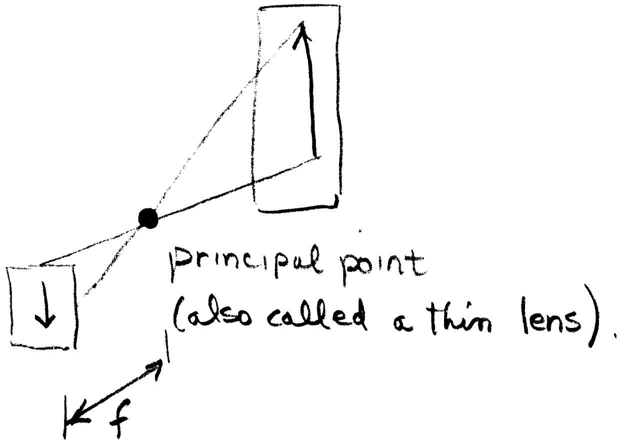
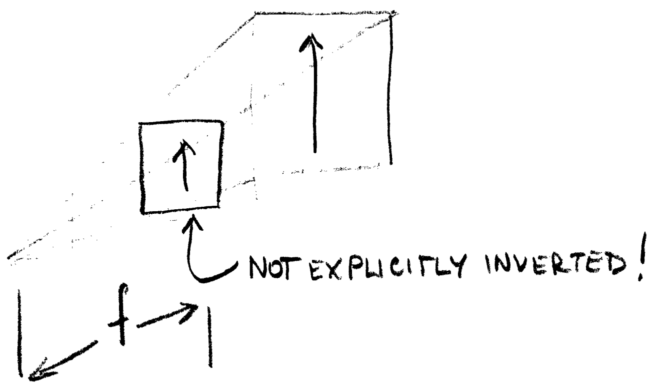
Perspective (imaging)



computer scientists flip around.



f = focal length of lens.

In general $\frac{1}{o} = \frac{1}{i} - \frac{1}{f}$ (Len's Law)

o = object distance

i = image distance

f = focal length (a constant)

As $o \to \infty$ (or just gets large) $\frac{1}{i} \to \frac{1}{f}$ or $i \to f$

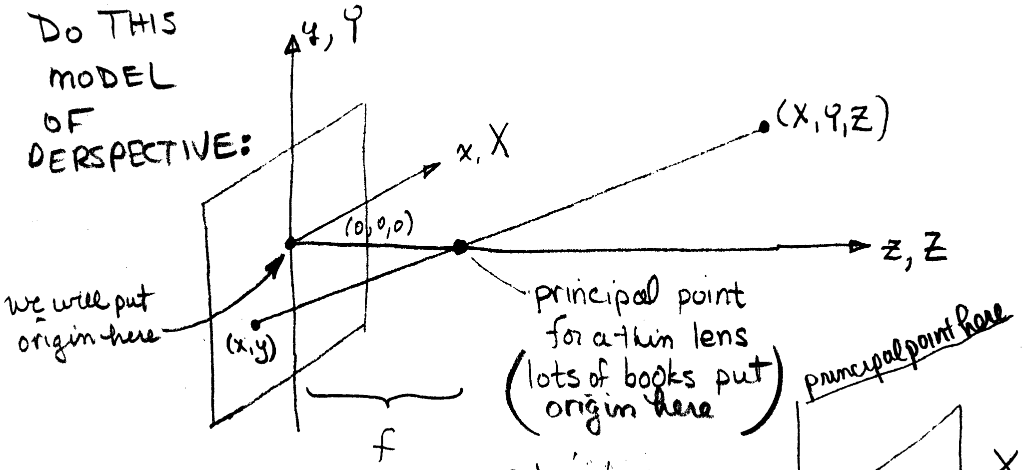
This is approximation we usually use.

