Perspective approximations

John Y Aloimonos

In recent years, researchers in computer vision working on problems such as object recognition, shape reconstruction, shape from texture, shape from contour, pose estimation, etc., have employed in their analyses approximations of the perspective projection as the image formation process. Depending on the task, these approximations often yield very good results and present the advantage of simplicity. Indeed when one relates lengths, angles or areas on the image with the respective units in the 3D world assuming perspective projection, the resulting expressions are very complex, and consequently they complicate the recovery process. However, if we assume that the image is formed with a projection which is a good approximation of the perspective, then the recovery process becomes easier. Two such approximations, are described, the paraperspective and the orthoperspective, and it is shown that for some tasks the error introduced by the use of such an approximation is negligible. Applications of these projections to the problems of shape from texture, shape from contour, and object recognition related problems (such as determining the view vector and pose estimation) are also described.

Keywords: perspective approximations, paraperspective, orthoperspective.

An image is the projection of a 3D world onto a 2D image plane. The projection process introduces various distortions of the objects in view, due to the following effects:

1 Distance effect: the objects in view appear larger when they are closer to the image plane.
2 Position effect: the distortion of a pattern depends also on the angle between the line of sight and the image plane, which depends on the position of the pattern.
3 Foreshortening effect: the distortion of a pattern depends on the angle between the surface normal of the plane on which the pattern lies and the line of sight.

Computer Vision Laboratory, Center for Automation Research, Computer Science Department and Institute for Advanced Computer Studies, University of Maryland, College Park, MD 20742-3411, USA

The perspective projection model (see Figure 1), which is the geometrically 'correct' model and can be used as a camera model, captures all three effects. But in the perspective projection, the resulting equations are complicated and often non-linear.

Orthographic projection (see Figure 2a) is simple, and there is some evidence that rays near the fovea are projected orthographically (e.g. see Reference 1). Because of this, orthographic projection has been used in human vision research. But the orthographic projection model captures only the foreshortening effect and ignores the other two. This can lead to two identical contours having the same image, even if one is much further away from the camera than the other. Therefore, methods that use orthographic projection are valid only in a limited domain, where the distance and position effects can be ignored.

Scaled orthographic projection (see Figure 2b) incorporates the forshortening and distance effects, but still ignores the position effect.

In this paper, we describe the paraperspective and the orthoperspective projections. They are approximations of perspective projection that capture all three effects – foreshortening, position and distance – like perspective projection, but at the same time are comparatively simple to work with, like orthographic projection.

Paraperspective projection, or its variants, have been used for the solution of various problems such as shape from texture[2–4], shape from patterns[5], and object recognition[6]. Orthoperspective projection has been used for pose estimation, and was first introduced by Dementhon and Davis[7].

PARAPERSPECTIVE PROJECTION

Let a left-handed coordinate system O, X, Y, Z be fixed with respect to the camera, with the Z axis pointing along the optical axis, and O the nodal point of the eye. We assume that the image plane is perpendicular to the Z axis, and intersects with the Z axis at the point (0, 0, 1) (i.e. the focal length is 1). Consider a plane $Z = pX + qY + c$ in the world, where (p, q) is the gradient of the plane[1]. Under perspective projection, a world point (X, Y, Z) is projected onto the point $(X/Z, Y/Z)$ in the image plane (see Figure 1).

Consider now the following approximation of perspective projection. Let S be a region on the world plane

0262–8856/90/030179–14

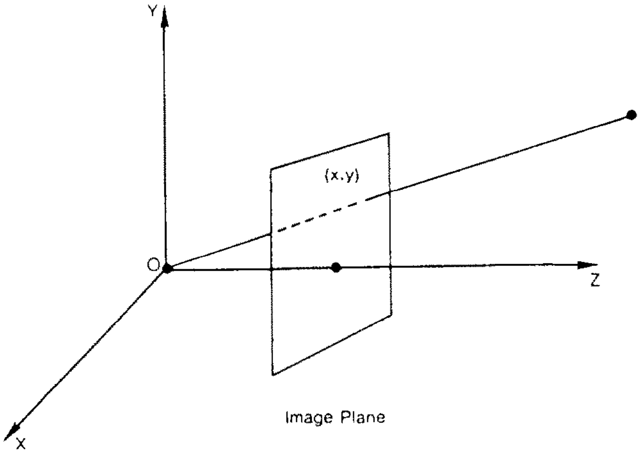


Figure 1. Perspective projection

$Z = pX + qY + c$, and consider the plane $Z = d$, where d is the Z-coordinate of the mass centre of the region S. We approximate perspective projection with the following process (see Figure 3):

(a) First, the region S is projected onto the plane $Z = d$, which is parallel to the image plane and includes the mass centre of the region S. The projection is performed by using the rays that are parallel to the central projecting ray OG, where G is the mass centre of the region S.

(b) The image on the plane $Z = d$ is projected perspectively onto the image plane. Since the plane $Z = d$ is parallel to the image plane, the transformation is a reduction by scaling factor $1/d$. (See Figure 4a, which illustrates the cross-sectional view of the projection process sliced by a plane that includes the central projecting ray and is perpendicular to the X-Z plane.)

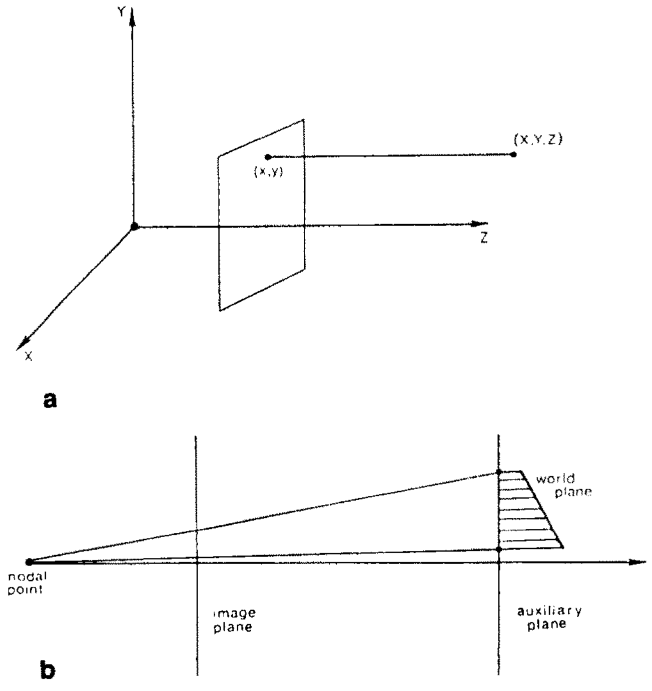


Figure 2. (a) Orthographic projection. (b) Cross-sectional view of the scaled orthographic projection process. The auxiliary plane (parallel to the image plane) should pass through the centre of mass of the planar patch. Here it is in front for clarity

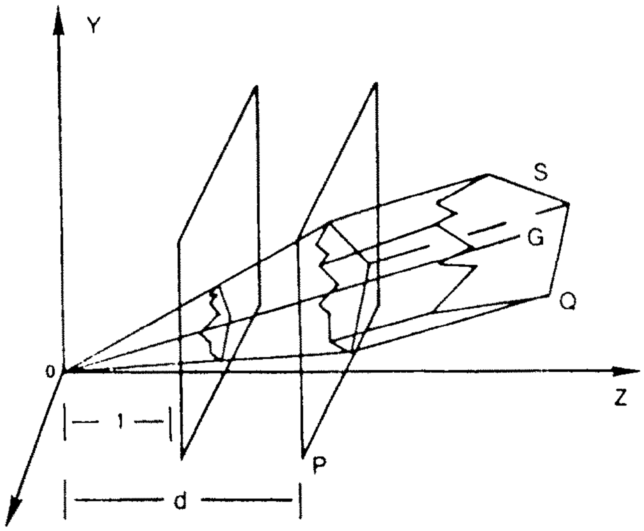


Figure 3. Paraperspective projection – a two-step process

It is clear that this model decomposes the image distortions into two parts: step (a) captures the foreshortening distortion and the position effect, and step (b) captures the distance effect.

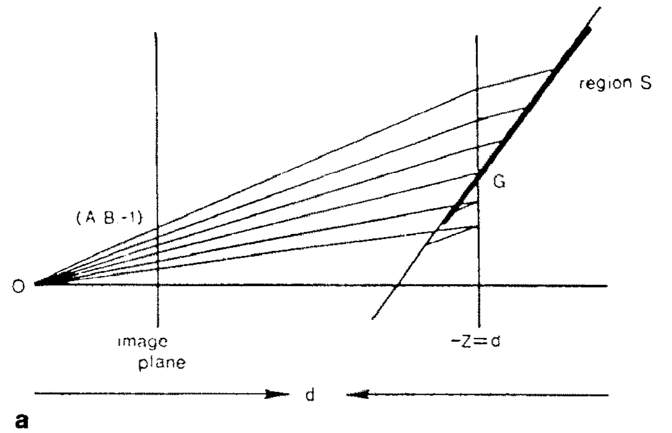
Paraperspective projection is a region-to-region projection – if we consider it only for a single point, it trivially reduces to perspective projection.

