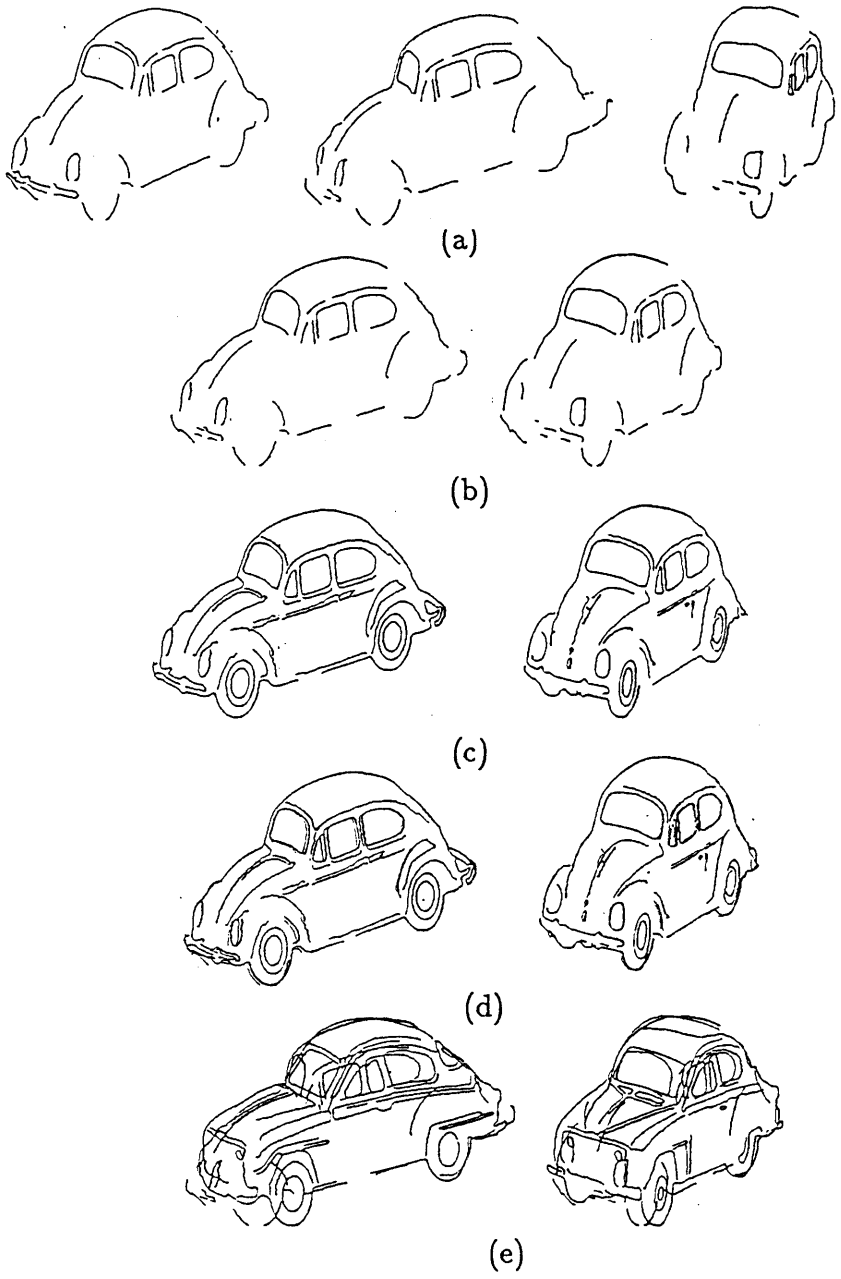

Figure 3: (a) Three model pictures of a VW car for rotations around the vertical axis. The second and the third pictures were obtained from the first by rotations of $\pm 30°$ around the Y-axis. (b) Two linear combinations of the VW model. The x-coefficients are $(0.556, 0.463, -0.018)$ and $(0.582, -0.065, 0.483)$ which correspond to a rotation of the first model picture by $\pm 15°$. These are artificial images, created by linear combinations of the first three views, rather than actual views. (c) Real images of a VW car. (d) Matching the linear combinations to the real images. Each contour image is a linear combination super-imposed on the actual image. The agreement is good within the entire range of $\pm 30°$. (e) Matching the VW model to pictures of the Saab car.

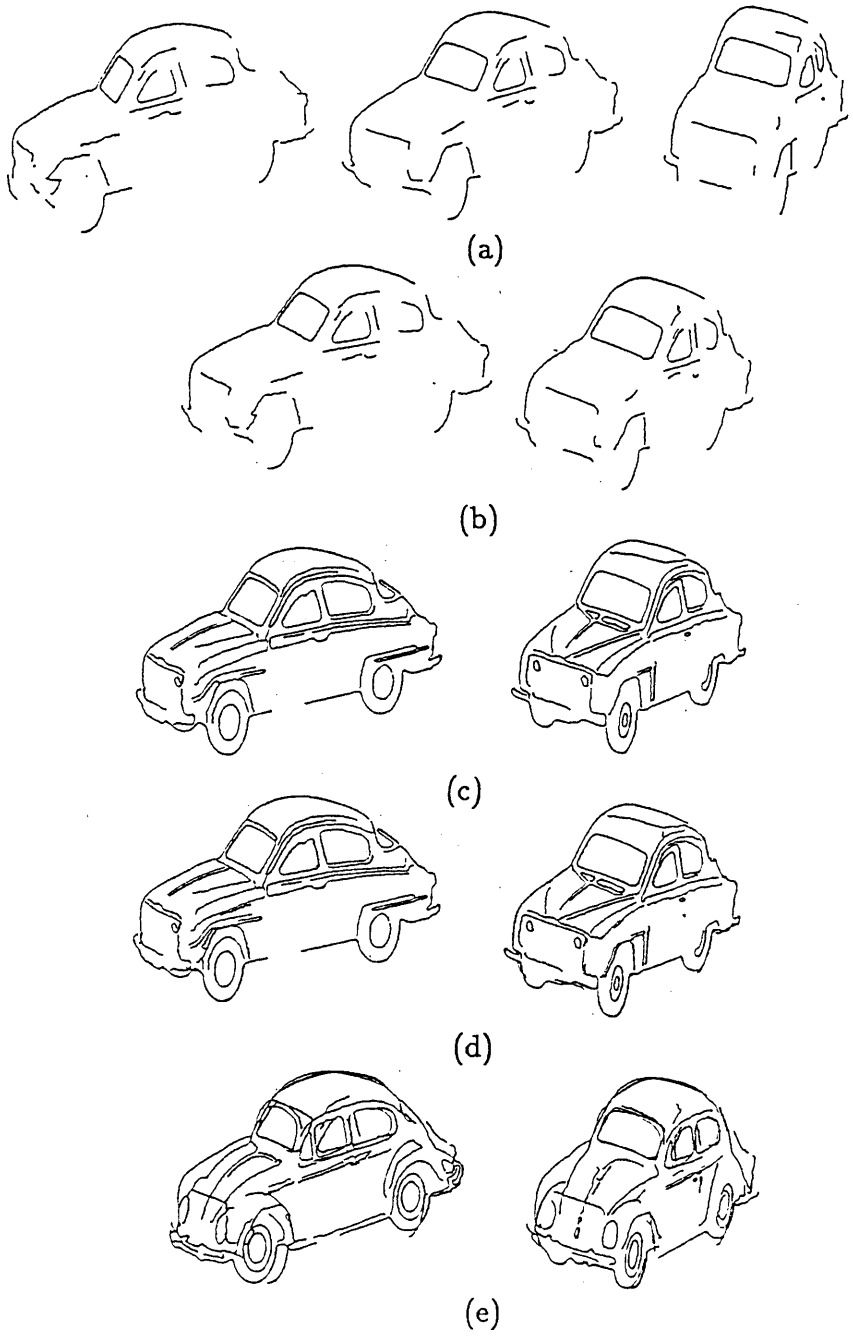

Figure 4: (a) Three model pictures of a Saab car taken with approximately the same transformations as the VW model pictures. (b) Two linear combinations of the Saab model. The x-coefficients are $(0.601, 0.471, -0.072)$ and $(0.754, -0.129, 0.375)$ which correspond to a rotation of the first model picture by $\pm 15°$. (c) Real images of a Saab car. (d) Matching the linear combinations to the real images. (e) Matching the Saab model to pictures of the VW car.

