A Comparison of Two and Three Dimensional Imaging

Ernest Hall, Donald Rosselot, Mark Aull and Manohar Balapa
Center for Robotics Research
University of Cincinnati
Cincinnati, OH 45221-0072
Email: Ernie.Hall@uc.edu
WWW: http://www.robotics.uc.edu/

ABSTRACT

Three dimensional visual recognition and measurement are important in many machine vision applications. In some cases, a stationary camera base is used and a three-dimensional model will permit the measurement of depth information from a scene. One important special case is stereo vision for human visualization or measurements. In cases in which the camera base is also in motion, a seven dimensional model may be used. Such is the case for navigation of an autonomous mobile robot. The purpose of this paper is to provide a computational view and introduction of three methods to three-dimensional vision. Models are presented for each situation and example computations and images are presented. The significance of this work is that it shows that various methods based on three-dimensional vision may be used for solving two and three dimensional vision problems. We hope this work will be slightly iconoclastic but also inspirational by encouraging further research in optical engineering.

Keywords: Three dimensional vision, stereo imaging, human visualization, perspective, projective model

1. INTRODUCTION

"The discovery of nature, of the ways of planets, plants and animals, required first the conquest of common sense. Science would advance, not by authenticating everyday experience, but by grasping paradox, adventuring into the unknown... Nothing could be more obvious than that the earth is stable and unmoving, and that we are the center of the universe."

D.J. Boorstin (1983)[1] in Hall and Hall (1985)

"Physicists who work with a concept called string theory envision our universe as an eerie place with at least nine spatial dimensions, six of them hidden from us, perhaps curled up in some way so they are undetectable. The big question is why we experience the universe in only three spatial dimensions instead of four, or six, or nine. Two theoretical researchers from the University of Washington and Harvard University think they might have found the answer. They believe the way our universe started and then diluted as it expanded – what they call the relaxation principle – favored formation of three- and seven-dimensional realities. The one we happen to experience has three dimensions."[2]

The purpose of this paper is not to argue the laws of physics or string theory, but rather to present the three and seven degree of freedom models for what we normally consider as two dimension vision and show that these models are useful in machine vision imaging applications. One application of this model is 3D tracking in computer graphics as described in[3]. Another application that has motivated much of our research is 3D tracking for mobile robots[4] and [5] .

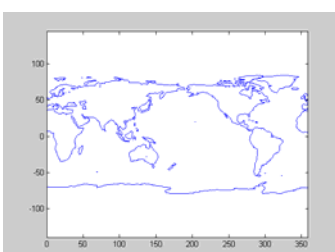


Figure 1 Two dimensional images from Matlab demo examples. CONTOUR creates a contour plot of the Earth from topographic data. The contour is based on points on the map that have zero altitude.

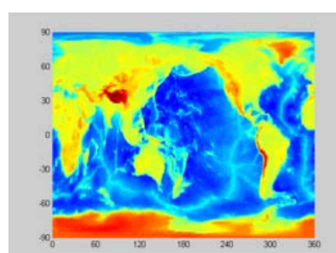


Figure 2 IMAGE creates a 2-D image plot from the data in topo and topomap1. The elevation information in this figure in shown by pseudo color. Altitudes correspond to shades of green, while depths (below sea level) correspond to shades of blue. MATLAB® 7.0.1[6]

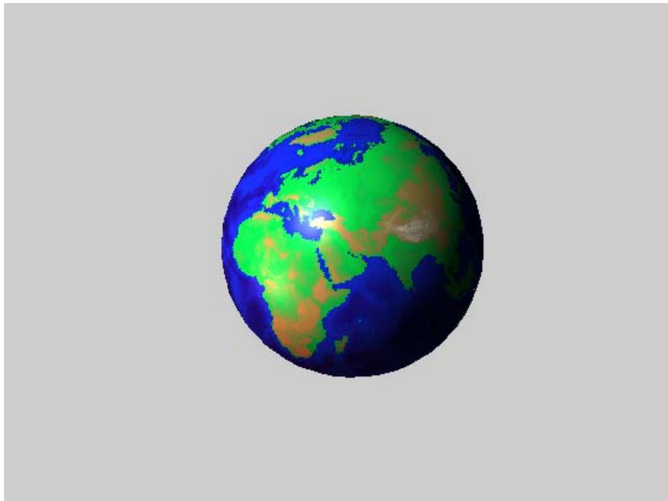


Figure 3. The SPHERE function returns x,y,z data that are points on the surface of a sphere (50 points in this case). Observe the altitude data in topo mapped onto the coordinates of the sphere contained in x, y and z. Two light sources illuminate the globe. MATLAB® 7.0.1 [6]

Is the earth flat or round? The example shown in Figures 1-3 is from a Matlab[1] demo program “Earth's Topography” using data available from the National Geophysical Data Center, NOAA US Department of Commerce under data announcement 88-MGG-02.[6]The data is stored in a MAT file called topo.mat. The variable topo contains the altitude data for the Earth. topomap1 contains the colormap for the altitude. Figure 3 shows exactly the same data as Figure 2 but mapped to a sphere and with two light sources.