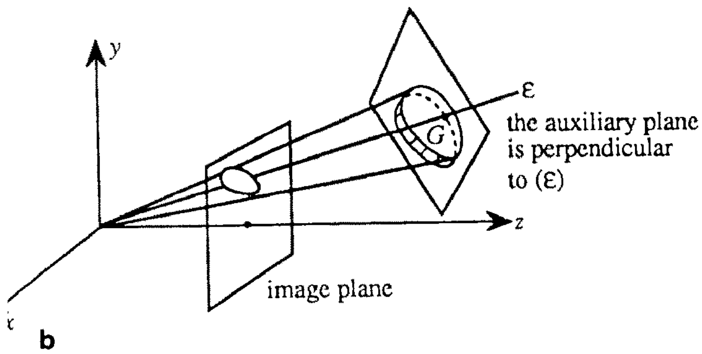
ORTHOPERSPECTIVE

In the case of paraperspective projection, the auxiliary plane passed from the centre of mass of the area under consideration and was parallel to the image plane. It is now obvious that if we change the orientation of the auxiliary plane, then we can obtain a different approximation of the perspective. If the auxiliary plane is perpendicular to the line connecting the focal point of the camera (origin of the camera coordinate system) and the centre mass of the 3D planar patch under consideration, then this kind of paraperspective projection is called orthoperspective (see Figures 4b and 4c for a pictorial description). It is obvious that the accuracy of para- or orthoperspective projection is equivalent to the following sequence of operations: first, a rotation (virtual) of the camera around the centre of projection to make a chosen line of sight coincide with the optical axis (OG in Figures 4b and 4c); then, a scaled orthographic projection of the recentred image, followed by an inverse camera rotation to bring the camera back to its original position.

RELATION BETWEEN IMAGE AND WORLD CHARACTERISTICS

A projection is useful when it is a close approximation to real (perspective) projection and is simple to work with. In image understanding, we are interested in relating properties of the image to 3D properties. Such properties, that frequently arise in motion, contour, shape, stereo and texture analysis, are length and area.





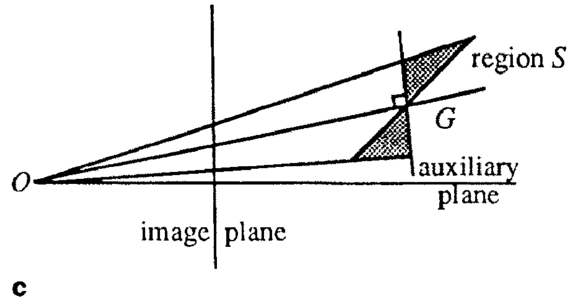


Figure 4. (a) Cross-sectional view of the scaled orthographic projection process-sliced by a plane that includes the central projection ray and is perpendicular to the X–Z *plane. (b) Example orthoperspective. (c) Cross-sectional view of the orthoperspective process. G: centre of mass*

In the rest of this section we study the relationship of lengths and areas in the image with lengths and areas in the world. First, we describe some mathematical prerequisites that have to do with the transformation from the image to the world*.

Let us fix a coordinate system $OXYZ$ with the Z axis as the optical axis and the image plane perpendicular to the Z-axis. If (x, y) is the coordinate system on the image plane (x axis parallel to X, y axis parallel to Y) with origin at the intersection of the Z-axis with the image plane, then a point (X, Y, Z) in the world is projected onto the image point (x, y). We assume that the world point (or contour) under consideration lies on the plane $Z=pX+qY+c$. Throughout this section we assume that the focal length of the camera is 1.

Transformation from image to world

Here we study the transformation from the image to the world, i.e. given a point (x, y) on the image plane we find the point (X, Y, Z) on the world plane. Let f be the inverse imaging function that maps any image point onto the corresponding world point. So, if (x, y) is an image point the 3D world point on the plane $Z=pX+qY+c$ that has (x, y) as its image is given by:

$$(X, Y, Z)=f(x, y)$$

1 *Perspective projection:* under perspective projection the inverse imaging function is given by

$$f(x, y)=\left(\frac{cx}{1-px-qy}, \frac{cy}{1-px-qy}, \frac{c}{1-px-qy}\right)$$

2 *Orthographic projection:* under orthographic projection the inverse imaging function is given by:

$$f(x, y)=(x, y, px+qy+c)$$

3 *Scaled orthographic projection:* under scaled orthographic projection the inverse imaging function is given by:

$$f(x, y)=(dx, dy, pdx+qdy+c)$$

4 *Paraperspective projection:* under paraperspective projection the inverse imaging function is given by:

$$f(x, y)=(dx+tA, dy+tB, d+t)$$

where:

$$d=\frac{c}{1-Ap-Bq} \quad \text{and} \quad t=\frac{d(px+qy-1)+c}{1-pA-qB}$$

5 *Orthoperspective projection:* under orthoperspective projection the inverse imaging function is:

$$f(x, y)=(kx+\lambda A, ky+\lambda B, k+\lambda)$$

$$k=\frac{d(1+A^2+B^2)}{1+Ax+By}, \quad \lambda=\frac{k(px+qp-1)+c}{1-pA-qB}$$

Relation between image and world areas

In this section we relate the area S_W of a world contour to the area S_I of the corresponding image contour.

*We are primarily interested in the transformation from the image to the world, since the image is our input. For the transformation from the world to the image, see Appendix 4.

We know that if we have an area S_I in the image plane, then the corresponding area S_W can be computed directly with the aid of the *first fundamental coefficients*[8], i.e. if we have an area S_I on the image plane, then the area S_W in the world plane $Z = pX + qY + c$ whose projection is S_I is given by:

$$S_W = \int_{S_I}\int \sqrt{EG - F^2}\, dx\, dy \tag{1}$$

with E, F, G the first fundamental coefficients – E, F, G are given by:

$$E = D_x f \cdot D_x f,\ F = D_x f \cdot D_y f,\ G = D_y f \cdot D_y f$$

where f is the inverse image transformation and '$\cdot$' represents inner product of vectors.

1 *Perspective projection:* for perspective projection, the first fundamental coefficients are:

$$E = \frac{c^2}{(1 - px - qy)^4}\,[(1 - qy)^2 + p^2y^2 + p^2]$$

$$F = \frac{c^2}{(1 - px - qy)^4}\,[(1 - qy)\,qx + (1 - px)py + pq]$$

$$G = \frac{c^2}{(1 - px - qy)^4}\,[q^2x^2 + (1 - px)^2 + q^2]$$

Substituting in equation (1) we get:

$$S_W = \int_{S_I}\int \frac{c^2}{(1 - px - qy)^3}\sqrt{1 + p^2 + q^2}\, dx\, dy$$

The above equation cannot be further simplified in the general case, since we do not have a specific area S_I.

2 *Orthographic projection:* in the case of orthographic projection the first fundamental coefficients are:

$$E = 1 + p^2,\ F = pq,\ G = 1 + q^2$$

By substituting in equation (1) we get:

$$S_W = \int_{S_I}\int \sqrt{1 + (p^2)(1 + q^2) - (pq)^2}\, dx\, dy$$

which simplifies to:

$$S_W = S_I\sqrt{1 + p^2 + q^2}$$

3 *Scaled orthography:* for scaled orthography, the first fundamental forms are:

$$E = d^2(1 + p^2)$$
$$F = d^2pq$$
$$G = d^2(1 + q^2)$$

Substituting in equation (1) we get:

$$S_W = \int_{S_I}\int \sqrt{1 + (p^2)(1 + q^2) - (pq)^2}$$

which simplifies to:

$$S_W = S_I d^2\sqrt{1 + p^2 + q^2}$$

The ratio S_W/S_I differs from simple orthography by the addition of a scaling factor d^2.

4 *Paraperspective projection:* for paraperspective projection:

$$E = \frac{c^2}{(1 - Ap - Bq)^4}\,[(1 - Bq)^2 + B^2p^2 + p^2]$$

$$F = \frac{c^2}{(1 - Ap - Bq)^4}\,[Aq(1 - Bq) + (1 - Ap)Bp + pq]$$

$$G = \frac{c^2}{(1 - Ap - Bq)^4}\,[a^2q^2 + (1 - Ap)^2 + q^2]$$

and:

$$S_W = \int_{S_I}\int \frac{c^2}{(1 - Ap - Bq)^3}\sqrt{1 + p^2 + q^2}\, dx\, dy$$

which simplifies to:

$$S_W = S_I\,\frac{c^2\sqrt{1 + p^2 + q^2}}{(1 - Ap - Bq)^3}$$

For an alternative derivation of the same result see Reference 2.