2 Determining the Alignment Coefficients

In the previous sections we have shown that the set of possible views of an object can often be expressed as the linear combination of a small number of views. In this section we examine the problem of determining the transformation between a model and a viewed object. The model is given in this scheme as a set of k corresponding 2-D images $\{M_1, ..., M_k\}$. A viewed object P is an instance of this model if there exists a set of coefficients $\{a_1, ..., a_k\}$ (with a possible set of restrictions $F(a_1, ..., a_k) = 0$) such that:

$$P = a_1 M_1 + ... + a_k M_k \tag{3}$$

In practice we may not obtain a strict equality. We will attempt to minimize, therefore, the difference between P and $a_1 M_1 + ... + a_k M_k$. The problem we face is how to determine the coefficients $\{a_1, ..., a_k\}$. In the following subsections we will discuss three alternative methods for approaching this problem.

2.1 Minimal Alignment: Using a Small Number of Corresponding Features

The coefficients of the linear combination that align the model to the image can be determined using a small number of features, identified in both the model and the image to be recognized. This is similar to previous work in the framework of the alignment approach [Fishler & Bolles 1981, Huttenlocher & Ullman 1987, Lowe 1985, Ullman 1986,1989]. It has been shown that three corresponding points or lines are usually sufficient to determine the transformation that aligns a 3-D model to a 2-D image [Ullman 1986,1989, Huttenlocher & Ullman 1987, Shoham & Ullman 1988], assuming the object can undergo only rigid transformations and uniform scaling. In previous methods, 3-D models of the object were stored. The corresponding features (lines and points) were then used to recover the 3-D transformation separating the viewed object from the stored model.

The coefficients of the linear combination required to align the model views with the image can be derived in principle, as in previous methods, by first recovering the 3-D transformations. They can also be derived directly, however, by simply solving a set of linear equations. This method requires k points to align a model of k pictures to a given image. Therefore, four points are required to determine the transformation for objects with sharp edges, and six points for objects with smooth boundaries. In this way we can deal with any transformation that can be approximated by linear combinations of pictures, without recovering the 3-D transformations explicitly.

The coefficients of the linear conbination are determined by solving the following equations. We assume that a small number of corresponding points (the "alignment

points") have been identified in the image and the model. Let X be the matrix of the x-coordinates of the alignment points in the model. That is, x_{ij} is the x-coordinates of the j'th point in the i'th model-picture. $\mathbf{p}_x$ is the vector of x-coordinates of the alignment points in the image, and $\mathbf{a}$ is the vector of unknown alignment parameters. The linear system to be solved is then $X\mathbf{a} = \mathbf{p}_x$. The alignment parameters are given by $\mathbf{a} = X^{-1}\mathbf{p}_x$ if an exact solution exists. We may use an overdetermined system (by using additional points), in which case $\mathbf{a} = X^{+}\mathbf{p}_x$ (where X^{+} denotes the pseudo-inverse of X). The matrix X^{+} does not depend on the image and can be pre-computed for the model. The recovery of the coefficients therefore requries only a multiplication of $\mathbf{p}_x$ by a known matrix. Similarly, we solve for $Y\mathbf{b} = \mathbf{p}_y$ to extract the alignment parameters $\mathbf{b}$ in the y-direction from Y (the matrix of y-coordinates in the model), and $\mathbf{p}_y$ (the corresponding y-coordinates in the image).

It is also worth noting that the computation can proceed in a similar fashion on the basis of correspondence between straight line segments rather than points. In this case, due to the "aperture problem" [Marr & Ullman 1981], only the perpendicular component (to the contour) of the displacement can be measured. This component can be used, however, in the equations above. In this case each contour segment contributes a single equation (as opposed to a point correspondence, that gives two equations).

One question that may arise in this context is whether the visual system can be expected to extract reliably a sufficient number of alignment features. Two comments are noteworthy. First, this difficulty is not specific to the linear combination scheme, but applies to other alignment schemes as well. Second, although the task is not simple, the phenomenon of apparent motion suggests that mechanisms for establishing feature correspondence do in fact exist in the visual system.

It is interesting to note in this regard that the correspondence established during apparent motion appears to provide sufficient information for the purpose of recognition by linear combinations. For example, when the car pictures in figure 5(a) are shown in apparent motion, the points marked in the left picture appear perceptually to move and match the corresponding points marked in the right picture. These points, with the perceptually established match, were used to align the model and images in figure 5. That is, the coordinates of these points were used in the equations above to recover the alignment coefficients. The model contained six pictures of a Saab car in order to cover all rigid transformations for an object with smooth boundaries. As can be seen, a close agreement was obtained between the image and the transformed model. (The model contained only a subset of the contours, the ones that were clearly visible in all of the different pictures.)