2. TWO and THREE DIMENSIONAL MAPPINGS

2.1 Earth

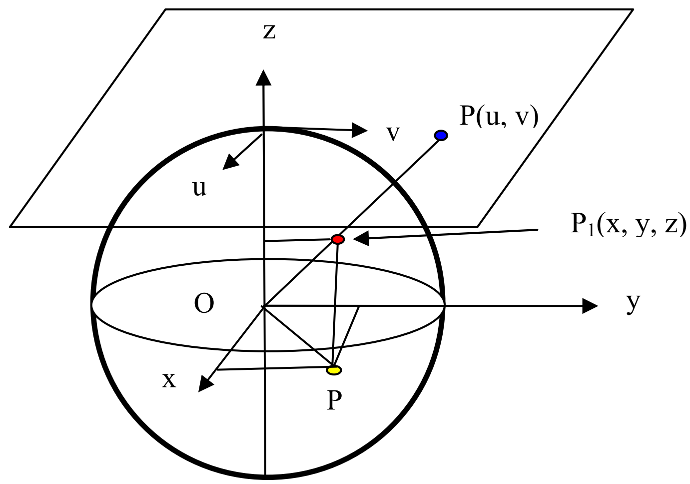
The earth’s surface is a good place to begin to understand the mappings that are involved with producing Figures 1-3. The flat earth question is natural and has been argued for thousands of years. One reason for the confusion is provided by the Mathworld[7] description of a manifold[8].

> *“A manifold is a topological space that is locally Euclidean (i.e., around every point, there is a neighborhood that is topologically the same as the open unit ball in $\mathbb{R}^n$). To illustrate this idea, consider the ancient belief that the Earth was flat as contrasted with the modern evidence that it is round. The discrepancy arises essentially from the fact that on the small scales that we see, the Earth does indeed look flat. In general, any object that is nearly "flat" on small scales is a manifold, and so manifolds constitute a generalization of objects we could live on in which we would encounter the round/flat Earth problem, as first codified by Poincaré . “*

This definition was found to be more controversial than convincing to many so a numerical example was also developed. A very simplified model of the earth may be considered as a sphere shown in Figure 4 that shows a spherical surface of unity radius with a three dimensional Cartesian coordinate frame centered at the origin of the sphere with x axis increasing out of the paper, the y axis increasing to the right and the z axis increasing upward. Also consider a portion of an infinite plane called the (u, v) plane that is tangent to the surface of the sphere at point (0, 0, 1). This plane is also shown by a two dimensional Cartesian coordinate frame with a u axis pointing in the same direction as the x axis and the perpendicular v axis in the same direction as the y axis. A point $P_1(x, y, z)$ on the top hemispherical surface of the sphere and a corresponding point P(u, v) in the tangent plane are also shown. Note that another point on the bottom half hemisphere also can be located along the same line and introduces ambiguity so this derivation must be restricted to the top hemisphere.

---

[1] Matlab is a registered trademark of the Math Works



Point $P_1(x, y, z)$ is on the surface of the unit sphere described by $x^2 + y^2 + z^2 = 1$ and point P (u, v) is the corresponding point on the plane.

Figure 4. A spherical surface and tangent plane with corresponding points $P_1(x, y, z)$ on the surface of the unit sphere and point P(u, v) in the tangent plane.

The question is: how do the coordinates of a point on the three dimensional spherical surface (x, y, z) relate to the corresponding point in the two dimensional tangent plane at (u, v)?

A cross sectional view of the x-z plane is shown in Figure 5 that shows the points $P_1(x,y,z)$ and P(u,v) as co-linear, i.e. on the same line that originates from the origin. Using this relationship, it can be seen that the triangle from the origin, O, to P_1 and point (0,y,z) is similar to the triangle from the origin, O, to P(u, v) and point (0,0,1). Since the triangles are similar, the following relationships hold. For any point such as $P_1(x, y, z)$ on the surface of the unit radius sphere:

$$x^2 + y^2 + z^2 = 1 \tag{1}$$

And for the corresponding point P(u, v) along the ray from the origin O passing through $P_1(x, y, z)$ to the intersection with the tangent plane, there is a similar triangle relationship as shown in Figure 5.

$$\frac{u}{1} = \frac{x}{z} \text{ and } \frac{v}{1} = \frac{y}{z} \tag{2}$$

These equations can be solved by first squaring the first two equations:

$$u^2 = \frac{x^2}{z^2} \text{ and } v^2 = \frac{y^2}{z^2} \text{ and } z^2 = 1 - x^2 - y^2 \tag{3}$$

then substituting the relationships

$$\begin{aligned} z^2 &= 1 - u^2 * z^2 - v^2 * z^2 \\ z^2 &= \frac{1}{(1 + u^2 + v^2)} \\ z &= \frac{1}{\sqrt{u^2 + v^2 + 1}} \end{aligned} \tag{4}$$

So the transformations are

$$x = \frac{u}{\sqrt{u^2 + v^2 + 1}} \quad (5)$$

$$y = \frac{v}{\sqrt{u^2 + v^2 + 1}} \quad (6)$$

$$z = \frac{1}{\sqrt{u^2 + v^2 + 1}} \quad (7)$$

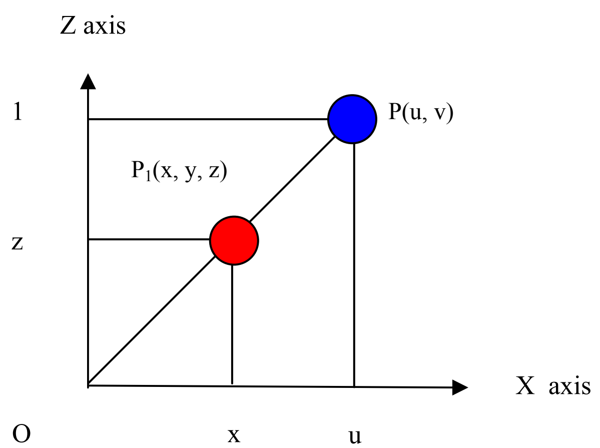


Figure 5. Co-linear points, P and P_1 form similar triangles.

These equations show that there is a one to one mapping between three dimensional points $P_1(x, y, z)$ on the top hemispherical surface of the sphere and corresponding points two dimensional points, P(u, v), on the infinite tangent plane.