5 *Orthoperspective projection:* in the case of orthoperspective projection the formula for computing areas is quite complicated and does not present any advantage. The same holds true for the case of length computation. (However, see the section below on applications for the advantages of orthoperspective in computations involving lengths in 3D.)

Relation between image and world lengths

In this section we relate the length L_W of a line segment in the world to the length l of the corresponding line segment in the image.

1 *Perspective projection:* again, the desired relation is given directly from the *first fundmental coefficients*. We have:

$$L_W = \int_l \sqrt{E dx^2 + 2F dx\, dy + G dy^2}$$

on the image plane, where E, F, G are the first fundamental coefficients, and in the case of perspective projection are given by:

$$E = \frac{c^2}{(1 - px - qy)^4}\,[(1 - qy)^2 + p^2y^2 + p^2]$$

$$F = \frac{c^2}{(1 - px - qy)^4}\,[(1 - qy)qx + (1 - px)py + pq]$$

$$G = \frac{c^2}{(1 - px - qy)^4}\,[q^2x^2 + (1 - px)^2 + q^2]$$

We can substitute the values of E, F, G but we cannot get rid of the integral if we do not assume a specific line segment l.

2 *Orthographic projection:* let θ be the angle that the line on the image plane makes with the x-axis. The speed of f, the inverse transformation, in the direction $\omega = (\cos\theta,\ \sin\theta)$ is the directional

derivative of f in the direction ω. In particular:

$$f'_\theta(x, y) = Df(x, y)\cdot\omega'$$

or:

$$f'_\theta(x,y) = \begin{matrix} D_1 f_1 & D_2 f_1 \\ D_1 f_2 & D_2 f_2 \\ D_1 f_3 & D_2 f_3 \end{matrix} \quad \begin{matrix} \cos\theta \\ \sin\theta \end{matrix}$$

Also, we have $L_W = \|f'_\theta\|\cdot l$. In the case of orthographic projection, $\|f'_\theta\|$ is equal to

$$\sqrt{1+(p\cos\theta+q\sin\theta)^2}$$

so we get:

$$L_W = l\cdot\sqrt{1+(p\cos\theta+q\sin\theta)^2}$$

3 *Scaled orthography:* We have:

$$\|f'_\theta\| = d\sqrt{1+p^2+q^2}$$

and:

$$L_W = ld\sqrt{1+(p\cos\theta+q\sin\theta)^2}$$

4 *Paraperspective projection:* as in orthographic projection, we have $L_W = \|f'_\theta\|\cdot l$ and:

$$f'_\theta(x, y) = \begin{matrix} D_1 f_1 & D_2 f_1 \\ D_1 f_2 & D_2 f_2 \\ D_1 f_3 & D_2 f_3 \end{matrix} \quad \begin{matrix} \cos\theta \\ \sin\theta \end{matrix}$$

By simple calculation, for perspective projection we get:

$$f'_\theta(x, y) = \frac{c}{(1-Ap-Bq)^2}\cdot \begin{pmatrix} 1-qB & qA \\ pB & 1-pA \\ -p & -q \end{pmatrix} \begin{pmatrix} \cos\theta \\ \sin\theta \end{pmatrix}$$

Simplifying, we get:

$$\|f'_\theta\| = \frac{c\cdot\sqrt{\begin{matrix}[(1-qB)^2+(pB)^2+p^2]\cos^2\theta+ \\ [(1-pA)^2+(qA)^2+q^2]\sin^2\theta+ \\ 2[(1-qB)qA+(1-pA) \\ pB+pq]\sin\theta\cos\theta\end{matrix}}}{(1-Ap-Bq)^2}$$

NEW AREA-PERSPECTIVE INVARIANT

The use of paraperspective projection in area relationships may provide insight into various problems, because of the simplicity of the formulae. This is seen from the following proposition.

Proposition

Let a coordinate system $OXYZ$ be fixed with respect to the left camera, with the Z axis pointing along the optical axis. We assume that the image plane Im_1 is perpendicular to the Z axis at the point (0, 0, 1) and O is the nodal point of the left camera. Let the nodal point of the right camera be (R, L, O), and let its image plane be identical to the previous one, i.e. $\mathrm{Im}_1 = \mathrm{Im}_2$. Consider a polygon Π on the world plane $Z = pX + qY + c$, defined by the points (X_i, Y_i, Z_i), $i = 1, \ldots, n$, and having area S_W. Let S_1, S_2 be the areas of Π under paraperspective projection on the left and right camera images, respectively and S'_1, S'_2 the areas of perspective projections of the polygon Π on the left and right camera images, respectively, i.e. S'_1, S'_2 represent the actual areas in the images. Then:

$$\frac{S_1}{S_2} = \frac{S'_1}{S'_2} = \frac{1-A_1p-B_1q}{1-A_2p-B_2q} \tag{2}$$

Proof

The proof is given in several parts.

Let $(A_1, B_1)$and (A_2, B_2) the centres of mass of the projections of the contour Π on the left and right image planes, respectively (it has to be noted that (A_1, B_1) and (A_2, B_2) are the centres of mass of the actual left and right image contours as opposed to the projections of the centre of mass of Π onto the left and right image planes). As long as they are inside the contours and satisfy some property, they can be used. Then, we have from before, under paraperspective:

$$\frac{S_1}{S_W} = \frac{1}{d^2}\frac{1-A_1p-B_1q}{\sqrt{1+p^2+q^2}} \tag{3}$$

and:

$$\frac{S_2}{S_W} = \frac{1}{d^2}\frac{1-A_2p-B_2q}{\sqrt{1+p^2+q^2}} \tag{4}$$

where d is the depth of the centre of gravity of the world plane (which is the same for both cameras, since the right camera is displaced only along the X and Y axes with respect to the left camera).

We get from equations (3) and (4):

$$\frac{S_2}{S_1} = \frac{1-A_2p-B_2q}{1-A_1p-B_1q} \tag{5}$$

So equation (2) holds under paraperspective. We prove equation (2) to be exact under perspective, i.e. we show:

$$\frac{S'_2}{S'_1} = \frac{1-A_2p-B_2q}{1-A_1p-B_1q}$$

By definition, we have

$$A_1 = \frac{1}{n}\sum(x_i) \quad\text{and}\quad B_2 = \frac{1}{n}\sum(y_i),$$

where x_i, y_i is the image of X_i, Y_i, Z_i $i = 1, \ldots, n$. So we get:

$$A_1 = \frac{1}{n}\sum\left(\frac{X_i}{Z_i}\right) \quad B_1 = \frac{1}{n}\sum\left(\frac{Y_i}{Z_i}\right)$$

Similarly, we get

$$A_2 = \frac{1}{n}\sum\left(\frac{X_i-R}{Z_i}\right),\quad B_2 = \frac{1}{n}\sum\left(\frac{Y_i-L}{Z_i}\right).$$

Using $Z_i = pX_i + qY_i + c$, it is straightforward to show that:

$$\frac{1-A_2p-B_2q}{1-A_1p-B_1q} = 1 + \frac{pR+qL}{c} \tag{6}$$

On the other hand, we can prove that:

$$\frac{S'_2}{S'_1} = 1 + R\frac{\sum\left(\frac{Y_i - Y_{i+1}}{Z_iZ_{i+1}}\right)}{M} + L\frac{\sum\left(\frac{X_{i+1} - X_i}{Z_iZ_{i+1}}\right)}{M} \tag{7}$$

with:

$$M = \sum\left(\frac{X_iY_{i+1} - X_{i+1}Y_i}{Z_iZ_{i+1}}\right) \tag{8}$$

where the indices are taken modulo n, i.e. $n+1$ is actually 1. We can also prove that:

$$\frac{\sum\left(\frac{Y_i - Y_{i+1}}{Z_iZ_{i+1}}\right)}{M} = \frac{p}{c} \tag{9}$$

and:

$$\frac{\sum\left(\frac{X_{i+1} - X_i}{Z_iZ_{i+1}}\right)}{M} = \frac{q}{c} \tag{10}$$

From equations (7), (8), (9) and (10) we get:

$$\frac{S'_2}{S'_1} = \frac{1-A_2p-B_2q}{1-A_1p-B_1q} \quad \text{(q.e.d.).}$$

The image points (A_1, B_1) and (A_2, B_2) may be chosen to be the images of a particular point on the contour or the centroid of the area enclosed by the contour and the theorem still holds. We prove the latter result in Appendix 3.