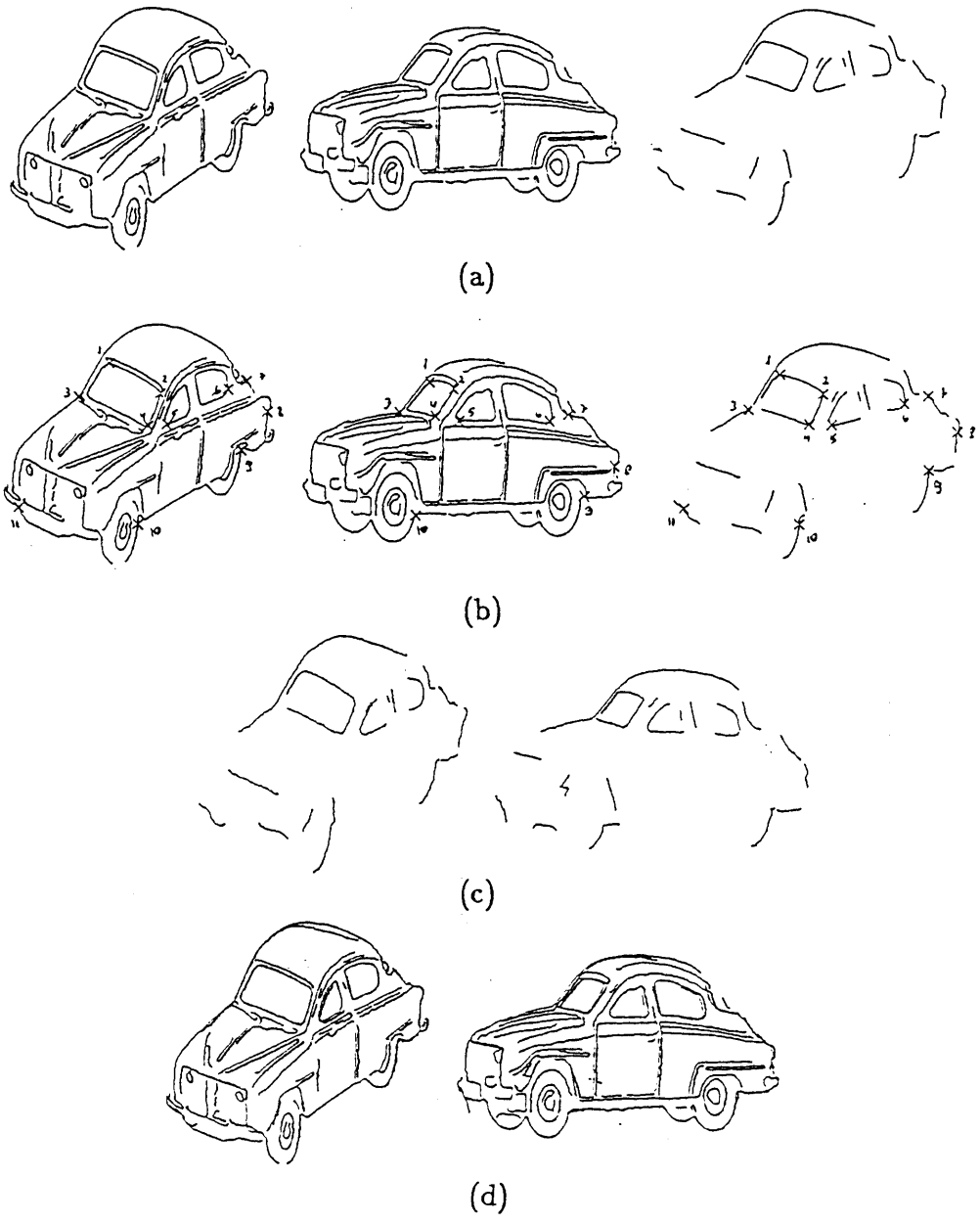

Figure 5: Aligning a model to images using corresponding features. (a) Two images of a Saab car, and one of the six model pictures. (b) The corresponding points used to align the model to the images. The correspondence was determined using apparent motion, as explained in the text. (c) The transformed model. (d) The transformed model super-imposed on the original images.

2.2 Searching for the Coefficients

An alternative method to determine the best linear combination is by a search in the space of possible coefficients. In this method we choose some initial values for the set $\{a_1, ..., a_k\}$ of coefficients, then we apply a linear combination to the model using this set of coefficients. We repeat this process using a different set of coefficients, and take the coefficient values that produced the best match of the model to the image.

The most problematic aspect of this method is that the domain of coefficients might be large, therefore the search might be prohibitive. We can reduce the search space by first performing a rough alignment of the model to the image. The identification of general features in both the image and the model, such as a dominant orientation, the center of gravity, and a measurement of the overall size of the imaged object, can be used for compensating roughly for image rotation, translation and scaling. Assuming that this process compensates for these transformations up to a bounded error, and that the rotations in 3-D space covered by the model are also restricted, then we could restrict the search for the best coefficients to a limited domain. Moreover, the search can be guided by an optimization procedure. We can define an error measure (for instance, the area enclosed between the transformed model and the image) that must be minimized, and use minimization techniques such as gradient descent to make the search more efficient. The preliminary stage of rough alignment may help preventing such methods from reaching a local minimum instead of the global one.

2.3 Linear Mappings

The linear combination scheme is based on the fact that a 3-D object can be modeled by the linear combination of a small number of pictures. That is, the set of possible views of an object is embedded in a linear space of a low dimensionality. We can use this property to construct a linear operator that maps each member of such a space to a predefined vector, which identifies the object. This method is different from the previous two in that we do not recover explicitly the coenfficients $(a_1, ..., a_k)$ of the linear combination. Instead, we assume that a full correspondence has been established between the viewed object and the stored model. We then use a linear mapping to test wether the viewed object is a linear combination of the model views.

Suppose that a pattern P is represented by a vector $\mathbf{p}$ of its coordinates (e.g., $(x_1, y_1, x_2, y_2, ..., x_n, y_n)$). Let P_1 and P_2 be two different patterns representing the same object. We can now construct a matrix L that maps both $\mathbf{p_1}$ and $\mathbf{p_2}$ to the same output vector $\mathbf{q}$. That is $L\mathbf{p_1} = L\mathbf{p_2} = \mathbf{q}$. Any linear combination $a\mathbf{p_1} + b\mathbf{p_2}$ will then be mapped to the same output vector $\mathbf{q}$, multiplied by the scalar $a + b$. We can choose, for

example, $\mathbf{q} = \mathbf{p}_1$, in which case any view of the object will be mapped by L to a selected "canonical view" of it.

We have seen above that different views of the same object can usually be expressed as linear combinations $\sum a_i \mathbf{p_i}$ of a small number of representative views, P_i. If the mapping matrix L is constructed in such a manner that $L\mathbf{p_i} = \mathbf{q}$ for all the views P_i in the same model, then any combined view $\hat{\mathbf{p}} = \sum a_i \mathbf{p_i}$, will be mapped by L to the same $\mathbf{q}$ (up to a scale), since $L\hat{\mathbf{p}} = (\sum a_i)\mathbf{q}$.

L can be constructed as follows. Let $\{\mathbf{p}_1, ..., \mathbf{p}_k\}$ be k linearly independent vectors representing the model pictures (we can assume that they are all linearly independent since a picture that is not is obviously redundant). Let $\{\mathbf{p}_{k+1}, ..., \mathbf{p}_n\}$ be a set of vectors such that $\{\mathbf{p}_1, ..., \mathbf{p}_n\}$ are all linearly independent. We define the following matrices:

$$\begin{aligned} P &= (\mathbf{p}_1, ..., \mathbf{p}_k, \mathbf{p}_{k+1}, ..., \mathbf{p}_n) \\ Q &= (\mathbf{q}, ..., \mathbf{q}, \mathbf{p}_{k+1}, ..., \mathbf{p}_n) \end{aligned}$$

We require that:

$$LP = Q$$

Therefore:

$$L = QP^{-1}$$

Note that since P is composed of n linearly independent vectors, the inverse matrix P^{-1} exists, therefore L can always be constructed.

By this definition we obtain a matrix L that maps any linear combination of the set of vectors $\{\mathbf{p}_1, ..., \mathbf{p}_k\}$ to a scaled pattern $\alpha\mathbf{q}$. Furthermore, it maps any vector orthogonal to $\{\mathbf{p}_1, ..., \mathbf{p}_k\}$ to itself. Therefore, if $\hat{\mathbf{p}}$ is a linear combination of $\{\mathbf{p}_1, ..., \mathbf{p}_k\}$ with an additional orthogonal noise component, it would be mapped by L to $\mathbf{q}$ combined with the same amount of noise.

In constructing the matrix L, one may use more than just k vectors $\mathbf{p}_i$, particularly if the input data is noisy. In this case a problem arises of estimating the best k-dimensional linear subspace spanned by a larger collection of vectors. This problem is treated in Appendix B.

In our implementation we have used $L\mathbf{p}_i = 0$ for all the view vectors $\mathbf{p}_i$ of a given object. The reason is that if a new view of the object $\hat{\mathbf{p}}$ is given by $\sum a_i \mathbf{p}_i$ with $\sum a_i = 0$, then $L\hat{\mathbf{p}} = 0$. This means that the linear mapping L may send a legal view to the zero vector, and it is therefore convenient to choose the zero vector as the common output for all the object's views. If it is desirable to obtain at the output level a canonical view of the object such as $\mathbf{p}_1$ rather than the zero vector, then one can use as the final output the vector $\mathbf{p}_1 - L\hat{\mathbf{p}}$

The decision regarding whether or not $\hat{\mathbf{p}}$ is a view of the object represented by L can be based on comparing $\| L\hat{\mathbf{p}} \|$ with $\| \hat{\mathbf{p}} \|$. If $\hat{\mathbf{p}}$ is indeed a view of the object, then this ratio will be small (exactly 0 in the noise free condition). If the view is "pure noise" (in the space orthogonal to the span of $(\mathbf{p}_1, ...\mathbf{p}_k)$), then this ratio will be equal to 1.

The general idea is somewhat similar to the associative mappings presented in [Kohonen, Oja & Lehtiö 1981]. However, in our scheme, unlike the one presented by Kohonen, Oja & Lehtiö [1981], we take advantage of the fact that intermediate views of 3-D objects can be expressed as the linear combination of model views. Our scheme therefore uses the coordinates of image contours, rather than the image intensity values.