The range of values of (x, y, z) is restricted by the spherical surface. However, the points, (u, v) in the plane may extend throughout the entire plane defined by z=1. That is, the points on the top hemispherical surface defined implicitly by Equation 1, $x^2+y^2+z^2=1$ must be in the range $-1 \le x \le 1$, $-1 \le y \le 1$ and $0 \le z \le 1$, while the points on the tangent plane may be in the range,

$$-\infty < u < \infty, \quad -\infty < v < \infty \quad (8)$$

When z=0, the surface equation reduces to that of a circle (equatorial circle)

$$x^2 + y^2 = 1$$
$$\text{and} \quad (9)$$
$$u = \infty \text{ and } v = \infty$$

So given a point (x, y, z) on the spherical surface, the corresponding point (u, v) in the tangent plane may be found by the forward transformations:

$$u = \frac{x}{z}, \quad v = \frac{y}{z} \quad (10)$$

And given a point on the tangent plane (u, v), the corresponding point (x, y, z) on the spherical surface may be found by the inverse transformation. If the radius of the sphere is not unity, similar mapping equations may be developed. Also, the mapping may also be developed for an offset radial vector and given zenith and azimuth angles.

2.4 Transformation equations using spherical trigonometry

The transformation involved is called a Gnomonic projection by cartographers such as Snyder[9]. Snyder's description is as follows:

"A point at a given angular distance from the chosen center point on the sphere is plotted on the Gnomonic projection at a distance from the center proportional to the trigonometric tangent of that angular direction, and at its true azimuth, or

$$\rho = R * \tan(c)$$
$$\theta = \pi - Az = 180 - Az$$
$$h' = \frac{1}{\cos^2(c)} \qquad (11)$$
$$k' = \frac{1}{\cos(c)}$$

where c is the angular distance of the given point from the center of projection, Az is the azimuth easy of north, and θ *is the polar coordinate east of south. The term k' is the scale factor in a direction perpendicular to the radius from the center of the map, not along the parallel except on the polar aspect. The scale factor h' is measured in the direction of the radius. Combining with standard equations, the formulas for rectangular coordinates of the oblique Gnomonic projection are as follows, given*

given

$R, \phi_1, \lambda_0, \phi, and \ \lambda$

to find x and y

$$x = R * k' * \cos(\phi) * \sin(\lambda - \lambda_0)$$
$$y = R * k' * (\cos(\phi_1) \sin(\phi) - \sin(\phi_1) * \cos(\phi) * \cos(\lambda - \lambda_0)$$

where

$$k' = \frac{1}{\cos(c)}$$
$$\cos(c) = \sin(\phi_1) * \sin(\phi) + \cos(\phi_1) * \cos(\phi) * \cos(\lambda - \lambda_0)$$

and

(ϕ_1, λ_0) are latitude and longitude of the projection center and origin.

(12)

The Y axis coincides with the central meridian λ_0 *, y increasing northerly. The meridians are straight lines, but the parallels are conic sections for which the eccentricity is*

$$eccentricity = \cos(\phi_1)/\sin(\phi) \qquad (13)$$

(If the eccentricity is zero, for $\phi_1 = \pm 90^\circ$, they are circles. *If the eccentricity is less than 1, they are ellipses; if equal to 1, a parabola; if greater than 1, a hyperbolic arc.)"*

For the north polar Gnomonic, letting $\phi_1 = 90^\circ$

$$x = R * \cot(\phi) * \sin(\lambda - \lambda_0)$$
$$y = -R * \cot(\phi) * \cos(\lambda - \lambda_0) \qquad (14)$$

In polar coordinates,

$$\rho = R * \cot(\phi)$$
$$\theta = \lambda - \lambda_0 \qquad (15)$$

For the south polar Gnomonic, with $\phi_1 = -90^\circ$

$$x = -R * \cot(\phi) * \sin(\lambda - \lambda_0)$$
$$y = R * \cot(\phi) * \cos(\lambda - \lambda_0) \quad (16)$$

In polar coordinates,

$$\rho = -R * \cot(\phi)$$
$$\theta = \pi - \lambda + \lambda_0 \quad (17)$$

For the equatorial Gnomonic, letting $\phi_1 = 0$

$$x = R * \tan(\lambda - \lambda_0)$$
$$y = R * \frac{\tan(\phi)}{\cos(\lambda - \lambda_0)} \quad (18)$$

In automatically computing a general set of coordinates for a Gnomonic map, the above equations should be used to reject points equal to or greater than 90 degrees from the center. That is, if cos(c) is zero or negative, the point is to be rejected. If cos(c) is positive, it may or may not be plotted depending on the desired limits of the map.

For the inverse formulas for the sphere, to find

ϕ and λ

given

R, ϕ_1, λ_0, x and y

$$\phi = \sin^{-1}(\cos(c) * \sin(\phi_1) + \frac{y * \sin(c) * \cos(\phi_1)}{\rho} \quad (19)$$

If ρ = 0, the equations are indeterminate, but

$\phi = \phi_1$, *and* $\lambda = \lambda_0$

If ϕ_1 is not $\pm 90^\circ$

$$\lambda = \lambda_0 + \tan^{-1}(\frac{x * \sin(c)}{(\rho * \cos(\phi_1) * \cos(c) - y * \sin(\phi_1) * \sin(c)}) \quad (20)$$

If $\phi_1 = 90^\circ$

$$\lambda = \lambda_0 + \tan^{-1}(\frac{x}{(-y)}) \quad (21)$$

If $\phi_1 = -90^\circ$

$$\lambda = \lambda_0 + \tan^{-1}(\frac{x}{y}) \quad (22)$$

In the above equations

$$\rho = \sqrt{x^2 + y^2}$$
$$c = \tan^{-1}(\frac{\rho}{R}) \quad (23)$$

A table of numerical data is given in Table 26, p. 168, Snyder[9] and is a useful computational aid.