Consider as f changes

long focal length (telephoto) $f \to \infty$ perspective $\to$ orthographic transform (a projection)

short focal length (wide angle) $f \to 0$ lots of distortion

DO THIS MODEL OF PERSPECTIVE:

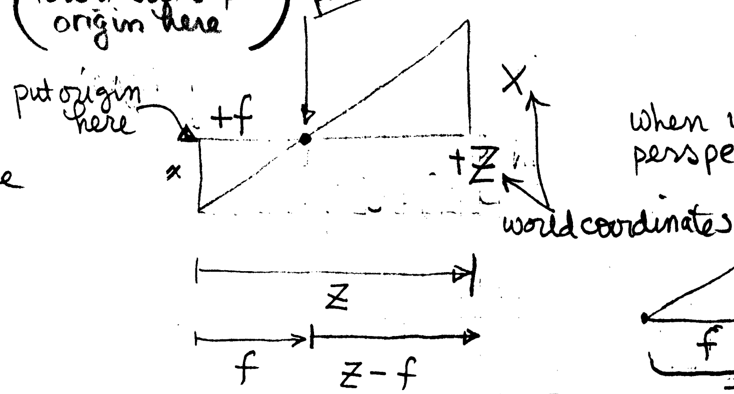


technically there is a z.

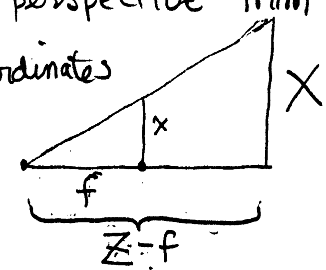
(x, y, z) camera coordinates

(X, Y, Z) world coordinates

Examining figure



when we go to perspective from



$$\therefore \frac{X}{Z-f} = \frac{-x}{f}$$ (x is negative because its inverted)

$$\therefore \frac{x}{f} = -\frac{X}{Z-f} = \frac{X}{f-Z}$$

similarily

$$\frac{y}{f} = -\frac{Y}{Z-f} = \frac{Y}{f-Z}$$

The focal plane coordinates are then

camera plane coordinates

$$\begin{cases} x = \dfrac{f}{f-Z} X \\ y = \dfrac{f}{f-Z} Y \end{cases}$$

world coordinates

This is exactly the perspective transformation, i.e.

$$x' = \frac{x}{1-\frac{z}{f}} = \frac{f}{f-z} x$$

Note: mapping a 3-D scene onto the image plane is a many-to-one transformation.

we actually have the parametric equation of the line given the image plane coordinates (x_0, y_0)

$$x_0 = \frac{f}{f-Z} X$$

$$X = \frac{f x_0 - x_0 Z}{f} = x_0 - \frac{x_0}{f} Z$$

$$Y = y_0 - \frac{y_0}{f} Z$$

Note that given any (x_0, y_0) there are infinitely many points on the line given by $(X(Z), Y(Z), Z)$. This means that you <u>CANNOT</u> recover 3-D point information by means of the inverse perspective transform <u>unless</u> you know at least one of the world coordinates of the point.