Figure 6 shows the application of the linear mapping to two models of simple geometrical structures, a cube (a) and a pyramid (b). For each model we have constructed a matrix that maps any linear combination of the model pictures to the first picture of the model. The matrices were applied to images (c) and (e), and the results are presented in (d) and (f).

2.4 The Use of Linear Receptive Fields

Two of the three methods above are correspondence-based. They require the identification of corresponding features in the model and the image to be recognized to recover the coefficients or to apply the linear mapping. In this section we suggest a method that may be used (along with some other methods) to alleviate to some degree the problem of establishing a pointwise correspondence.

The goal is to test whether a viewed pattern $\hat{P}$ is a linear combination of patterns in the model, without establishing a pointwise correspondence. To do this we use the following idea. Suppose that, as before, an intermediate view $\hat{P}$ is the linear combination of two views P_1 and P_2 in the model, that is, $\hat{P} = aP_1 + bP_2$. Let us take now an arbitrary group of l corresponding points in P_1, P_2 and $\hat{P}$. Let $a_1, ..., a_l$ denote the l points in pattern P_1, $b_1, ..., b_l$ in P_2 and $c_1, ..., c_l$ in $\hat{P}$. Let us denote by $A_x = \sum_{i=1}^{l} a_{ix}$ (i.e., the sum of the x-coordinates of all the points in $a_1, ...a_l$). Similarly $A_y = \sum_{i=1}^{l} a_{iy}$, $B_x = \sum_{i=1}^{l} b_{ix}$, $B_y = \sum_{i=1}^{l} b_{iy}$, $C_x = \sum_{i=1}^{l} c_{ix}$ and $C_y = \sum_{i=1}^{l} c_{iy}$. From the linear combination, $\hat{P} = aP_1 + bP_2$, it also follows that:

$$\begin{aligned} C_x &= aA_x + bB_x \\ C_y &= aA_y + bB_y \end{aligned}$$

(We have seen above examples in which different coefficients were used for the x- and y-coordinates. Here we have assumed for simplicity that they are identical). This demonstrates that we can use corresponding subsets of points without resolving the individual pointwise correspondence.

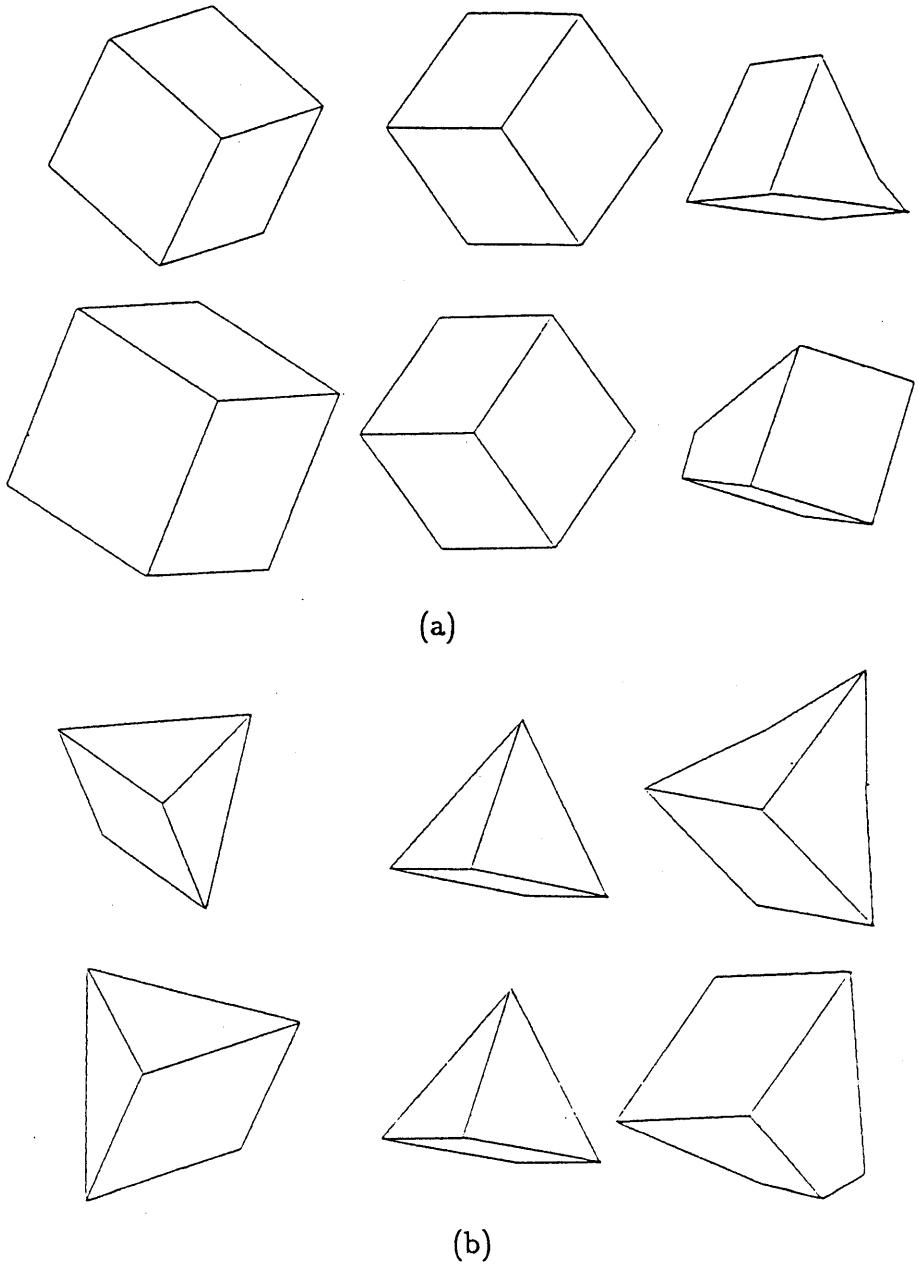

Figure 6: (a) Applying cube and pyramid matrices to the cubes of fig. 2. (b) Applying pyramid and cube matrices to the pyramids of fig. 2. Left column of pictures: the input images. Middle column: the result of applying the appropriate matrix to the images, these results are identical to the first model pictures (which serve as canonical views). Right column: the result of applying the wrong matrix to the images, these results are not similar to the canonical views.

It is worth mentioning that if we match a sufficient number of corresponding subsets of points, the exact point to point correspondence can also be resolved, and the two methods are equivalent. However, the number of subsets may be smaller than the number of points, or we can take subsets of points that are corresponding in most of the points, but not in all of them, and still obtain good results (as shown below).

To use the above idea it becomes necessary to establish a correspondence between subsets of the patterns instead of the individual points. There are several possible ways to approach this problem. Here we propose a simple method, motivated in part by considerations of biological plausibility, that is based on the notion of linear receptive fields.

A linear receptive field (LRF) is an operator that takes a weighted contribution of the points falling within a given region, using a linear weighting function. We will assume here that the LRF response is simply the average contribution of the points falling inside its region. That is, given an image P, the response r is given by $\alpha\bar{x} + \beta\bar{y}$ (for some parameters α, β) where the average is taken over all the points of P falling within the receptive field.

Let us examine the response of an LRF of this type to the model and the viewed object. Let P_1 and P_2 be two pictures in the model set, $\hat{P}$ is the viewed object, and assume that $\hat{P} = aP_1 + bP_2$. Let r_1, r_2 and $\hat{r}$ be the responses of the LRF to P_1, P_2 and $\hat{P}$ respectively. For each pattern, the LRF "sees" only a subset of the points comprising the pattern. The other points fall outside the receptive field. If the points seen by the LRF in P_1, P_2 and $\hat{P}$ are corresponding points (even if the pointwise correspondence is unknown), then it is clear from the considerations above that $\hat{r} = ar_1 + br_2$. In practice, some of the points may not have counterparts inside the LRF, but the relation will hold approximately provided that the majority of points remain within the limits of the receptive field in P_1, P_2 and $\hat{P}$. To obtain this condition it is desirable to: (1) use large receptive fields, and (2) apply some rough alignment, as suggested in section 2.2 above, prior to the match.