To interpretation the parameters, consider the following example.

$$\begin{aligned}&\text{Latitude and longitude of the projection center } (\phi_1, \lambda_0),\ \phi_1 = 0 \text{ and } \lambda_0 = 0\\&\text{Latitude } \phi\\&\text{Longitude } \lambda\\&\text{Radius R} = 1.0\end{aligned} \tag{24}$$

For example, with R=1.0, λ = 0, ϕ = π*80/180, then x = 0, y=R*tan(ϕ)/cos(λ) = 5.671.

3. THREE DIMENSIONAL VISION

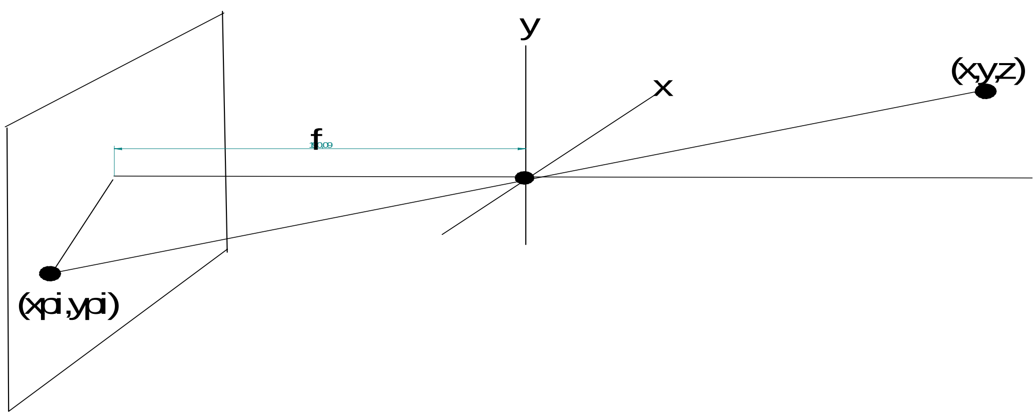


Figure 6 Perspective projection onto an image plane

The simple pinhole camera model for a perspective projective transformation is shown in Figure 6 and is easily modeled in homogeneous coordinates by the matrix transformation from ground to homogeneous image coordinates.

$$\begin{bmatrix} Wx_{PI} \\ Wy_{PI} \\ Wz_{PI} \\ W \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} * \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1/f & 1 \end{bmatrix} * \begin{bmatrix} x_g \\ y_g \\ z_g \\ 1 \end{bmatrix} \tag{25}$$

The physical image coordinates are determined by dividing by the scale term W.

$$\left(x_{PI}, y_{PI}, z_{PI}\right) = (Wx_{PI}/W, Wy_{PI}/W, Wz_{PI}/W) \tag{26}$$

In general the lens center may be translated and rotated with respect to the global coordinate system. This results in a general perspective projective matrix representation as shown. With projection, the image z term and corresponding matrix row can be considered discarded.

$$\begin{bmatrix} Wx_{PI} \\ Wy_{PI} \\ Wz_{PI} \\ W \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} \\ A_{21} & A_{22} & A_{23} & A_{24} \\ A_{31} & A_{32} & A_{33} & A_{34} \\ A_{41} & A_{42} & A_{43} & A_{44} \end{bmatrix} * \begin{bmatrix} x_g \\ y_g \\ z_g \\ 1 \end{bmatrix} \tag{27}$$

The stereo vision principles approach uses the scaling between the two coordinate systems to determine the relationship between the physical and image coordinates. If the image is projected on the z axis, the model equations relating the two coordinates are described by:

$$\begin{aligned} W\,x_{PI} &= A_{11}x_g + A_{12}y_g + A_{13}z_g + A_{14} \\ W\,y_{PI} &= A_{21}x_g + A_{22}y_g + A_{23}z_g + A_{24} \\ W &= A_{31}x_g + A_{32}y_g + A_{33}z_g + A_{34} \end{aligned} \tag{28}$$

The coefficients for the above equations may be computed by eliminating the scaling term, W, from the first two equations to obtain Equations 4 and 5. Then one can utilizing matching calibration points to solve for the unknown A coefficients.

$$\begin{aligned} &A_{11}x_g + A_{12}y_g + A_{13}z_g + A_{14} - A_{31}x_gx_{PI} - A_{32}y_gx_{PI} - A_{33}z_gx_{PI} = 0 \\ &A_{21}x_g + A_{22}y_g + A_{23}z_g + A_{24} - A_{31}x_gy_{PI} - A_{32}y_gy_{PI} - A_{33}z_gy_{PI} = 0 \end{aligned} \tag{29}$$

Equations 4 and 5 represent two equations in twelve unknown coefficients, A_{nm}. The coefficients are computed using magic matrix techniques by utilizing six matching calibration points. Since we are dealing with a homogeneous coordinate system, the matrix will include an arbitrary scale factor. If the coefficient A_{34} is set as unity, the resulting transformation matrix will be normalized.