Using perspective approximations

The perspective approximations described here are useful for the solution of various problems in image understanding. The reason is that many relations between image and world characteristics become very simple when one employs an approximation of the perspective projection and the introduced error is very small. Although analytic results related to the error introduced by the approximation are desirable, in practice it makes sense to talk about the error introduced in a particular application. However, simulations indicate that the biggest factor in the error introducd by para- or orthoperspective is the slant of the planar patch in view. When the slant becomes quite large, the paraperspective approximation is no longer a valid one. The focal length of the camera has no effect on the error. The errors increase as the area increases; for small areas the approximation is much better. This is expected, since as the area tends to 0, paraperspective decreases as c (a measure of the distance of the world plane along the line of sight) increases, the decrease is quite slow in the range in which we might want to use paraperspective projection. Beyond this range the projective geometry is more orthographic than perspective. The error is less when the contour is close to the line of sight. In the case of the ratio of areas error, the error decreases when the two contours are close to each other, since they then tend to be distorted in the same way. Tilt does affect the error. Overall, we see that the distortions due to the *distance effect* and *position effect* are pretty similar for perspective, paraperspective, and orthoperspective, but the projections differ more in the distortions due to the *foreshortening effect*.

Object recognition: determining the view vector

The view vector $\vec{v}$ is the unit vector facing the $+\vec{z}$ direction in the camera frame of reference. Ben-Arie and Meiri[8] determined the view vector using scaled orthography as the camera projection model. Here we show how they could have used paraperspective projection to obtain a more accurate estimation of the view vector. This is important for their work, as they found that in the experiment reported in their paper, a correct match required 1809 node examinations by their search algorithm, whereas when geometric distortion was eliminated only 320 node examinations were required. Ben-Arie and Meiri use an area-based method to determine the view vector. Here we use a similar method, but under paraperspective projection instead of orthographic, to improve their result.

Let three planar surfaces in the world have areas S_{W_1}, S_{W_2}, S_{W_3} and let their images have areas S_{I_1}, S_{I_2}, and S_{I_3}, respectively (see Figure 5). Let the surface normal vectors be $\vec{n}_1$, $\vec{n}_2$ and $\vec{n}_3$, respectively, and let (x_i, y_i) $i = 1, 2, 3$ be the centroids of the image areas. Consider the average:

$$(\bar{x}, \bar{y}) = \frac{1}{2}\left(\sum_{i=1}^{3} x_i, \ \sum_{i=1}^{3} y_i\right)$$

of these centroids. In the coordinate frame of the camera, let its focus be at (0, 0, 0), its image plane at $Z = -1$. Also, let $\vec{v} = (0, 0, -1)$ be the view vector and let:

$$\vec{u} = \frac{(\bar{x}, \bar{y}, -1)}{\sqrt{1+\bar{x}^2+\bar{y}^2}}$$

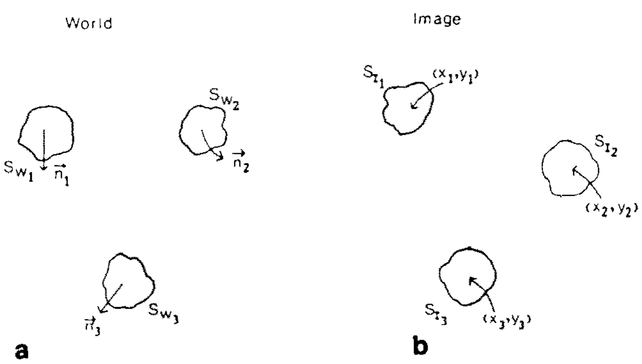
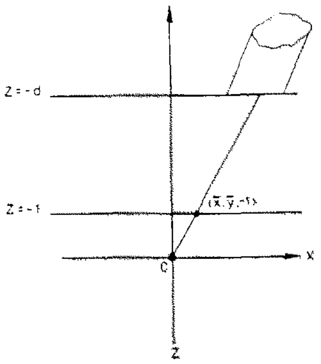


Figure 5. Three planar surface patches (a) and their images (b)



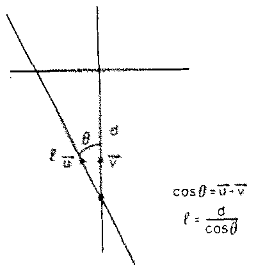


Figure 6. (Left) Cross-sectional view of the paraperspective projection. Figure 7. (Right) Approximate value for the focus of projection F

be the parallel projection vector for the paraperspective projection. Let the world plane of the paraperspective projection be at $Z=-d$ (see Figure 6). Then:

$$\frac{S_{I_1}}{S_{W_1}}=\frac{1}{d^2}\frac{\vec{u}\cdot\vec{n}_1}{\vec{v}\cdot\vec{u}} \tag{12.1}$$

$$\frac{S_{I_2}}{S_{W_2}}=\frac{1}{d^2}\frac{\vec{u}\cdot\vec{n}_2}{\vec{v}\cdot\vec{u}} \tag{12.2}$$

$$\frac{S_{I_3}}{S_{W_3}}=\frac{1}{d^2}\frac{\vec{u}\cdot\vec{n}_3}{\vec{v}\cdot\vec{u}} \tag{12.3}$$

The quantities $\vec{v}\cdot\vec{u}=(1+\bar{x}^2+\bar{y}^2)^{-1/2}$ and S_{W_i}, S_{I_i}, $\vec{n}_i$ $i=1, 2, 3$ are known.

Let $C_i=(\vec{v}\cdot\vec{u})\frac{S_{I_i}}{S_{W_i}}$, $i=1,2,3$ and $\lambda=\frac{1}{d^2}$

Then from equation (12) we have that:

$$C_i=\lambda\vec{u}\cdot\vec{n}_i \quad i=1, 2, 3.$$

But, it can be easily shown (see Appendix 1) that:

$$\vec{u}=k[C_1(\vec{n}_2\times\vec{n}_3)+C_2(\vec{n}_3\times\vec{n}_1)+C_3(\vec{n}_1\times\vec{n}_2)]$$

for some constant k that makes $\vec{u}$ a unit vector, and for which $\vec{u}\cdot\vec{n}_i$ is negative for some choice of i. Also:

$$\lambda=\frac{1}{k[\vec{n}_1, \vec{n}_2, \vec{n}_3]}=\frac{1}{d^2}$$

where $[\cdot,\cdot,\cdot]$ is the triple product.

We haven't yet solved for $\vec{v}$, because we don't know the orientation of the camera frame about the vector $\vec{u}$. However, there are many ways to determine $\vec{v}$ once $\vec{u}$ is known. One such method is described below.

Choose a region i (the one for which $\|(x_i, y_i)-(\bar{x}, \bar{y})\|$ is maximum). Let the centroid of the i^{th} world region be (X_i, Y_i, Z_i). Under paraperspective projection the projection of the centroid of a planar region is the centroid of the projection of the region, since paraperspective projection is an affine transformation. Therefore:

$$(x_i, y_i)=P(X_i, Y_i, Z_i),$$

where $P(X, Y, Z)$ denotes the projection of the point (X, Y, Z). Now calculate an approximate value for the focus of projection $\vec{F}$ (see Figure 7). We have

$$\vec{F}\simeq(\bar{X}, \bar{Y}, \bar{Z})-\frac{d}{\vec{v}\cdot\vec{u}}\cdot\vec{u}$$

where:

$$(\bar{X}, \bar{Y}, \bar{Z})=\frac{1}{3}\sum_{i=1}^{3}(X_i, Y_i, Z_i).$$

Let $\vec{w}$ be a unit vector pointing from the focus to the centroid of the i^{th} region. Then $\vec{w}_i=k[(x_i, y_i, z_i)-\vec{F})$ for some value k that makes $\vec{w}_i$ a unit vector. Now, we can solve for $\vec{v}$ from the equations:

$$\vec{v}\cdot\vec{u}=1-\frac{(\bar{x}^2+\bar{y}^2)^{1/2}}{(1+\bar{x}^2+\bar{y}^2)^{1/2}} \tag{13.1}$$

$$\vec{v}\cdot\vec{u}=1-\frac{(x_i^2+y_i^2)^{1/2}}{(1+x_i^2+y_i^2)^{1/2}} \tag{13.2}$$

and the knowledge that $\vec{v}$ is a unit vector. There will be two solutions in general. The correct solution is the one for which $\vec{F}+d\cdot\vec{v}=(\bar{x}, \bar{y}, \bar{z})$. This concludes the application of paraperspective projection to the determination of the view vector in object recognition.

Shape from contour

We descibe here an application of paraperspective projection to finding the shape of a planar contour from three projections without using any correspondences. This is given by the following proposition.

Proposition

Let a coordinate system $OXYZ$ be fixed, with the Z axis pointing along the optical axis. We assume that the image plane Im_1 is perpendicular to the Z axis at the point $(0, 0, -1)$. Consider a plane Π with equation $Z=pX+qX+c$ in the world, where (p, q) is the gradient of the plane that contains a contour C. Furthermore, we consider two more cameras with image planes Im_2 and Im_3, whose coordinate systems (nodal points) are such that any world point has the same depth with respect to any of the cameras. Then, assuming paraperspective projection of the contour C onto any of the image planes, the images C_1, C_2 and C_3 of the contour on the three cameras are enough to uniquely determine the orientation of the plane Π, without having to solve the point to point correspondence between C_1, C_2, C_3 (see Figure 8).