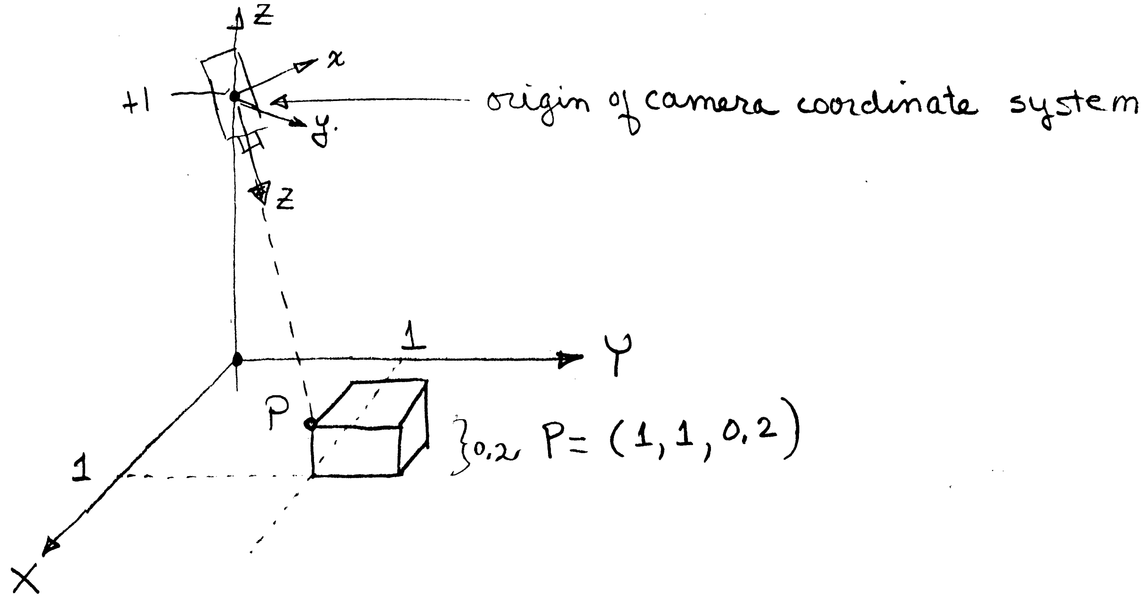
<u>Homogeneous representation</u>

$$\begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{f} & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = \begin{bmatrix} x \\ y \\ z \\ 1 - \frac{z}{f} \end{bmatrix}$$

All elements are now scaled by $\frac{f-z}{f}$

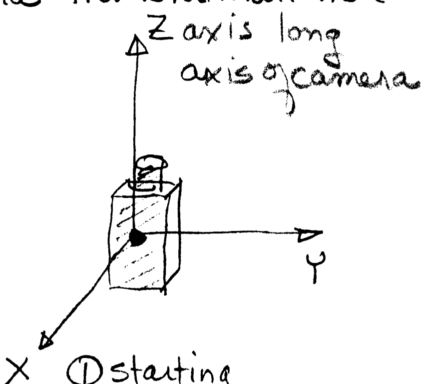
In general we don't see a z-perspective transform with a camera although we regard it as occurring.

Example:

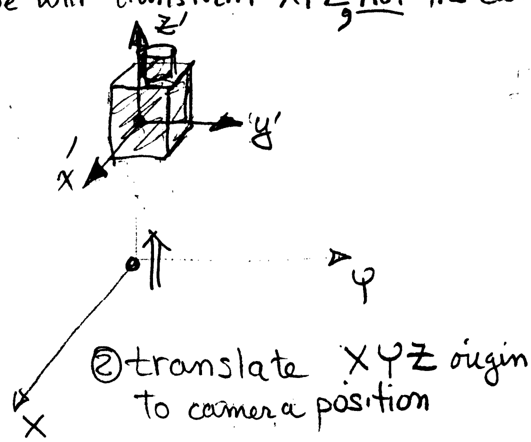


The camera is viewing point P. The camera is offset from the origin of the (X, Y, Z) coordinate system along Z by +1 and has a pan = 45°, tilt = 135°.

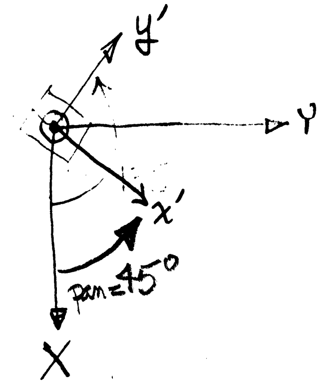
Camera transformations (we will transform XYZ, not the camera, mathematically)



① starting position

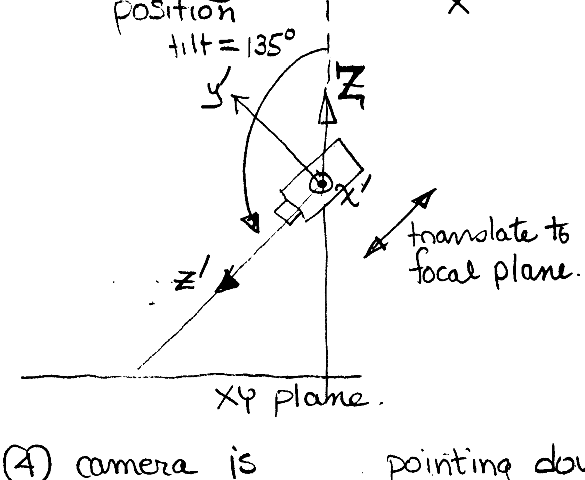


② translate XYZ origin to camera position



TOP VIEW

③ rotate XYZ abou z-axis 45° (pan angle)



we will transform world coordinates to camera coordinates

④ camera is pointing down. Tilt Z-axis by 135° to look at block. (rotate about x')

⑤ now move to image plane from center of camera.