With six calibration data points and $A_{34} = 1$, the following matrix equation was formulated.

$$\mathbf{QA = 0} \tag{30}$$

Where

$$Q = \begin{bmatrix} xg1 & yg1 & zg1 & 1 & 0 & 0 & 0 & 0 & -xg1.xPI1 & -yg1.xPI1 & -zg1.xPI1 & -xPI1 \\ 0 & 0 & 0 & 0 & xg1 & yg1 & zg1 & 1 & -xg1.yPI1 & -yg1.yPI1 & -zg1.yPI1 & -yPI1 \\ xg2 & yg2 & zg2 & 1 & 0 & 0 & 0 & 0 & -xg2.xPI2 & -yg2.xPI2 & -zg2.xPI2 & -xPI2 \\ 0 & 0 & 0 & 0 & xg2 & yg2 & zg2 & 1 & -xg2.yPI2 & -yg2.yPI2 & -zg2.yPI2 & -yPI2 \\ xg3 & yg3 & zg3 & 1 & 0 & 0 & 0 & 0 & -xg3.xPI3 & -yg3.xPI3 & -zg3.xPI3 & -xPI3 \\ 0 & 0 & 0 & 0 & xg3 & yg3 & zg3 & 1 & -xg3.yPI3 & -yg3.yPI3 & -zg3.yPI3 & -yPI3 \\ xg4 & yg4 & zg4 & 1 & 0 & 0 & 0 & 0 & -xg4.xPI4 & -yg4.xPI4 & -zg4.xPI4 & -xPI4 \\ 0 & 0 & 0 & 0 & xg4 & yg4 & zg4 & 1 & -xg4.yPI4 & -yg4.yPI4 & -zg4.yPI4 & -yPI4 \\ xg5 & yg5 & zg5 & 1 & 0 & 0 & 0 & 0 & -xg5.xPI5 & -yg5.xPI5 & -zg5.xPI5 & -xPI5 \\ 0 & 0 & 0 & 0 & xg5 & yg5 & zg5 & 1 & -xg5.yPI5 & -yg5.yPI5 & -zg5.yPI5 & -yPI5 \\ xg6 & yg6 & zg6 & 1 & 0 & 0 & 0 & 0 & -xg6.xPI6 & -yg6.xPI6 & -zg6.xPI6 & -xPI6 \\ 0 & 0 & 0 & 0 & xg6 & yg6 & zg6 & 1 & -xg6.yPI6 & -yg6.yPI6 & -zg6.yPI6 & -yPI6 \end{bmatrix} \tag{31}$$

And

$$A = [A_{11}, A_{12}, A_{13}, A_{14}, A_{21}, A_{22}, A_{23}, A_{24}, A_{31}, A_{32}, A_{33}, A_{34}]^{Tr} \tag{32}$$

There are 12 unknowns in the matrix equation. However, since the matrix equation is homogeneous, there is an arbitrary value or scale. The transformation coefficients can be solved by moving the last column in the matrix Q to the right-hand side and applying the least square regression method. Therefore, one coefficient can be arbitrarily selected leaving 11 coefficients to be determined. Since each image point (x, y) gives two equations, a minimum of five and one half image points could give a solution. A greater number of points permit a least squares solution. After the A coefficients are determined, W is computed and for any image coordinate, xPI and yPI, the corresponding ground coordinates may be computed as shown in the following.

$Q_1 A = [B]$
where

$$\begin{bmatrix} xg1 & yg1 & zg1 & 1 & 0 & 0 & 0 & 0 & -xg1.xPI1 & -yg1.xPI1 & -zg1.xPI1 \\ 0 & 0 & 0 & 0 & xg1 & yg1 & zg1 & 1 & -xg1.yPI1 & -yg1.yPI1 & -zg1.yPI1 \\ xg2 & yg2 & zg2 & 1 & 0 & 0 & 0 & 0 & -xg2.xPI2 & -yg2.xPI2 & -zg2.xPI2 \\ 0 & 0 & 0 & 0 & xg2 & yg2 & zg2 & 1 & -xg2.yPI2 & -yg2.yPI2 & -zg2.yPI2 \\ xg3 & yg3 & zg3 & 1 & 0 & 0 & 0 & 0 & -xg3.xPI3 & -yg3.xPI3 & -zg3.xPI3 \\ 0 & 0 & 0 & 0 & xg3 & yg3 & zg3 & 1 & -xg3.yPI3 & -yg3.yPI3 & -zg3.yPI3 \\ xg4 & yg4 & zg4 & 1 & 0 & 0 & 0 & 0 & -xg4.xPI4 & -yg4.xPI4 & -zg4.xPI4 \\ 0 & 0 & 0 & 0 & xg4 & yg4 & zg4 & 1 & -xg4.yPI4 & -yg4.yPI4 & -zg4.yPI4 \\ xg5 & yg5 & zg5 & 1 & 0 & 0 & 0 & 0 & -xg5.xPI5 & -yg5.xPI5 & -zg5.xPI5 \\ 0 & 0 & 0 & 0 & xg5 & yg5 & zg5 & 1 & -xg5.yPI5 & -yg5.yPI5 & -zg5.yPI5 \\ xg6 & yg6 & zg6 & 1 & 0 & 0 & 0 & 0 & -xg6.xPI6 & -yg6.xPI6 & -zg6.xPI6 \\ 0 & 0 & 0 & 0 & xg6 & yg6 & zg6 & 1 & -xg6.yPI6 & -yg6.yPI6 & -zg6.yPI6 \end{bmatrix} * \begin{bmatrix} A_{11} \\ A_{12} \\ A_{13} \\ A_{14} \\ A_{21} \\ A_{22} \\ A_{23} \\ A_{24} \\ A_{31} \\ A_{32} \\ A_{33} \\ \end{bmatrix} = \begin{bmatrix} xPI1 \\ yPI1 \\ xPI2 \\ yPI2 \\ xPI3 \\ yPI3 \\ xPI4 \\ yPI4 \\ xPI5 \\ yPI5 \\ xPI6 \\ yPI6 \end{bmatrix} \quad (33)$$