Proof

Let S_1, S_2, and S_3 be the areas of the contours C_1, C_2 and C_3, respectively. Let also the depth of the centre of gravity of the contour C be β. If S_W is the area of the contour C in the plane Π, and (A_1, B_1), (A_2, B_2) and (A_3, B_3) the centres of gravity of the image con-

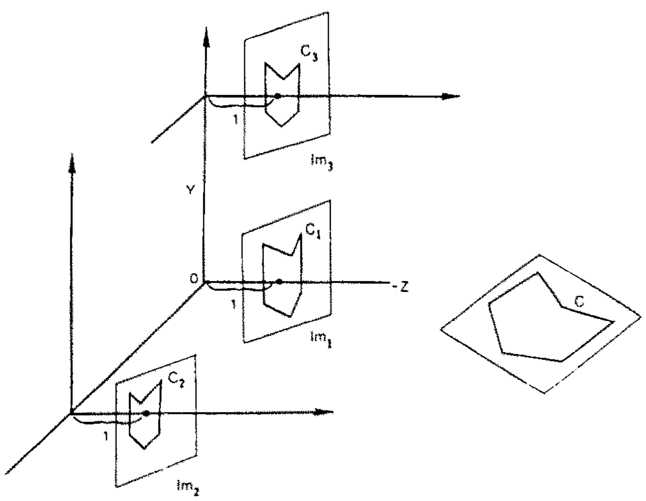


Figure 8. Three camera imaging system

tours C_1, C_2 and C_3, respectively, then the area ratio constraint gives:

$$\frac{S_1}{S_W}=\frac{1}{\beta^2}\frac{1-A_1p-B_1q}{\sqrt{1+p^2+q^2}} \tag{14}$$

$$\frac{S_2}{S_W}=\frac{1}{\beta^2}\frac{1-A_2p-B_2q}{\sqrt{1+p^2+q^2}} \tag{15}$$

$$\frac{S_3}{S_W}=\frac{1}{\beta^2}\frac{1-A_3p-B_3q}{\sqrt{1+p^2+q^2}} \tag{16}$$

Dividing the above equations appropriately, we derive:

$$\frac{S_1}{S_2}=\frac{1-A_1p-2B_1q}{1-A_2p-2B_2q} \tag{17}$$

$$\frac{S_2}{S_3}=\frac{1-A_2p-2B_2z}{1-A_3p-2B_3q} \tag{18}$$

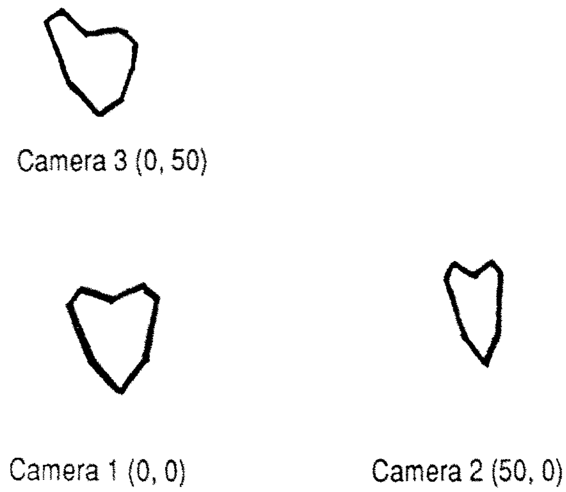
Equations (17) and (18) constitute a linear system with unknowns p and q, which in general has a unique solution (Q.E.D.).

A degenerate case in the solution of the above linear system occurs if and only if the centres of all three image planes are collinear. A proof of this is given in Appendix 2.

Experiments with synthetic images indicate the feasibility of this approach. Figures 9 and 10 describe relevant experiments. The three different projections of a single planar contour on three different image planes (of cameras whose nodal points are not collinear) are shown. In each figure, the actual and the computed gradient of the world contour are described. Note that we take into account discretization effects and that every contour point is considered in calculating the centre of mass. All solutions were calculated from the solution of the system of equations (17) and (18).

Shape from texture

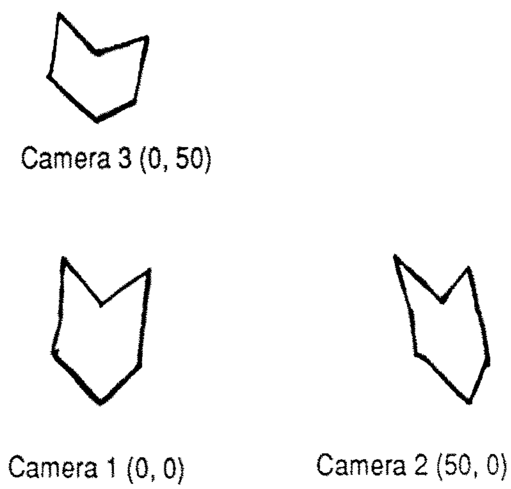
The problem of shape from texture is an important one in vision, and there has been a great deal of good



Actual: p,q = 15.000000 25.000000
Estimated: p,q = 14.999901 24.999964

Figure 9. Finding the orientation of a contour from three projections without correspondences

work in this area (for a complete survey see Reference 3). Here we assume that we are viewing a planar surface that is covered with texels. We assume that the texels have been identified and counted. Then, the problem is to recover the orientation of this planar surface, from the distribution of the texels. We have to make assumptions about the distribution of the texels on the world plane. Otherwise there exist infinitely many solutions for the problem. The assumption we explore here is the 'uniform density' assumption introduced by Gibson in the 1950s[9]. According to this assumption, the texels (the individual elements that constitute the texture) are uniformly distributed on the world plane (in any unit area on the plane there exist approximately the same number of texels), but on the image there is a gradient in the texture. From this *sameness* and *difference* Gibson suggested that we (humans) understand shape from texture.



Actual: p,q = 30.000000 5.000000
Estimated: p,q = 30.000000 4.999972

Figure 10. Finding the orientation of a contour from three projections without correspondences

We do not want to develop a theory for the extraction of shape from texture here. We simply want to explore Gibson's assumption under the paraperspective and perspective projection assumptions, and show how the use of paraperspective benefits us. We could also have chosen the directional isotropy assumption introduced by Witkin[10].

With the imaging system described above (see Figure 1 for perspective and Figure 3 for paraperspective), assume that the planar surface in view, by equation $Z = pX + qY + c$, is covered with uniformly distributed texels. Our goal is to recover the gradient (p, q) of the plane from its image.

Consider two areas S_1 and S_2 on the image plane that contain K_1 and K_2 texels respectively (see Figure 11). Let us denote by $D(S_1)$ and $D(S_2)$ the areas on the 3D plane whose images are areas S_1 and S_2, respectively. Then, the uniform density assumption states that:

$$\frac{K_1}{D(S_1)} = \frac{K_2}{D(S_2)} \tag{19}$$

In the sequel we examine how we can use equation (19) (and other equations similar to it by considering different areas) to recover the gradient.

Under perspective projection we have:

$$D(S_i) = \int_{S_i}\int \frac{C^2}{(1-px-qy)^3} \sqrt{1+p^2+q^2}\,dx\,dy, i=1, 2$$

and equation (19) becomes:

$$\frac{K_1}{\int_{S_1}\int (1-px-qy)^3\,dx\,dy} = \frac{K_2}{\int_{S_2}\int (1-px-qy)^3\,dx\,dy} \tag{20}$$

If we consider another pair of areas then we can construct another equation like equation (20) and then in principle solve for (p, q).