For this example:

do translation

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 0 \end{bmatrix}$$ this translates it -1

then pan

$$R_{\theta,z} = \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}_{\theta=45^\circ} = \begin{bmatrix} 0.71 & +0.71 & 0 & 0 \\ -0.71 & +0.71 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$\sin 135^\circ > 0$
$\cos 135 < 0$

$$R_{\alpha x} = \begin{bmatrix} -1 & 0 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha & 0 \\ 0 & -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}_{\alpha=135^\circ} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -.71 & +.71 & 0 \\ 0 & -.71 & -.71 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R = R_\alpha R_\theta = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & \cos\alpha & \sin\alpha & 0 \\ 0 & -\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} \cos\theta & \sin\theta & 0 & 0 \\ -\sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \cos\alpha & \sin\theta & 0 & 0 \\ -\sin\theta\cos\alpha & \cos\theta\cos\alpha & \sin\alpha & 0 \\ \sin\theta\sin\alpha & -\cos\theta\sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}.$$

Translate to camera plane.

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -.02 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

from camera manufacturers data
$f = 35mm$.

$$\underline{r} \begin{cases} x = 0 \\ y = 0 \\ z = .02m \end{cases}$$

Perspective transform

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{f} & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{.035} & 1 \end{bmatrix}$$

all units in meters

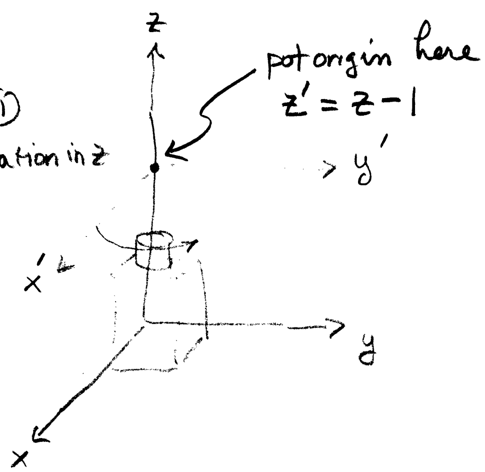
How would we represent mathematically world coordinates to camera frame?

Camera = [perspective ⑤] [move back in z ④] [+135° rot about x ③] [+45° rot about +z ②] [move up in z ①] [world coordinates]

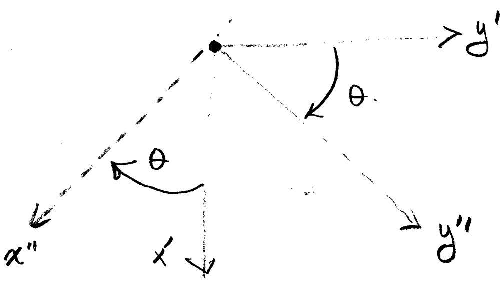
Camera Coordinates $= \underline{\underline{C}}\, w_R = P\ T\ R\ G\ w_R$

- G — translate $O(x,y,z)$ to gimbal center of camera
- R — pan-tilt rotation
  - pan - horizontal rotatión
  - tilt - from original z-axis
- T — displacement of image plane from center of camera. Usually use principal point for center of camera
- P — perspective transformation