To solve for the coefficients, one may use the Moore-Penrose pseudo inverse,
First multiply by the transpose of the reduced matrix Q_1

$$Q_1^T * Q_1 * A = Q_1^T B \quad (34)$$

then multiply both sides by the inverse og the squart matrix $Q_1^T * Q_1$

$$A = (Q_1^T * Q_1)^{-1} Q_1^T B \quad (35)$$

Now given the A matrix coefficients, and the physical image coordinates, one may determine the three dimensional ground coordinates. If this pseudo inverse matrix computation comes out ill conditioned or with a small condition number, another way of doing this computation is needed. For example, the original equations may be rewritten in matrix form as:

$$\begin{bmatrix} W * x_{PI} \\ W * y_{PI} \\ W \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} & A_{13} & A_{14} \\ A_{21} & A_{22} & A_{23} & A_{24} \\ A_{31} & A_{32} & A_{33} & A_{34} \end{bmatrix} * \begin{bmatrix} x_g \\ y_g \\ z_g \\ 1 \end{bmatrix} \quad (36)$$

$$\begin{bmatrix} W*x_{PI} \\ W*y_{PI} \\ W \end{bmatrix} = \begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} * \begin{bmatrix} x_g \\ y_g \\ z_g \end{bmatrix} + \begin{bmatrix} A_{14} \\ A_{24} \\ A_{34} \end{bmatrix}$$

or

$$\begin{bmatrix} A_{11} & A_{12} & A_{13} \\ A_{21} & A_{22} & A_{23} \\ A_{31} & A_{32} & A_{33} \end{bmatrix} * \begin{bmatrix} x_g \\ y_g \\ z_g \end{bmatrix} = \begin{bmatrix} W*x_{PI} - A_{14} \\ W*y_{PI} - A_{24} \\ W - A_{34} \end{bmatrix}$$

or

$$\begin{bmatrix} x_g \\ y_g \\ z_g \end{bmatrix} = inv(A) * \begin{bmatrix} W*x_{PI} - A_{14} \\ W*y_{PI} - A_{24} \\ W - A_{34} \end{bmatrix} \qquad (37)$$

where inv(A) is the inverse of the 3 by 3 A matrix, W is the scaling factor, A_{nm} are coefficients, x_{PI} and y_{PI} are x and y image coordinates, and x_g, y_g, and z_g are the ground coordinates.

So with this formulation, the ground coordinates can be computed with an estimate of the single scale parameter W. Further experimentation is needed to determine if this method is robust.

The following example was computed as a test case.

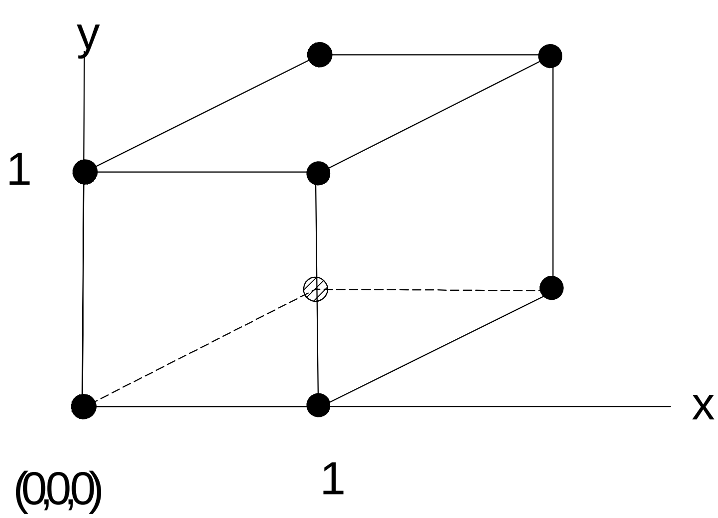


Figure 7. Example computation for the 8 vertices of a 3D cube

A popular example [10] uses the 8 vertices of a 3D cube as shown in Figure 7 and computes a perspective projective transformation using a lens centered ay (0.5, 0.5, 4.0). Three transformations are effected: a translation, a perspective and a projection on the z=0 plane. These are represented by the matrices given below.

$$T_3 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \textbf{ Projective} \quad (38)$$

$$T_2 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0.5 & 1 \end{pmatrix} \textbf{ Perspective} \quad (39)$$

$$T_1 = \begin{pmatrix} 1 & 0 & 0 & -0.5 \\ 0 & 1 & 0 & -0.5 \\ 0 & 0 & 1 & -4 \\ 0 & 0 & 0 & 1 \end{pmatrix} \textbf{ Translation} \quad (40)$$

The composite transformation matrix is the product of the three component matrices.

$$T = T_3 \cdot T_2 \cdot T_1 = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0.5 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & -0.5 \\ 0 & 1 & 0 & -0.5 \\ 0 & 0 & 1 & -4 \\ 0 & 0 & 0 & 1 \end{pmatrix} \quad (41)$$

Now the original data points for the vertices of the cube may be written in the homogeneous coordinate matrix as:

$$D = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \quad (42)$$

And now may be multiplied by the composite transformation matrix, T

$$T = \begin{pmatrix} 1 & 0 & 0 & -0.5 \\ 0 & 1 & 0 & -0.5 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & -1 \end{pmatrix} \quad (43)$$

The new image points in homogeneous coordinates may then be computed as:

$$D_{new} = T * D \quad (44)$$

$$D_{new} = \begin{pmatrix} 1 & 0 & 0 & -0.5 \\ 0 & 1 & 0 & -0.5 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & -1 \end{pmatrix} \cdot \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} \tag{45}$$

$$\begin{pmatrix} 1 & 0 & 0 & -0.5 \\ 0 & 1 & 0 & -0.5 \\ 0 & 0 & 0 & 0 \\ 0 & 0 & 0.5 & -1 \end{pmatrix} \cdot \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \\ 1 & 1 & 1 & 1 & 1 & 1 & 1 & 1 \end{pmatrix} = \begin{pmatrix} -0.5 & -0.5 & -0.5 & -0.5 & 0.5 & 0.5 & 0.5 & 0.5 \\ -0.5 & -0.5 & 0.5 & 0.5 & -0.5 & -0.5 & 0.5 & 0.5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -0.5 & -1 & -0.5 & -1 & -0.5 & -1 & -0.5 \end{pmatrix}$$