But a closed form solution seems impossible, unless we consider specific areas, to evaluate the integrals. But even if we assume some particular areas to evaluate the integrals involved in equation (20), the resulting equation is of a high degree (degree five if we consider S_1 and S_2 to be squares), and as such difficult to solve. Other clever techniques (e.g. see Reference 4) may be applied, but they result in iterative schemata that may not converge if the initial approximation is not close to the actual value. (A treatment of these techniques can be found in Reference 3.) The point we want to make here is that it seems very hard to find the solution from equations like (20); also, numerical errors will be introduced from the approximation of the derivatives. The paraperspective projection offers a great advantage here, since a closed form solution can be obtained. Indeed, under paraperspective we have:

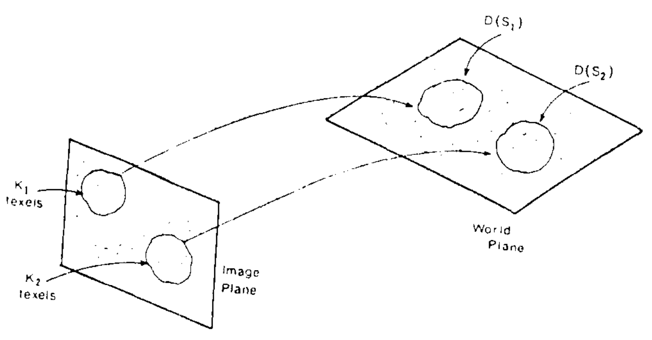


Figure 11. Two areas on the image plane that contain K1 and K2 texels, respectively (see text)

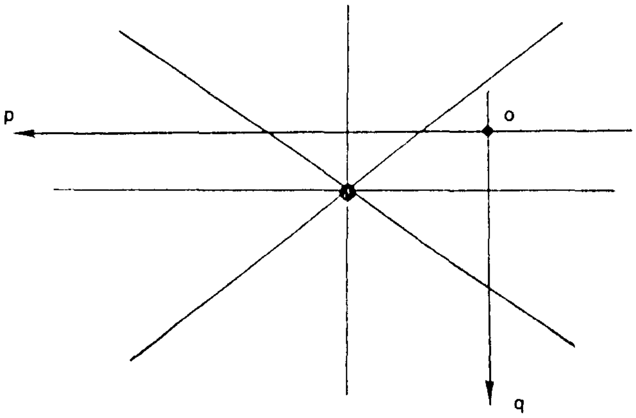


Figure 12. Gradient (p, q) is constrained to lie on the intersection of straight lines (constraints)

$$D(S_i) = \frac{S_i C^2 \sqrt{1+p^2+q^2}}{(1-A_i p - B_i q)^3}, \quad i=1, 2.$$

So, equation (19) becomes:

$$\left[\left(\frac{K_2 S_1}{K_1 S_2}\right)^{1/3} A_2 - A_1\right] p + \left[\left(\frac{K_2 S_1}{K_1 S_2}\right)^{1/3} B_2 - B_1\right] q + \left(\frac{K_2 S_1}{K_1 S_2}\right)^{1/3} - 1 = 0 \tag{21}$$

Equation (21) represents a line in $p-q$ space. So, considering any two regions on the image plane, we constrain the gradient space (see Figure 12). It is now clear that taking two pairs of image regions we can solve for p and q. But because of the errors introduced by paraperspective projection approximation and the sampling process (image digitization and density fluctuations of the regions) we may get inaccurate results. To overcome this problem we can used the least-squares or the Hough transform formalism[1].

Figure 13 is the perspective image of a plane covered with random dots parallel to the image plane. Figure 14 is the image of the previous plane after it is rotated and translated, with tilt = 135° and slant = 30°. The least-squares algorithm developed in this section (equation 21) that is based on paraperspective projection recovered tilt = 134.4° and slant = 29.70°. Figures 15 and 16 show images similar to 14 and 15, respectively, but now the texels are line segments. The algorithm recovered tilt = 133.77° and slant = 30.40°.

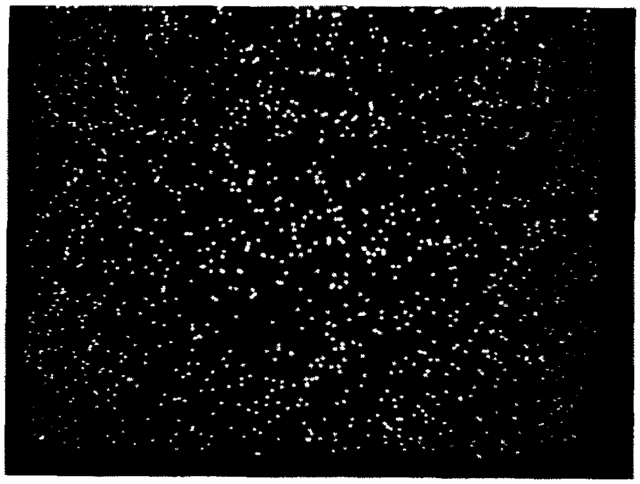


Figure 13. Perspective image of a plane covered with random dots, parallel to the image plane

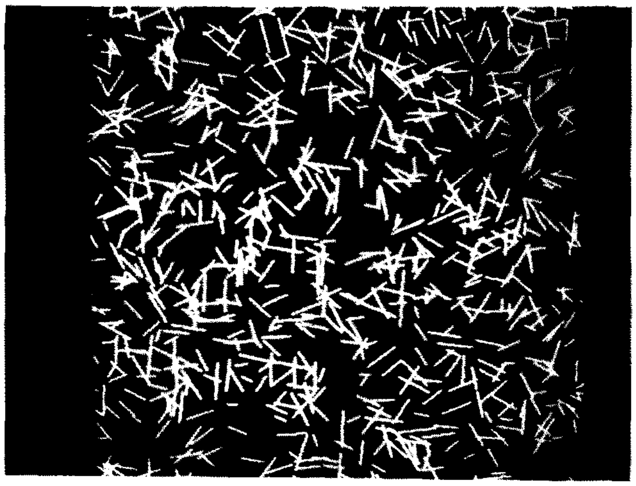


Figure 15. Same as Figure 13, with the texels as line segments

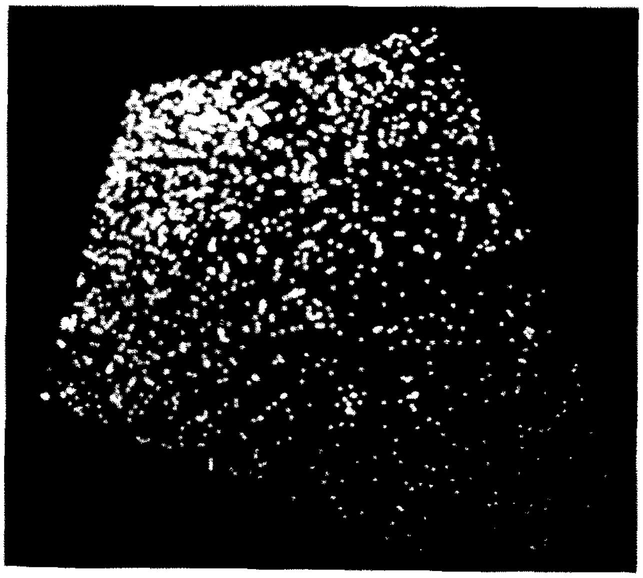


Figure 14. Plane of Figure 13 translated and rotated

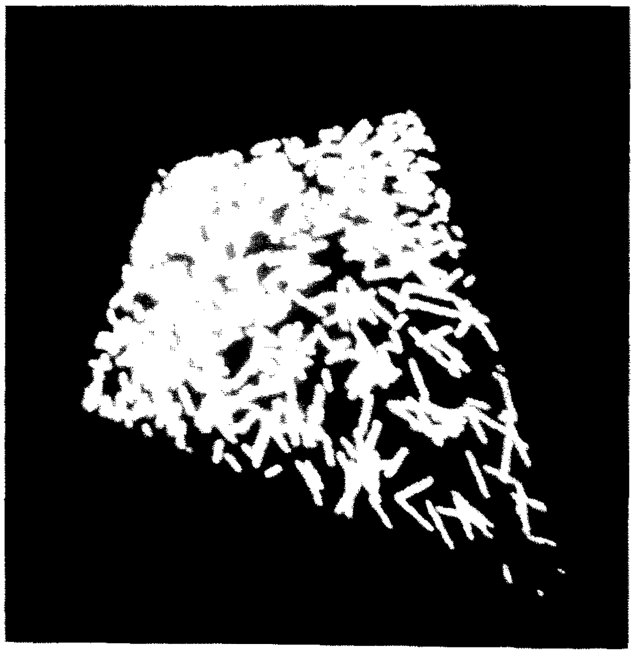


Figure 16. Analog of Figure 14

The accuracy of the above described method for the recovery of shape from texture using the uniform density assumption under paraperspective projection, is due to the fact that equation (19), i.e. the constraint under paraperspective, is not very far from the corresponding constraint under perspective projection. We prove this in the remainder of this section.

We need to prove that if we consider two images S_1, S_2 on the image plane with K_1 and K_2 texels respectively, then the relation:

$$\frac{K_2 S_1}{K_1 S_2} = \left(\frac{1 - A_1 p - B_1 q}{1 - A_2 p - B_2 q}\right)^3$$

(which is true under paraperspective) is approximately true under perspective.