① translation in z

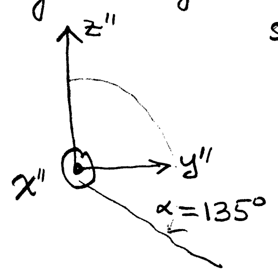


② top view



③ Now rotate about x-axis to make coordinate systems align

from side view

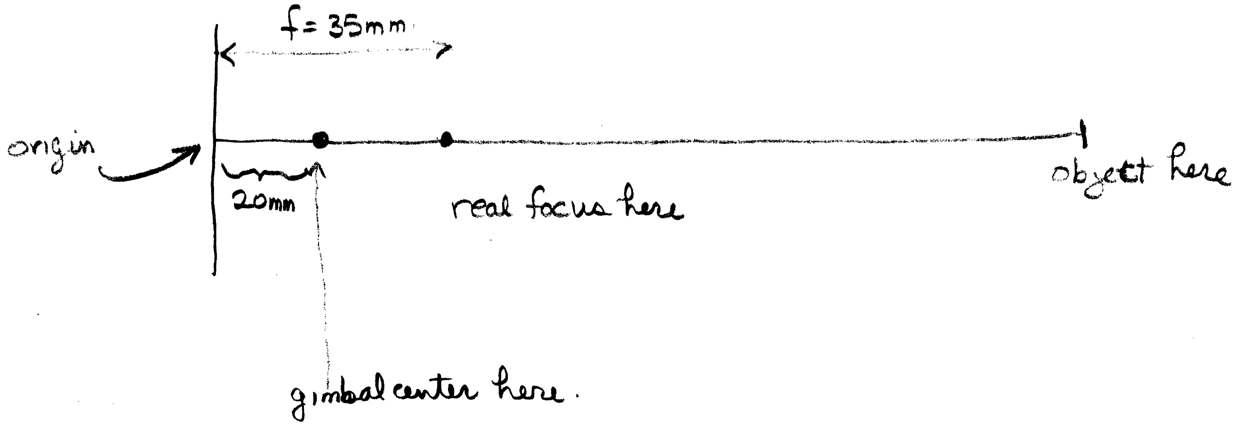


so camera is now looking at block.

④ move to focal plane is simple translation in Z

$z''' = z''' - f$

⑤ perspective transform!



Now, I need to redo calculations to get overall camera transformation

$$\underline{\underline{C}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{.035} & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -.02 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & -.71 & +.71 & 0 \\ 0 & -.71 & -.71 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} .71 & .71 & 0 & 0 \\ -.71 & .71 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -\frac{1}{.035} & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -.02 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} .71 & .71 & 0 & 0 \\ .5 & -.5 & .71 & 0 \\ .5 & -.5 & -.71 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -1 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -.02 \\ 0 & 0 & (1-\frac{1}{.035}) & 1+\frac{.02}{.035} \end{bmatrix} \begin{bmatrix} .71 & .71 & 0 & 0 \\ .5 & -.5 & .71 & -.71 \\ .5 & -.5 & -.71 & +.71 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} .71 & .71 & 0 & 0 \\ .5 & -.5 & .71 & -.71 \\ .5 & -.5 & -.71 & +.71-.02 \\ .5(1-\frac{1}{.035}) & -.5(1-\frac{1}{.035}) & -.71(1-\frac{1}{.035}) & .71(1-\frac{1}{.035})+(1+\frac{.02}{.035}) \end{bmatrix}$$

To image point on block $\begin{bmatrix} 1 \\ 1 \\ .2 \\ 1 \end{bmatrix}$

compute $\underline{\underline{C}} \begin{bmatrix} 1 \\ 1 \\ .2 \\ 1 \end{bmatrix}$

overa... camera transformation

$$\underline{\underline{C}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 1 & 1 \end{bmatrix} \begin{bmatrix} -0.5 & -0.71 & +0.5 & 0 \\ -0.5 & 0.71 & +0.5 & 0 \\ -0.71 & 0 & -0.71 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & .02 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -\frac{1}{.035} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} -0.5 & -0.71 & +0.5 & 0 \\ -0.5 & +0.71 & +0.5 & 0 \\ -0.71 & 0 & -0.71 & 0 \\ +0.71 & 0 & +0.71 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -\frac{1}{.035} \\ 0 & 0 & .02 & 1-\frac{.020}{.035} \end{bmatrix}$$