We can now proceed along the following line. Let $\mathbf{r} = (r_1, r_2, ..., r_m)$ be an ordered set of LRFs. We define a model to be the result of applying this set $\mathbf{r}$ to each of the model pictures. Given an image I, we first perform a process of rough alignment as described earlier, and denote the result by I'. We apply the set $\mathbf{r}$ to I', and then we check whether the result is a linear combination of the model pictures, that is, we look for a set $\{a_1, a_2, ..., a_k\}$ of coefficients such that for every $1 \leq i \leq m$ it holds that:

$$r_i(I') = \sum_{j=1}^{k} a_j r_i(P_j) \tag{4}$$

Practically, since a strict equality can rarely be achieved, we look for a set $\{a_1, a_2, ..., a_k\}$

of coefficients that minimize the difference between the two terms:

$$\min_{\{a_1 \ldots a_k\}} \| \mathbf{r}(I') - \sum_{j=1}^{k} a_j \mathbf{r}(P_j) \| \tag{5}$$

This problem can be approached, as with the pointwise correspondence, by either computing a pseudo-inverse, or by performing the appropriate linear mapping.

A preliminary stage of rough alignment is required in this scheme to bring each point in the image to lie close to a corresponding position in the model (one of the model pictures). Consequently, each linear receptive field will contain a relatively large proportion of corresponding points. As a result, the application of the set of LRFs to the image will yield approximately a linear combination of the results of applying the same set of LRFs to the model pictures. The justification for this approximation is given in Appendix C. We show there that as the proportion of corresponding points within each LRF increases, the result obtained by the application of this set of LRFs to the image gets closer to a linear combination of the results obtained by applying these LRFs to the model pictures.

The use of linear receptive fields serves in this scheme two distinct purposes. The first is to establish correspondence between subsets of image points, rather than individual points. The second is a conversion between two different types of representations. The linear mapping method assumes that the position of points is given by the numerical values of their x- and y-coordinates. The input image is given, however, in a different representation: a 2-D array of points. The LRF serves to translate the position of a point within the receptive field to a value representing the coordinate of the point. Other conversion schemes are possible, but the LRF is a simple one that also appears to be bilogically palusible. It is interesting to note that cells with linear receptive fields have been described in area 7a of macaque monkeys [Zipser & Andersen 1988]. In Zipser & Andersen's model these cells also serve the roll of converting position in the plane to a firing rate that represents x- or y-coordinate.

3 General Discussion

We have proposed above a method for recognizing 3-D objects from 2-D images. In this method, an object-model is represented by the linear combinations of several 2-D views of the object. It was shown that for objects with sharp edges as well as with smooth bounding contours the set of possible images of a given object is embedded in a linear space spanned by a small number of views. For objects with sharp edges the linear combination representation is exact. For objects with smooth boundaries

it is an approximation that often holds over a wide range of viewing angles. Rigid transformations (with or without scaling) can be distinguished from more general linear transformations of the object by testing certain constraints placed upon the coefficients of the linear combinations.

We have proposed three alternative methods for determining the transformation that matches a model to a given image. The first method uses a small set of corresponding features identified in both the model and the image. Alternatively, the coefficients can be determined using a search. The third method uses a linear mapping as the main step in a scheme that maps the different views of the same object into a common representation.

To avoids the need for pointwise correspondence, we suggested the possible use of linear receptive fields to establish approximate correspondence between subsets of points.

The development of the scheme so far has been primarily theoretical, and initial testing on a small number of objects shows good results. Future work should include more extensive testing using natural objects, as well as the advancement of the theoretical issues discussed below.

In the concluding section we discuss three issues. First, we place the current scheme within the framework of alignment methods in general. Second, we discuss possible extensions. Finally, we list a number of general conclusions that emerge from this study.

3.1 Classes of alignment Schemes

The schemes discussed in this paper fall into the general class of alignment recognition methods. Other alignment schemes have been proposed by Bajcsy & Solina [1987], Chien & Aggarawall [1987], Faugeras & Hebert [1986], Fischler & Bolles [1981], Grimson & Lozano-Perez [1984], Lowe [1985], Thompson & Mundy [1987]. In an alignment scheme we seek for a transformation T_α out of a set of allowed transformations, and a model M from a given set of models, that minimizes a distance measure $d(M, T_\alpha, P)$ (where P is the image of the object). T_α is called the alignment transformation, it is supposed to bring the model M and the viewed object P into an optimal agreement.

The distance measure d typically contains two contributions:

$$d(M, T_\alpha, P) = d_1(T_\alpha M, P) + d_2(T_\alpha)$$

The first term $d_1(T_\alpha M, P)$ measures the residual distance between the picture P and the transformed model $T_\alpha M$ following the alignment, and $d_2(T_\alpha)$ penalizes for the transformation T_α that was required to bring M into a close agreement with P. For example, it may be possible to bring M into a close agreement with P by stretching it

considerably. In this case $d_1(T_\alpha M, P)$ will be small, but, if large stretches of the object are unlikely, $d_2(T_\alpha)$ will be large. We will see below that different classes of alignment schemes differ in the relative emphasis they place on d_1 and d_2.

Alignment approaches can be subdivided according to the method used for determining the aligning transformation T_α. The main approaches used in the past can be summarized by the following three categories.

Minimal alignment. In this approach T_α is determined by a small number of corresponding features in the model and the image. Methods using this approach assume that the set of possible transformations is restricted (usually to rigid 3-D transformations with possible scaling, or a Lie transformation group, [Brockett 1989]), so that the correct transformation can be recovered using a small number of constraints.

This approach has been used by Faugeras & Hebert [1986], Fischler & Bolles [1981], Huttenlocher & Ullman [1987], Shoham & Ullman [1988], Thompson & Mundy [1987], Ullman [1986, 1989]. In these schemes the term d_2 above is usually ignored, since there is no reason to penalize for a rigid 3-D aligning transformation, and the match is therefore evaluated by d_1 only.

The correspondence between features may be guided in these schemes by the labeling of different types of features, such as cusps, inflections, blob-centers, etc. [Huttenlocher & Ullman 1987, Ullman 1989], by using pairwise constraints between features [Grimson & Lozano-Perez 1984], or by a more exhaustive search (as in [Lamdan, Schwartz, & Wolfson 1987], where possible transformations are pre-computed and hashed).

Minimal alignment can be used in the context of the linear combination scheme discussed in this paper. This method was discussed in Section 2.1. A small number of corresponding features is used to determine the coefficients of the linear combination. The linear combination is then computed, and the result compared with the viewed image.

Full alignment. In this approach a full correspondence is established between the model and the image. This correspondence defines a distortion transformation that takes M into P. The set of transformations is not restricted in this approach to rigid transformations. Complex non-rigid distortions are included as well. In contrast with minimal alignment, in the distance measure d above, the first term $d_1(T_\alpha M, P)$ does not play an important role, since the full correspondence forces $T_\alpha M$ and P to be in close agreement. The match is therefore evaluated by the plausibility of the required transformation T_α. Our linear mapping scheme in section 2.3 is a full alignment scheme. A full correspondence is established to produce a vector that the linear mapping can then act upon.

Alignment search. In contrast with the previous approaches, this metod does not use feature correspondence to recover the transformation. Instead, a search is conducted

in the space of possible transformations. The set of possible transformations $\{T_\alpha\}$ is parametrized by a parameter vector α, and a search is performed in the parameter space to determine the best value of α. The deformable template method [Yuille, Cohen, & Hallinan, 1989] is an example for this approach. Section 2.2 described the possibility of performing such a search in the linear combination approach to determine the value of the required coefficients.

3.2 Extensions

The linear combination (LC) recognition scheme is restricted in several ways. It will be of interest to extend it in the future in at least three directions: relaxing the constraints, dealing effectively with occlusions, and dealing with large libraries of objects. We limit the discussion below to brief comments on these three issues.