To simplify the analysis, and without loss of generality, we assume that $K_1 = K_2$. This means that the 3D areas are equal. So we prove that if S_1 and S_2 are the projections of two regions R_1, R_2 with the same area, then the relation:

$$\frac{S_1}{S_2} = \left(\frac{1 - A_1 p - B_1 q}{1 - A_2 p - B_2 q}\right)^3$$

is approximately true under perspective projection. Using the nomenclature of the previous sections, let (x, y) denote the coordinates on the image plane and (x', y') the ones on the surface plane. Let also S denote the area of each of the regions R_1 and R_2 and:

$$J = \frac{\partial(x', y')}{\partial(x, y)}$$

the Jacobian of the transformation from the world plane to the image plane. Then:

$$S=\iint\left|\frac{\partial(x',y')}{\partial(x,y)}\right|dx\,dy=\int_{S_1}\int\frac{C_2(1+p^2+q^2)^{1/2}}{(1+px-qy)^3}$$

$$dx\,dy=\int_{S_2}\int\frac{C_2(1+p^2+q^2)^{1/2}}{(1+px-qy)^3}\,dx\,dy$$

So:

$$\int_{S_1}\int\frac{dx\,dy}{(1-px-qy)^3}=\int_{S_2}\int\frac{dx\,dy}{(1-px-qy)^3}\qquad(22)$$

But:

$$E=\frac{1}{(1-\mathrm{p}x-qy)^3}=\frac{1}{(1-pA_1-qB_1-p\tilde{x}-q\tilde{y})^3}$$

where:

$$x=A_1+\tilde{x}\quad y=B_1+\tilde{y}$$

If we expand E (Taylor series) around (A_1, B_1), we get, dropping all higher order terms:

$$E\simeq\frac{1}{(1-pA_1-qB_1)^3}+\frac{3\tilde{x}p}{(1-pA_1-qB_1)^4}+$$

$$\frac{3\tilde{y}q}{(1-pA_1-qB_1)^4}$$

So:

$$\int_{S_1}\int\frac{dx\,dy}{(1-px-qy)^3}\simeq\int_{S_1}\int\frac{1}{(1-pA_1-qB_1)^4}$$

$$(1-pA-qB+3p\tilde{x}+3q\tilde{y})\,\mathrm{d}\tilde{x}\,d\tilde{y}$$

or:

$$\int_{S_1}\int\frac{dx\,dy}{(1-px-qy)^3}\simeq\frac{S_1}{(1-pA_1-qB_1)^3}$$

Similarly:

$$\int_{S_1}\int\frac{dx\,dy}{(1-px-qy)^3}=\frac{S_1}{(1-pA_1-qB_2)^3}$$

So, from equation (22):

$$\frac{S_1}{S_2}\simeq\left(\frac{1-A_1p-B_1q}{1-A_2p-B_2q}\right)^3$$

under perspective projection.

Solving the three-point perspective problem

Model-based pose estimation of 3D objects from a single view is an important problem in computer vision. A commonly used technique consists of locating points of interest on models of the objects, detecting these points in the image and matching subsets of these image points against subsets of the model points. If we then know that a given subset of points of an object is projected on the image into a given subset of points, we have constraints on the object location in space. How many points should we take as subsets? Solving for the position and orientation of an object knowing the images of n points at known locations on the object is called the n-point perspective problem. Many researchers have considered three point solutions (for a survey see Reference 7), because this is the smallest subset which yields a finite number of object poses (generally two).

The drawback to solving the triangle pose problem with exact perspective is that it is slow, requiring many floating point operations. The speed of computation of the triangle pose is of importance in the performance of an object recognition system, since it is performed for many or all possible combinations of triples of image and model points. Some researchers have proposed simpler computational solutions of perspective[7], but it has been established that para- or orthoperspective provide the best solutions. The three-point perspective problem amounts to the following: the points p_0, p_1, p_2 are the known images of world points P_0, P_1, P_2 with known relative 3D positions, but unknown positions along their lines of sight. We know P_0, P_1 has length l_1 and P_0, P_2 has length l_2. Also the angle $P_0P_2=\alpha$ (known). The pose of the triangle is then determined by the angles ϕ_1 and ϕ_2 that P_0P_1 and P_0P_2 form with the line of sight OP (see Figures 17 and 18). Under perspective projection the solution comes from solving the system[7]:

$$\cos\alpha=\sin\phi_1\,\sin\phi_2\cos\phi+\cos\phi_1\cos\phi$$

$$S_1:\ \frac{\sin(\phi_1-\gamma_1)}{\sin(\phi_2-\gamma_2)}=k\quad\text{and}\quad k=\frac{\sin(\gamma_1/l_1)}{\sin(\gamma_2/l_2)}$$

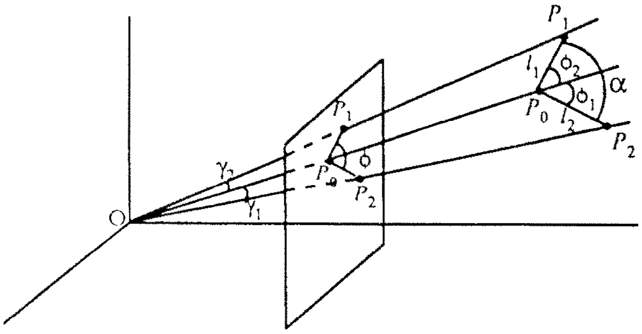


Figure 17. Three-point perspective (perspective projection)

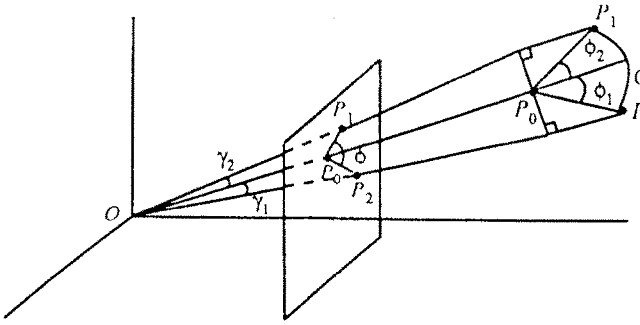


Figure 18. Three-point perspective (orthoperspective projection)

where ϕ is the angle between the planes Op_0p_0 and Op_0p_2, while under orthoperspective the solution come from solving the system[7]:

$$S_2: \quad \begin{aligned} &\cos\alpha = \sin\phi_1 \sin\phi_2 \cos\phi + \cos\phi_1 \cos\phi \\ &\frac{\sin\phi_1}{\sin\phi_2} = k \quad \text{and} \quad k = \frac{\tan\gamma_1/l_1}{\tan\gamma_2/l_2} \end{aligned}$$

System S_1 is quite hard to solve as it results in an equation of degree 8 in $\sin(\phi-\gamma_1)$. However, S_2 results in a simple quadratic and the accuracy of the solution is high.

CONCLUSION

Approximations of the perspective projection (geometric aspect of image formation) have been presented. Among them, the paraperspective and orthoperspective approximations stand out for their power (closeness of approximation) and simplicity. It has been demonstrated that various problems in image understanding are facilitated through the use of such approximations. Examples include object recognition problems (determining the view vector, pose estimation), shape from texture and shape from contour, and in general, problems that could benefit from perspective approximations seem to be the ones related to reconstruction of parameters of the outside world from image characteristics.

ACKNOWLEDGEMENTS

Thanks go to Barbara Burnett for her help in the preparation of this paper.

REFERENCES

1 **Horn, B K P** *Robot Vision* McGraw-Hill, UK (1986)
2 **Ohta, Y, Maenobu, K and Sakai, T** 'Obtaining surface orientation from texels under perspective projection' *Proc. Int. Joint. Conf. Artif. Intell.* (1981) pp 746–751
3 **Aloimonos, J** 'Shape from texture' *Biol. Cybern.* Vol 58 No 3 (1988) 345–360
4 **Aloimonos, J** *Computing intrinsic images* PhD Thesis, Department of Computer Science, University of Rochester, NY, USA (1987)
5 **Aloimonos, J and Swain, M** 'Shape from patterns: regularization' *Int. J. Comput. Vision* Vol 2 No 1 (1988) 171–187
6 **Connolly, C I, Mundy, J L, Stenstrom, J R and Thompson, D W** 'Matching from 3D range models into 2D intensity scenes' *Proc. 1st IEEE Int. Conf. Comput. Vision* London, UK (1987) pp 65–79
7 **Dementhon, D and Davis, L S** *New exact and approximate solutions to the three-point perspective problem* CAR-TR-471, Computer Vision Laboratory, University of Maryland, MD, USA (1989)
8 **Ben-Arie, J and Meiri, A Z** '3D object recognition by optimal matching search of multinary relation graphs' *Comput. Vision, Graph. & Image Process.* Vol 37 No 2 (1987) pp 345–361
9 **Gibson, J J** *The Perception of the Visual World* Houghton Mifflin, Boston, MA, USA (1950)
10 **Witkin, A** 'Recovering surface shape and orientation from texture' *Artif. Intell.* Vol 17 No 1 (1981) pp 17–45
11 **Lipschutz, G** *Differential Geometry,* Schaum's Outline Series (1969)

APPENDIX 1

Following the nomenclature detailed above in the section on determining the view vector, and assuming that $C_i = \lambda\vec{u}\cdot\vec{u}_i$, $i = 1, 2, 3$, then:

(a) $\vec{u} = k\,[C_1(\vec{n}_2\times\vec{n}_3) + C_2(\vec{n}_3\times\vec{n}_1) + C_3(\vec{n}_1\times\vec{n}_2)]$ with k a constant that makes $\vec{u}$ a unit vector and for which $\vec{u}\cdot\vec{n}_i$ is negative for some choice of i.