$(x\ y\ z\ 1)$

$$= \begin{bmatrix} -0.5 & -0.71 & +0.5 & -\frac{0.5}{0.035} \\ -0.5 & 0.71 & +0.5 & -\frac{0.5}{.035} \\ -0.71 & 0 & -0.71 & +\frac{0.71}{.035} \\ +0.71 & 0 & 0.71+.020 & -\frac{0.71}{.035}+1-\frac{0.020}{0.035} \end{bmatrix}$$

not sure of this

Result of imaging point on block is $[1\ \ 1\ \ (1-\sqrt{2})\ \ 1]\ \underline{\underline{C}} = [0\ \ 0\ \ 2.02\ \ —]$

$\therefore$ this point lies on the optical axis of the camera.

evaluate transformation

$X = 1$
$Y = 1$
$Z = 0.2$

$$x = \frac{-0.5X - 0.5Y - 0.71Z + 0.71}{-0.5\frac{X}{.035} - 0.5\frac{Y}{.035} + 0.71\frac{Z}{.035} + 1 - \frac{0.71}{.035} - \frac{0.020}{0.035}}$$

$$x = (0.035)\frac{-0.5X - 0.5Y - 0.71Z + 0.71}{-0.5X - 0.5Y + 0.71Z + (.035 - 0.710 - 0.020)}$$

$$x = (0.035)\frac{\cancel{-0.5} - \cancel{0.5} - 0.142 + 0.71}{-0.5 - \cancel{0.5} + 0.142 - 0.695} = (0.035)\frac{+0.4343}{-0.985} = .0154 \text{ m.}$$

$$y = (0.035)\frac{0.71X - 0.71Y}{-0.985} = 0$$

$$z = (0.035)\frac{+0.5X + 0.5Y - 0.71Z + (0.71 + 0.020)}{-0.985}$$

$$= (0.035)\frac{\cancel{+0.5} + 0.5 - \cancel{0.71}(.2) \cancel{+0.71} + 0.02}{-0.985} = (0.035)\frac{1.59}{-.985}$$

$\cong -0.036$ (not too far off from image plane)

so image is a little bit out of focus

check on transform and where point is:

If we neglect perspective transform:

$$\underline{\underline{C'}} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 0 \end{bmatrix} \begin{bmatrix} -0.5 & -0.71 & +0.5 & 0 \\ +0.5 & 0.71 & +0.5 & 0 \\ -0.71 & 0 & -0.71 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & .02 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} -0.5 & -0.71 & +0.5 & 0 \\ -0.5 & 0.71 & +0.5 & 0 \\ -0.71 & 0 & -0.71 & 0 \\ +0.71 & 0 & +0.71 & 1 \end{bmatrix} \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & .02 & 1 \end{bmatrix}$$

$-X$

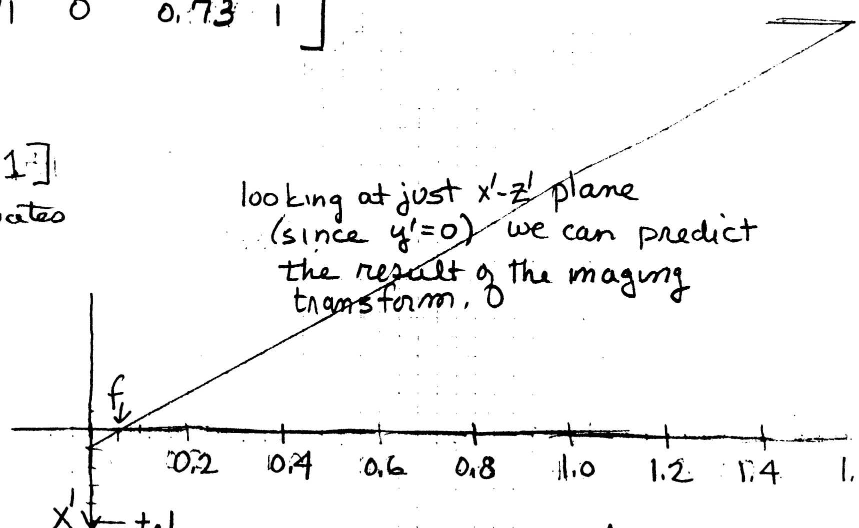
$$= \begin{bmatrix} -0.5 & -0.71 & +0.5 & 0 \\ -0.5 & 0.71 & +0.5 & 0 \\ -0.71 & 0 & -0.71 & 0 \\ +0.71 & 0 & 0.71+.02 & 1 \end{bmatrix} = \begin{bmatrix} -0.5 & -0.71 & +0.5 & 0 \\ -0.5 & 0.71 & +0.5 & 0 \\ -0.71 & 0 & -0.71 & 0 \\ +0.71 & 0 & 0.73 & 1 \end{bmatrix}$$