To get physical image coordinates x,y,z one must do the nonlinear conversion from homogeneous coordinates as shown.

$$\begin{pmatrix} Wx_{PI} \\ Wy_{PI} \\ 0 \\ W \end{pmatrix} = \begin{pmatrix} -0.5 & -0.5 & -0.5 & -0.5 & 0.5 & 0.5 & 0.5 & 0.5 \\ -0.5 & -0.5 & 0.5 & 0.5 & -0.5 & -0.5 & 0.5 & 0.5 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -0.5 & -1 & -0.5 & -1 & -0.5 & -1 & -0.5 \end{pmatrix} \tag{46}$$

The A matrix and its inverse is

$$A = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & .5 \end{pmatrix} \tag{47}$$

$$AINV = \begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix} \tag{48}$$

Now we can compute all of the 8 vertex points

This is the WxPI WyPI W term for 8 points - note the different values of W:

$$WPI = \begin{pmatrix} -0.5 & -0.5 & -0.5 & -0.5 & 0.5 & 0.5 & 0.5 & 0.5 \\ -0.5 & -0.5 & 0.5 & 0.5 & -0.5 & -0.5 & 0.5 & 0.5 \\ -1 & -0.5 & -1 & -0.5 & -1 & -0.5 & -1 & -0.5 \end{pmatrix} \tag{49}$$

This is the A14 A24 A34 term for 8 points - note they are all the same

$$A1234 = \begin{pmatrix} -0.5 & -0.5 & -0.5 & -0.5 & -0.5 & -0.5 & -0.5 & -0.5 \\ -0.5 & -0.5 & -0.5 & -0.5 & -0.5 & -0.5 & -0.5 & -0.5 \\ -1 & -1 & -1 & -1 & -1 & -1 & -1 & -1 \end{pmatrix} \tag{50}$$

Compute the difference for all 8 vertices

$$\text{WPI} - \text{A1234} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0.5 & 0 & 0.5 & 0 & 0.5 & 0 & 0.5 \end{pmatrix} \quad (51)$$

Now multiply by the inverse of A and see that it gives the exact same values for the x, y and z values for all 8 vertices of the cube

$$\begin{pmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 2 \end{pmatrix} \cdot \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 0.5 & 0 & 0.5 & 0 & 0.5 & 0 & 0.5 \end{pmatrix} = \begin{pmatrix} 0 & 0 & 0 & 0 & 1 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 & 0 & 0 & 1 & 1 \\ 0 & 1 & 0 & 1 & 0 & 1 & 0 & 1 \end{pmatrix} \quad (52)$$

These may be compared to the original vertex data D. This one example illustrates the concept and shows the mathematical steps. The result is also shown in Figure 8.

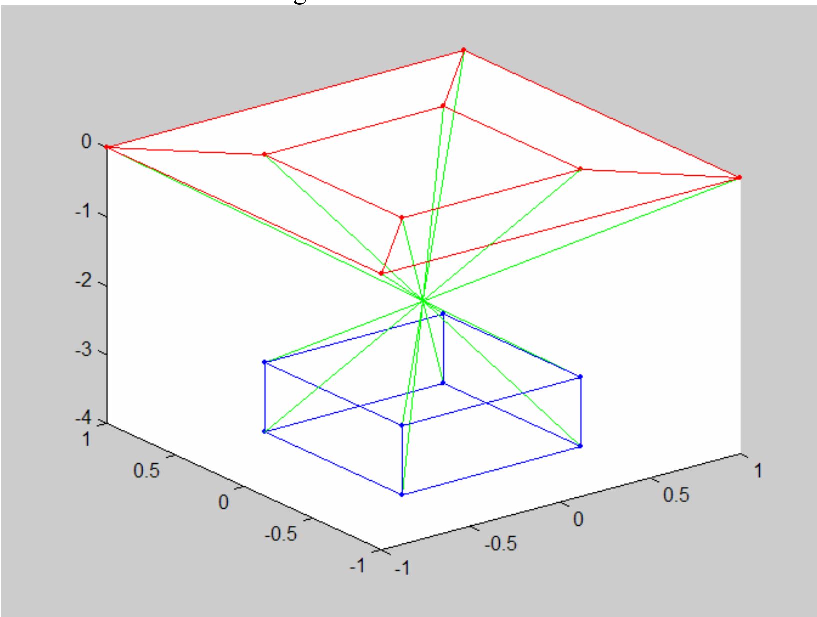


Figure 8 Perspective projective mapping of a unit cube

This example clearly shows that one can exactly compute the three dimensional coordinates of points such as the vertices of the unit cube after a perspective, projective transformation if one has the transformation matrices and the homogeneous coordinates for the data. This includes the scale term W. Often we only have the physical coordinates such as XPI=W*XPI/W and YPI =W*YPI/W. However, there are many methods to determine the scale term W. The normal stereo uses images separated in space. However, images separated in time, spectral color, or other methods can also be used. Single eye depth measurements in humans are not only possible but also permit those with monocular vision to drive cars in most states.