(b) $\lambda = \dfrac{1}{k[\vec{n}_1, \vec{n}_2, \vec{n}_3]}$

where $[\cdot, \cdot, \cdot]$ is the triple vector product.

Proof

The proof is immediate by eliminating λ from equations $C_i = \lambda\vec{u}\cdot\vec{u}_i$, assuming that $[\vec{u}_1, \vec{u}_2, \vec{u}_3] = \vec{u}_1(\vec{u}_2\times\vec{u}_3) \neq 0$, i.e. the vectors $\vec{u}_1$, $\vec{u}_2$ and $\vec{u}_3$ are not coplanar.

APPENDIX 2

Following the nomenclature of the Proposition in the above section on shape from contour, and assuming that the centres of the three cameras are the points (0, 0), (dx_2, dy_2) and (dx_3, dy_3), respectively, the following is true: the system of equations (17) and (18) is degenerate if and only if the points (0, 0), (dx_2, dy_2) and (dx_3, dy_3) are collinear.

Proof

Let the 3D plan be described by the equation $Z = pX + qY + c$ (with respect to the first camera (of nodal point (0, 0). Equations (17) and (18) are equivalently written as:

$$\frac{S_1}{S_2} = \frac{C}{C - pdx_2 - qdy_2} \tag{A1}$$

$$\frac{S_2}{S_3} = \frac{C - pdx_2 - qdy_2}{C - pdx_3 - qdy_3} \tag{A2}$$

The above system (A1, A2) is equivalent to the system (17, 18). It can be written as $(p\ q)A = \bar{b}$, with:

$$A = \begin{matrix} dx_2\dfrac{S_1}{S_2} & dx_3\dfrac{S_2}{S_3} - dx_2 \\ dy_2\dfrac{S_1}{S_2} & dy_3\dfrac{S_2}{S_3} - dy_2 \end{matrix} \quad \text{and} \quad \bar{b} = c \begin{matrix} \dfrac{S_1}{S_2} - 1 \\ \dfrac{S_2}{S_3} - 1 \end{matrix}$$

A necessary and sufficient condition for the above system to have a unique solution, is $\det(A) \neq 0$ or:

$$\frac{S_1}{S_3}(dx_2\, dy_3 - dy_2\, dx_3) \neq 0 \text{ or}$$

$$\frac{dx_2}{dy_2} \neq \frac{dx_3}{dy_3},$$

i.e. (0, 0), (dx_2, dy_2), (dx_3, dy_3) are not collinear (Q.E.D.).

APPENDIX 3

Proposition

Let a coordinate system $OXYZ$ be fixed with respect to the left camera, with the Z axis pointing along the optical axis. Let the nodal point of the right camera be the point $(Sx1, Sy, O)$, and let its image be plane parallel to the previous one, at $Z=1$. Consider a region Ω in the world plane $Z = pX + qY + c$ having area S_W. Let S_1, S_2 be the areas of the perspective projection of Ω in the left and right cameras, respectively, and let A_1, A_2 be their respective centroid. Then we have:

$$\frac{S_2}{S_1} = \frac{1 - pA_2 - qB_2}{1 - pA_1 - qB_1}$$

Proof

If (x', y') denote the coordinates on the surface plane and (x_i, y_i) $i = 1, 2$ the coordinates on image planes 1 and 2, respectively, and assuming that the foci of the two cameras are at the points (0, 0) and (S_x, S_y), respectively, with coinciding image planes $z = -1$, we have:

$$(A_i, B_i) = \frac{\int_{\Omega_i} (x_i, y_i)\, dx_i\, dy_i}{\int_{\Omega_i} dx_i, dy_i} \quad i = 1, 2$$

where Ω_i $i = 1, 2$ the two contours. Consequently:

$$(A_i, B_i) = \int_{\Omega_i} (x_i, y_i) \left| \frac{\partial(x_i, y_i)}{\partial(x', y')} \right| dx'\, dy' \quad 1 = 1, 2$$

where $J_i = \dfrac{\partial(x_i, y_i)}{\partial(x', y')}$ $i = 1, 2$

the Jacobian of the transformation from the world to each one of the image planes. We consider only cases in which the orientation of the plane may be continuously transformed to $p = q = 0$ without passing through an orientation where it appears edge-on in one of the cameras, i.e. both cameras are looking at the front of the plane, not the back. In this case, the J_is are positive, and we may drop the absolute value signs. So:

$$J_i = \frac{(1 - px_i - qy_i)^3}{c_i^2(1 + p^2 + q^2)^{1/2}} \quad i = 1, 1$$

with $c_1 = c$, $c_2 = c + pS_x + qS_y$. Note that since:

$$-z = \frac{c_1}{1 - px_1 - qy_1} = \frac{c_2}{1 - px_2 - qy_2},$$

we have:

$$J = \frac{c_i}{(-z)^3(1 + p^2 + q^2)^{1/2}}$$

So:

$$\frac{1 - pA_2 - qB_2}{1 - pA_1 - qB_1} = \frac{1 - p\dfrac{\int_\Omega x_2 J_2\, dx'\, dy'}{\int_\Omega J_2\, dx'\, dy'} - q\dfrac{\int_\Omega y_2 J_2\, dx'\, dy'}{\int_\Omega J_2\, dx'\, dy'}}{1 - p\dfrac{\int_\Omega x_1 J_1\, dx'\, dy'}{\int_\Omega J_1\, dx'\, dy'} - q\dfrac{\int_\Omega y_1 J_1\, dx'\, dy'}{\int_\Omega J_1\, dx'\, dy'}}$$

or:

$$\frac{1 - pA_2 - qB_2}{1 - pA_1 - qB_1} = \frac{\int_\Omega J_1\, dx'\, dy' \int_\Omega (1 - px_2 - qy_2) J_2\, dx'\, dy'}{\int_\Omega J_2\, dx'\, dy' \int_\Omega (1 - px_1 - qy_1) J_1\, dx'\, dy'}$$

or:

$$\frac{1 - pA - qB_2}{1 - pA_1 - qB_1} = \frac{\dfrac{c_1}{(1 + p^2 + q^2)^{1/2}} \int \left(\dfrac{1}{-z}\right)^3 dx'\, dy' \int \dfrac{c_2}{(-z)} \dfrac{c_2}{(1 + p^2 + q^2)^{1/2}} \dfrac{1}{(-z)}\, dx'\, dy'}{\dfrac{c_2}{(1 + p^2 + q^2)^{1/2}} \int \dfrac{1}{(-z)^3}\, dx'\, dy' \int \dfrac{c_1}{(-z)} \dfrac{c_1}{(1 + p^2 + q^2)^{1/2}} \dfrac{1}{(-z)^3}\, dx'\, dy'}$$

or:

$$\frac{1 - pA_2 - qB_2}{1 - pA_1 - qB_1} = \frac{c_2}{c_1} = \frac{c - pS_2 + qS_y}{c} \tag{A3}$$

On the other hand:

$$S_1 = \int_{\Omega} \left| \frac{\partial(x_i, y_i}{\partial(x', y')} \right| dx' \, dy'$$

$$S_2 = \int_{\Omega} \left| \frac{\partial(x_2, y_2)}{\partial(x', y')} dx' \, dy'$$

Also:

$$J_1 = \frac{c}{(-z)^3(1+p^2+q^2)^{1/2}} \quad \text{and}$$

$$J_2 = \frac{c+pS_x+qS_y}{(-z)^3(1+p^2+q^2)^{1/2}}$$

So:

$$\frac{S_2}{S_1} = \frac{c+pS_x+qS_y}{c} \frac{\int \left(-\frac{1}{z}\right)^3 dx' \, dy'}{\int \left(-\frac{1}{z}\right)^3 dx' \, dy'} \quad \text{or}$$

$$\frac{S_2}{S_1} = \frac{c+pS_x+qS_y}{c} \tag{A4}$$

Equations (A7) and (A2) prove the claim.

APPENDIX 4

Here x, y denote retinal coordinates, X, Y, Z world coordinates, and A, B, d, parameters of the paraperspective projection (see Figure 4).

1 *Perspective projection:* under perspective projection we have:

$$x = \frac{X}{Z}, \quad y = \frac{Y}{Z}$$

2 *Orthographic projection:* under orthographic projection:

$$x = X, \quad y = Y$$

3 *Scaled orthographic projection:* under scaled orthography:

$$x = \frac{X}{d}, \quad y = \frac{Y}{d}$$

4 *Paraperspective projection:* under paraperspective projection:

$$x = \frac{X-(Z-d)A}{d}$$

$$y = \frac{Y-(Z-d)B}{d}$$