Relaxing the constraints

The scheme as presented assumes rigid transformation and an orthographic projection. Under these conditions, all the views of a given object are embedded in a low-dimensional linear subspace of a much larger space. What happens if the projection is perspective rather than orthographic, or if the transformations are not entirely rigid? The effect of perspectivity appears to be quite limited. We have applied the LC scheme to objects with ratio of distance-to-camera to object-size down to 4:1, with only minor effects on the results (less then 3% deviation from the orthographic projection for rotations up to 45°).

As for non-rigid transformations, an interesting general extension to explore is where the set of views is no longer a linear subspace, but still occupies a low dimensional manifold within a much higher dimensional space. This manifold resembels locally a linear subspace, but it is no longer "globally straight". By analogy, one can visualize the simple linear combinations case in terms of a 3-D space, in which all the orthographic views of a rigid object are restricted to some 2-D plane. In the more general case the plane will bend, to become a curved 2-D manifold within the 3-D space.

This appears to be a general case of interest for recognition as well as for other learning tasks. For recognition to be feasible, the set of views $\{V\}$ corresponding to a given object cannot be arbitrary, but must obey some constraints, e.g., in the form $F(V_i) = 0$. Under general conditions, these restrictions will define locally a manifold embedded in the larger space. Algorithms that can learn to classify efficiently sets that form low dimensional manifolds embedded in high dimensional spaces will therefore be of general value.

Occlusion

In the linear combination scheme we assumed that the same set of points is visible in the different views. What happens if some of the object's points are occluded by either self-occlusion or by other objects?

As we mentioned in Section 1.3.5 self-occlusion is handled by representing an object not by a single model, but by a number of models covering its different "aspects" [Koenderink & Van Doorn 1979].

As for occlusion by other objects, this problem is handled in a different manner by the minimal alignment and the full alignment versions of the LC scheme. In the minimal alignment version, a small number of corresponding features are used to recover the coefficients of the linear combination. In this scheme, occlusion does not present a major special difficulty. After computing the linear combination, a good match will be obtained between the transformed model the visible part of the object, and recognition may proceed on the basis of this match. (Alignment search will behave in a similar manner.)

In the linear mapping version, an object's view is represented by a vector $\mathbf{v}_i$ of its coordinates. Due to occlusion, some of the coordinates will remain unknown. A way of evaluating the match in this case in an optimal manner is suggested in Appendix D.

Multiple models

We have considered above primarily the problem of matching a viewed object with a single model. If there are many candidate models, a question arises regarding the scaling of the computational load with the number of models.

In the LC scheme, the main problem is in the stage of performing the correspondence, since the subsequent testing of a candidate model is relatively straightforward. The linear mapping scheme is particularly attractive in this regard: once the correspondence is known, the testing of a model requires only a multiplication of a matrix by a vector.

With respect to the correspondence stage, the question is how to perform efficiently correspondence with multiple models. This problem remains open for future study, we just comment here on a possible direction. The idea is to use pre-alignment to a prototype in the following manner. Suppose that $M_1, ..., M_k$ is a family of related models. A single model M will be used for representing this set for the purpose of alignment. The correspondence T_i between each M_i in the set and M is pre-computed. Given an observed object P, a single correspondence $T : M \rightarrow P$ is computed. The individual transformations $M_i \rightarrow P$ are computed by the compositions $T \circ T_i$.

3.3 General conclusions

In this section we summarize birefly a number of general characteristics of the linear combinations scheme. In this scheme, as in some other alignemnt schemes, significant aspects of visual object recognition are more low-level in nature and more pictorial compared with structural description recognition approaches [e.g., Biederman 1985]. The scheme uses directly 2-D views rather than an explicit 3-D model. The use of the 2-D views is different, however, from a simple associative memory [Abu-Mostafa & Psaltis 1987] where new views are simply compared in parallel to all previously stored views. Rather than measuring the distance between the observed object and each of the stored views, a distance is measured from the observed object to the linear subspace, (or a low dimensional manifold) defined by previous views.

The linear combination scheme "reduces" the recognition problem in a sense to the problem of establishing a correspondence between the viewed object and candidate models. The mehtod demonstrates that if a correspondence can be established, the remaining computation is relatively straightforward. Establishing a reliable correspondence between images is not an easy task, but it is a general task solved by the visual system (e.g. in motion measurement and stereoscopic vision), and related processes may also be involved in visual object recognition.

Acknowledgement: We wish to thank E. Grimson, S. Edelman, T. Poggio and A. Yuille for helpful comments, T. Poggio also for his suggestions regarding the use of two views, and A. Yuille for Appendix B.

Appendix A

In section 1.4.2 we showed that the images of an object with smooth surfaces rotating in 3-D space can be represented as the linear combination of five views, and mentioned that the coefficients for these linear combinations satisfy seven functional constraints. In this appendix we list these constraints.

We use the same notation as in section 1.4.2. Let $R_1, ..., R_5, \hat{R}$, be 3×3 rotation matrices, and $R'_1, ..., R'_5, \hat{R}'$ be the corresponding 2×5 matrices defined in section 1.4.2. Let $\mathbf{r}_1, ..., \mathbf{r}_5, \hat{\mathbf{r}}$ be the first row vectors, and $\mathbf{s}_1, ..., \mathbf{s}_5, \hat{\mathbf{s}}$ the second row vectors of $R'_1, ..., R'_5, \hat{R}'$, respectively. In section 1.4.2 we showed that each of the two row vectors of $\hat{R}'$ is a linear combination of the corresponding row vectors of R'_1, R'_2,...,R'_5. That is,

$$\hat{\mathbf{r}} = \sum_{i=1}^{5} a_i \mathbf{r}_i$$

$$\hat{\mathbf{s}} = \sum_{i=1}^{5} b_i \mathbf{s}_i$$

The functional constraints can be expressed as:

$$\hat{r}_1^2 + \hat{r}_2^2 + \hat{r}_3^2 = 1$$

$$\hat{s}_1^2 + \hat{s}_2^2 + \hat{s}_3^2 = 1$$

$$\hat{r}_1\hat{s}_1 + \hat{r}_2\hat{s}_2 + \hat{r}_3\hat{s}_3 = 0$$

$$\hat{r}_1 + \hat{r}_4 = \hat{s}_2 + \hat{s}_5$$

$$\hat{r}_2 + \hat{r}_5 = -(\hat{s}_1 + \hat{s}_4)$$

$$(\hat{r}_1 + \hat{r}_4)^2 + (\hat{s}_1 + \hat{s}_4) = 1$$

$$\hat{r}_4\hat{s}_5 = \hat{s}_4\hat{r}_5$$

(Constraints 1,2,3 and 7 are immediate. Constraints 4,5,6 can be verified by expressing all the entries in terms of the rotation angles α, β, γ.)

To express these constraints as a function of the coefficients, every occurrence of a term $\hat{r}_{ij}$ should be replaced by the appropriate linear combination, as follows:

$$\hat{r}_j = \sum_{i=1}^{5} a_i (r_i)_j$$

$$\hat{s}_j = \sum_{i=1}^{5} b_i (s_i)_j$$

In the case of a similarity transformations (i.e., with scale change) the first two constraints are substituted by:

$$\hat{r}_1^2 + \hat{r}_2^2 + \hat{r}_3^2 = \hat{s}_1^2 + \hat{s}_2^2 + \hat{s}_3^2$$

Appendix B

In this appendix we describe a method to find a space of a given dimension, that lies as close as possible to a given set of points.