Seven Degree of Freedom Camera Model

Many studies such as those described in [11-14] have been made for three dimension vision. We have found that a full seven degree of freedom model is helpful in mobile robot navigation. The matrix transformation for this model is shown below.

$$\begin{pmatrix} x_I \\ y_I \\ z_I \\ w_I \end{pmatrix} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & \frac{-1}{f} & 1 \end{pmatrix} \cdot \begin{pmatrix} \cos\phi & \sin\phi & 0 & 0 \\ -\sin\phi & \cos\phi & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} \cos\theta & 0 & \sin\theta & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\theta & 0 & \cos\theta & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha & 0 \\ 0 & \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} 1 & 0 & 0 & -H \\ 0 & 1 & 0 & -K \\ 0 & 0 & 1 & -L \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} x_g \\ y_g \\ z_g \\ 1 \end{pmatrix} \tag{53}$$

$$\begin{pmatrix} x_I \\ y_I \\ z_I \\ w_I \end{pmatrix} = \begin{bmatrix} \cos\phi\cdot\cos\theta\cdot x_g + y_g\cdot\sin\phi\cdot\cos\alpha + y_g\cdot\cos\phi\cdot\sin\theta\cdot\sin\alpha - z_g\cdot\sin\phi\cdot\sin\alpha + z_g\cdot\cos\phi\cdot\sin\theta\cdot\cos\alpha - \cos\phi\cdot\cos\theta\cdot H - K\cdot\sin\phi\cdot\cos\alpha - K\cdot\cos\phi\cdot\sin\theta\cdot\sin\alpha + L\cdot\sin\phi\cdot\sin\alpha - L\cdot\cos\phi\cdot\sin\theta\cdot\cos\alpha \\ -\sin\phi\cdot\cos\theta\cdot x_g + y_g\cdot\cos\phi\cdot\cos\alpha - y_g\cdot\sin\phi\cdot\sin\theta\cdot\sin\alpha - z_g\cdot\cos\phi\cdot\sin\alpha - z_g\cdot\sin\phi\cdot\sin\theta\cdot\cos\alpha + \sin\phi\cdot\cos\theta\cdot H - K\cdot\cos\phi\cdot\cos\alpha + K\cdot\sin\phi\cdot\sin\theta\cdot\sin\alpha + L\cdot\cos\phi\cdot\sin\alpha + L\cdot\sin\phi\cdot\sin\theta\cdot\cos\alpha \\ -\sin\theta\cdot x_g + \cos\theta\cdot\sin\alpha\cdot y_g + \cos\theta\cdot\cos\alpha\cdot z_g + \sin\theta\cdot H - \cos\theta\cdot\sin\alpha\cdot K - \cos\theta\cdot\cos\alpha\cdot L \\ \frac{-\left(-\sin\theta\cdot x_g + \cos\theta\cdot\sin\alpha\cdot y_g + \cos\theta\cdot\cos\alpha\cdot z_g + \sin\theta\cdot H - \cos\theta\cdot\sin\alpha\cdot K - \cos\theta\cdot\cos\alpha\cdot L - f\right)}{f} \end{bmatrix} \tag{54}$$

The seven degree of freedom model that takes into account the translation offset of the lens center (H,K,L), the rotation about the x, y and z axes and the focal length of the camera, f, can be used to formulate many different imaging solutions. If the image plane is projected on the y axis, then the y component can be set to zero. If the image plane is projected on the z axis then the z component can be set to zero. The physical image coordinates are equal to the homogeneous coordinate divided by the scale term W.

4. CONCLUSIONS

In this paper we have provided some simple computational examples regarding two and three dimensional vision in an attempt to clear up some "conventional wisdom misconceptions" such as depth measurement requires two eyes. And to demonstrate that the earth can be considered (flat) two or (round) three dimensional depending on the local or global viewing situation. Sometimes two and three dimensional information can be mapped directly.

References

1. E. L. Hall and B. C. Hall, **Robotics - A User Friendly Introduction**, Holt, Rinehart and Winston, (1985).
2. *http://www.spacedaily.com/news/cosmology-05za.html and http://arxiv.org/PS_cache/hep-th/pdf/0506/0506053.pdf*
3. http://www.siggraph.org/publications/newsletter/v33n4/contributions/roble.html
4. E. L. Hall, **Computer Image Processing and Recognition**, Academic Press, New York (1979).
5. O. Faugeras, **Three-Dimensional Computer Vision**, MIT Press, Cambridge (1993).
6. "Matlab 7.0.1," The Mathworks.
7. E. W. Weisstein, et al., Manifold, From *MathWorld* <*http://mathworld.wolfram.com/*>--A Wolfram Web Resource.
8. G. F. Simmons, **Topology and Modern Analysis**, Mc-Graw Hill, New York (1963).
9. J. P. Snyder, **Map Projections - A Working Manual**, U.S. Government Printing Office, Washington, D.C (1987).
10. E. L. Hall, "Fundamental Principles of Robot Vision," in **Handbook of Pattern Recognition and Image Processing,** pp. 543-575, Academic Press, New York (1994).
11. M. Z. Brown, D. Burschka and G. D. Hager, "Advances in computational stereo," **IEEE Trans. on Pattern Analysis and Machine Intelligence**, 25(8), (2003).
12. F. Hong and Y. Baozong, "An Accurate and Practical Camera Calibration System for 3D Computer Vision,"**Chinese Journal of Electronics**, 1(1), 63-71 (1991).
13. Y. Liu, T. S. Huang and O. D. Faugeras, "Determination of Camera Location from 2-D to 3-D Line and Point Correspondence.," **IEEE Transaction on Pattern Analysis and Machine Intelligence**, 12(1), 28-37 (1990).
14. D. Rosselot, "Processing real-time stereo video for obstacle detection using disparity space," MS Thesis, Department of Electrical & Computer Engineering and Computer Science, University of Cincinnati (2005).