Let $\{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$ be a set of points in $\mathcal{R}^n$. We would like to find the $(n-k)$ dimensional space that lies as close as possible (in the least-square sense) to the points

$\{\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m\}$. Let P be the $n \times m$ matrix given by $(\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m)$. Let $\{u_1, ..., u_n\}$ be a set of orthonormal vectors in $\mathcal{R}^n$, and define $\mathcal{U}_k = span\{u_{k+1}, ..., u_n\}$. The sum of the distances (squared) of the points $\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m$ from $\mathcal{U}_k$ is given by:

$$D^2(\mathcal{U}_k) = \sum_{i=1}^{k} \| P^t u_i \|^2$$

(Since $\sum_{i=1}^{k} (\mathbf{p}_i u_i)^2$ is the squared distance of $\mathbf{p}_i$ from $\mathcal{U}_k$.)

Let $F = PP^t$. Then:

$$D^2(\mathcal{U}_k) = \sum_{i=1}^{k} \| P^t u_i \|^2 = \sum_{i=1}^{k} (P^t u_i)^t (P^t u_i) = \sum_{i=1}^{k} u_i^t F u_i$$

Any real matrix of the form XX^t, is symmetric and non-negative. Therefore, F has n eigenvectors and n real non-negative eigenvalues. Assume that the $\{u_1, ..., u_n\}$ above are the eigenvectors of F with eigenvalues $\lambda_1 \leq \lambda_2 \leq ... \leq \lambda_n$ respectively, then $Fu_i = \lambda_i u_i$, and therefore:

$$D^2(\mathcal{U}_k) = \sum_{i=1}^{k} \lambda_i$$

Claim: Let $\{\lambda_1, ..., \lambda_k\}$ be the k smallest eigenvalues of F, then:

$$\sum_{i=1}^{k} \lambda_i = \min_{\mathcal{V}_k} D^2(\mathcal{V}_k)$$

Where the minimum is taken over all the linear subspaces of dimension $n - k$. Therefore, $span\{u_{k+1}, ..., u_n\}$ is the best $(n - k)$ dimensional space through $\mathbf{p}_1, \mathbf{p}_2, ..., \mathbf{p}_m$.

Proof: Let $\mathcal{V}_k$ be a linear subspace of dimension $(n - k)$. We must establish that:

$$D^2(\mathcal{V}_k) \geq D^2(\mathcal{U}_k)$$

Let $\{v_1, ..., v_n\}$ be a set of orthonormal vectors in $\mathcal{R}^n$ such that $\mathcal{V}_k = span\{v_{k+1}, ..., v_n\}$. $V = (v_1, ..., v_n)$, and $U = (u_1, ..., u_n)$ are $n \times n$ orthonormal matrices. Let:

$$R = U^t V$$

Then:

$$UR = V$$

That is:

$$v_j = \sum_{i=1}^{n} r_{ij} u_i$$

R is also orthonormal, therefore:

$$\sum_{i=1}^{n} r_{ij}^2 = \sum_{j=1}^{n} r_{ij}^2 = 1$$

Now:

$$Fv_j = F(\sum_{i=1}^{n} r_{ij} u_i) = \sum_{i=1}^{n} r_{ij} \lambda_i u_i$$

And therefore:

$$v_j^t F v_j = (\sum_{i=1}^{n} r_{ij} u_i)(\sum_{i=1}^{n} r_{ij} \lambda_i u_i)$$

Since $u_i^t u_j = \delta_{ij}$ we obtain that:

$$v_j^t F v_j = \sum_{i=1}^{n} r_{ij}^2 \lambda_i$$

Therefore:

$$D^2(\mathcal{V}_k) = \sum_{j=1}^{k} v_j^t F v_j = \sum_{j=1}^{k} \sum_{i=1}^{n} r_{ij}^2 \lambda_i = \sum_{i=1}^{n} (\sum_{j=1}^{k} r_{ij}^2) \lambda_i$$

Let:

$$\alpha_i = \sum_{j=1}^{k} r_{ij}^2$$

Then:

$$D^2(\mathcal{V}_k) = \sum_{i=1}^{n} \alpha_i \lambda_i$$

Where $0 \leq \alpha_i \leq 1$ and $\sum_{i=1}^{n} \alpha_i = k$.

The claim we wish to establish is that the minimum is obtained when $\alpha_i = 1$ for $i = 1...k$, and $\alpha_i = 0$ for $i = k+1...n$. Assume that for $\mathcal{V}_k$ there exists $1 \leq m \leq k$ such that $\alpha_m < 1$, and $k+1 \leq l \leq n$ such that $\alpha_l > 0$. We can decrease α_l and increase α_m (by $\min(\alpha_l, 1 - \alpha_m)$), and this cannot increase the value of $D^2(\mathcal{V}_k)$. By repeating this process we will eventually reach the value of $D^2(\mathcal{U}_k)$. Since during this process the value cannot increase, we obtain that:

$$D^2(\mathcal{U}_k) \leq D^2(\mathcal{V}_k)$$

And therefore:

$$\sum_{i=1}^{k} \lambda_i = \min_{\mathcal{V}_k} D^2(\mathcal{V}_k)$$

Appendix C

In this appendix we establish that in the method using linear receptive fields the approximation improves with the proportion of corresponding points within each receptive field, and derive a bound on the error. We are given a set of points in the image $\hat{\mathbf{p}} = (\hat{p}_1, ..., \hat{p}_{\hat{n}})$ that fall within a given receptive field, and k sets of model points $\mathbf{p}_1 = (p_{11}, ..., p_{1n_1}), ..., \mathbf{p}_k = (p_{k1}, ..., p_{kn_k})$ that fall within the same receptive field. Let $\bar{\hat{\mathbf{p}}}$ be the average of $\hat{p}_1, ..., \hat{p}_{\hat{n}}$, and $\bar{\mathbf{p}}_i$ the average of $p_{i1}, ..., p_{in_i}$ for every $1 \leq i \leq k$. We next show that the difference between $\bar{\hat{\mathbf{p}}}$ and the linear combination of $\bar{\mathbf{p}}_1, ..., \bar{\mathbf{p}}_k$ is bounded by a term which is proportional to the relative number of corresponding points falling within the receptive field.

Claim: For some given constants $a_1, ..., a_k$, let l be the largest index such that for every $1 \leq j \leq l$ it holds that $\sum_{i=1}^{k} a_i p_{ij} = \hat{p}_j$. Denote $n = \max\{n_1, ..., n_k, \hat{n}\}$, $d = \max_{i,j,k}\{|p_{ij} - p_{ik}|, |\hat{p}_j - \hat{p}_k|\}$ and $q = 1 - \frac{l}{n}$, then:

$$\left|\bar{\hat{\mathbf{p}}} - \sum_{i=1}^{k} a_i \bar{\mathbf{p}}_i\right| \leq qd(1 + \sum_{i=1}^{k} |a_i|)$$

(where d is the diameter of the receptive field).

Proof: Let us first extend the sets of points in such a manner that each will have the same number of points, n. We will do so by setting $p_{ij} = \bar{\mathbf{p}}_i$ for every $1 \leq i \leq k$, $n_i < j \leq n$, and let $\hat{p}_j = \bar{\hat{\mathbf{p}}}$ for every $\hat{n} < j \leq n$. We now have a new set of vectors $\mathbf{p}_1, ..., \mathbf{p}_k, \hat{\mathbf{p}}$ each of length n, all having the same averages they had originally. Therefore:

$$\left|\bar{\hat{\mathbf{p}}} - \sum_{i=1}^{k} a_i \bar{\mathbf{p}}_i\right| = \left|\frac{1}{n}\sum_{j=1}^{n} \hat{p}_j - \sum_{i=1}^{k} \frac{a_i}{n} \sum_{j=1}^{n} p_{ij}\right| =$$

$$= \frac{1}{n}\left|\sum_{j=1}^{n}(\hat{p}_j - \sum_{i=1}^{k} a_i p_{ij})\right| \leq \frac{1}{n}\sum_{j=1}^{n}\left|\hat{p}_j - \sum_{i=1}^{k} a_i p_{ij}\right| =$$

$$= \frac{1}{n}\sum_{j=l+1}^{n}\left|\hat{p}_j - \sum_{i=1}^{k} a_i p_{ij}\right|$$

Now, let $d_{ij} = p_{ij} - p_{i1}$ and $\hat{d}_j = \hat{p}_j - \hat{p}_1$, we obtain:

$$\left|\bar{\hat{\mathbf{p}}} - \sum_{i=1}^{k} a_i \bar{\mathbf{p}}_i\right| \leq \frac{1}{n}\sum_{j=l+1}^{n}\left|\hat{p}_j - \sum_{i=1}^{k} a_i p_{ij}\right| =$$

$$= \frac{1}{n}\sum_{j=l+1}^{n}\left|\hat{p}_1 + \hat{d}_j - \sum_{i=1}^{k} a_i(p_{i1} + d_{ij})\right| = \frac{1}{n}\sum_{j=l+1}^{n}\left|\hat{d}_j - \sum_{i=1}^{k} a_i d_{ij}\right| \leq$$

$$\leq \frac{1}{n} \sum_{j=l+1}^{n} (\left|\hat{d}_j\right| + \sum_{i=1}^{k} |a_i| \, |d_{ij}|) \leq qd(1 + \sum_{i=1}^{k} |a_i|)$$

Therefore, the difference $\left|\bar{\hat{\mathbf{p}}} - \sum_{i=1}^{k} a_i \bar{\mathbf{p}}_i\right|$ is bounded by a term which is proportional to q.

¿From this claim we can conclude the following: Let $\bar{\mathbf{p}}_1, ..., \bar{\mathbf{p}}_k$ be the values obtained by applying a linear receptive field to the pictures of a given model, and let $\bar{\hat{\mathbf{p}}}$ be the value obtained by applying the same LRF to a given image. If the image can be presented as a linear combination of the model pictures, then the error $\left|\bar{\hat{\mathbf{p}}} - \sum_{i=1}^{k} a_i \bar{\mathbf{p}}_i\right|$ is bounded by a term which is proportional to q. Therefore we can in principle reduce this term by reducing q, that is, by constructing the LRF such that it will cover more corresponding points of each picture.

Appendix D

In the linear mapping method a matrix L was constructed that maps every legal view $\mathbf{v}$ of the object to a constant output vector. If the common output is chosen to be the zero vector, then $L\mathbf{v} = \mathbf{0}$ for any legal view of the object.

In this appendix we consider briefly the case where the object is only partially visible. We model this situation by assuming that we are given a partial vector $\mathbf{p}$. In this vector the first k coordinates are unknown, due to the occlusion, and only the last $n - k$ coordinates are observable. (A partial correspondence between the occluded object and the model is assumed to be known.)

In the vector $\mathbf{p}$ we take the first k coordinates to be zero. We try to construct from $\mathbf{p}$ a new vector $\mathbf{p}'$ by supplementing the missing coordinates so as to minimize $\| L\mathbf{p}' \|$. The relation between $\mathbf{p}$ and $\mathbf{p}'$ is:

$$\mathbf{p}' = \mathbf{p} + \sum_{i=1}^{k} a_i \mathbf{u}_i$$

where the a_i are unknown constants, and the $\mathbf{u}_i$ are unit vectors along the first k coordinates.

In matrix notation, we seek to complement the occluded view by minimizing:

$$\min_{\mathbf{a}} \| L\mathbf{p} + LU\mathbf{a} \|$$

Where the columns of the matrix U are the vectors $\mathbf{u}_i$ and $\mathbf{a}$ is the vector on the unknown a_i's.

The solution to this minimization problem is:

$$\mathbf{a} = -[LU]^{+}L\mathbf{p}$$

(where H^{+} denotes the pseudo-inverse of the matrix H). This means that the pseudo-inverse $(LU)^{+}$ will have to be computed. The matrix L is fixed, but U depends on the points that are actually visible.

This optimal value of $\mathbf{a}$ can also be used to determine the output vector of the recognition process $L\mathbf{p}'$:

$$\mathbf{L}\mathbf{p}' = (I - [LU][LU]^{+})L\mathbf{p}$$

$\mathbf{p}$ is then recognized as a legal view if this output is sufficiently close to zero.

References

Abu-Mostafa, Y.S. & Pslatis, D. 1987. Optical neural computing. *Scientific American, 256*, 66-73.

Bajcsy, R. & Solina, F. 1987. Three dimensional object representation revisited. *Proc. of 1st ICCV Conf. (London)*, 231-240.

Basri, R. & Ullman, S., 1988. The alignment of objects with smooth surfaces. *Proc. of 2nd ICCV Conf. (Florida)*, 482-488.

Biederman, I. 1985. Human image understanding: Recent research and a theory. *Computer Vision, Graphics, and Image Processing, 32,* 29-73.

Chien, C.H. & Aggarwal, J.K., 1987. Shape recognition from single silhouette. *Proc. of ICCV Conf. (London)* 481-490.

Brockett, R.W. 1989. Least squares matching problems. *Linear Algebra and its Applications,* 1-17.

Faugeras, O.D. & Hebert, M., 1986. The representation, recognition and location of 3-D objects. *Int. J. Robotics Research*, 5(3), 27-52.

Fischler, M.A. & Bolles, R.C., 1981. Random sample consensus: a paradigm for model fitting with application to image analysis and automated cartography. *Communications of the ACM*, 24(6), 381-395.

Grimson, W.E.L. & Lozano-Perez, T. 1984. Model-based recognition and localization from sparse data. *International Journal of Robotics Research, 3,* 3-35.

Huang, T. S. & Lee, C. H., 1989. Motion and Structure from Orthographic Projections. *IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 2, No. 5,* 536-540.

Huttenlocher, D.P. & Ullman, S., 1987. Object recognition using alignment. *Proc. of ICCV Conf. (London),* 102-111.

Koenderink, J.J. & Van Doorn, A.J., 1979. The internal representation of solid shape with respect to vision. *Biol. Cybernetics 32,* 211-216.

Kohonen, T., Oja, E. & Lehtiö, P., 1981. Storage and processing of information in distributed associative memory systems. *in Hinton, G.E. & Anderson, J.A., Parallel Models of Associative Memory. Hillsdale, NJ: Lawrence Erlbaum Associates,* 105-143.

Lamdan, Y. Schwartz, J.T. & Wolfson, H. 1987. On recognition of 3-D objects from 2-D images. *Courant Institute of Mathematical Sciences, Robotics Technical Report 122.*

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

Marr, D., 1977. Analysis of occluding contour. *Phil. Trans. R. Soc. Lond. B 275,* 483-524.

Marr, D. & Ullman, S., 1981. Directional selectivity and its use in early visual processing. *Proc. R. Soc. Lond. B 211,* 151-180.

Shoham, D. & Ullman, U., 1988. Aligning a model to an image using minimal information. *Proc. of 2nd ICCV Conf. (Florida),* 259-263.

Thompson, D.W. & Mundy J.L., 1987. Three dimensional model matching from an unconstrained viewpoint. *Proc. IEEE Int. Conf. on robotics and Automation,* Raleigh, N.C., 208-220.

Ullman, S., 1979. *The Interpretation of Visual motion.* Cambridge, MA: M.I.T. Press.

Ullman, S. 1983. Recent computational studies in the interpretation of structure from motion. In: A. Rosenfeld & J. Beck (Eds.), *Human and Machine Vision.* New York: Academic Press.

Ullman, S. 1989. Aligning pictorial descriptions: An approach to object recognition: *Cognition, 32(3),* 193-254. Also: 1986, *A.I. Memo 931, The Artificial Intelligence Lab., M.I.T..*

Yuille, A.L., Cohen, D.S. & Hllinan, P.W. 1988. Feature extraction from faces using deformable templates. *Proceedings of Computer Vision and Pattern Recognition, San Diego,* 104-109.

Zipser, D. & Andersen, R.A., 1988. A back-propagation programmed network that simulates response properties of a subset of posterior parietal neurons. *Nature, 331*, 679-684.