$$(x' \ y' \ z' \ t') = \begin{bmatrix} 1 & 1 & 0.2 & 1 \end{bmatrix} \begin{bmatrix} -0.5 & -0.71 & +0.5 & 0 \\ -0.5 & 0.71 & +0.5 & 0 \\ -0.71 & 0 & -0.71 & 0 \\ +0.71 & 0 & 0.73 & 1 \end{bmatrix}$$

$$\begin{array}{cccc} x' & y' & z' & \\ \end{array}$$
$$= \begin{bmatrix} -0.43 & 0 & 1.59 & 1 \end{bmatrix}$$

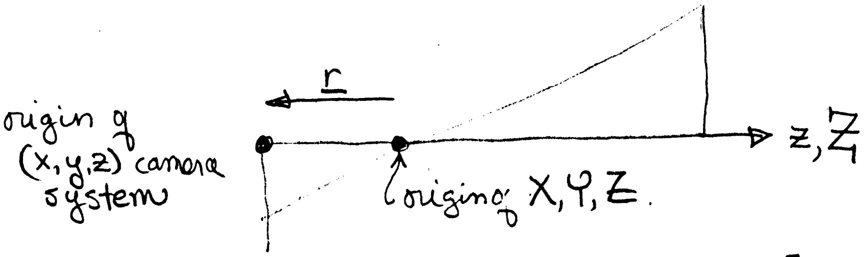
in camera frame coordinates

looking at just x'-z' plane (since $y'=0$) we can predict the result of the imaging transform.



says approximately +0.02 when imaged (high)

The problem is critically dependent upon knowing where the principal point i translate origin to camera plane (not shown in origin drawings

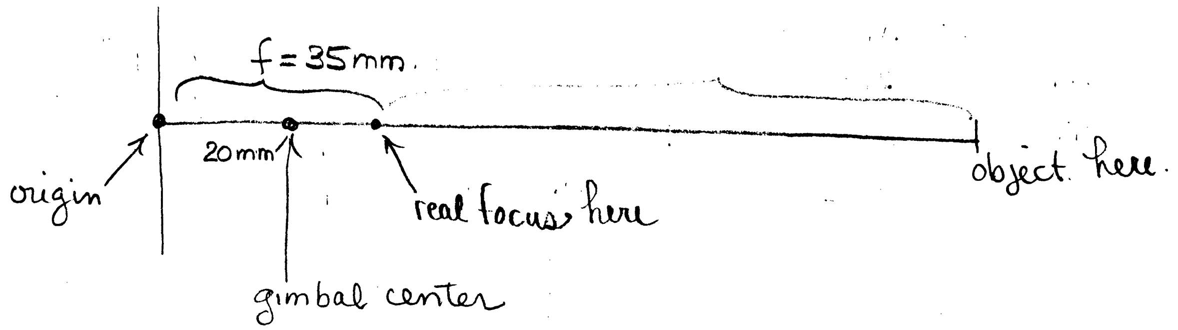


from camera manufacturer's data $\quad f = 35mm$

$$\underline{r}\begin{cases} x = 0 \\ y = 0 \\ z = 0.02\ m \end{cases}$$

$$T = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & +.02 & 1 \end{bmatrix}$$

$$P = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -\frac{1}{f} \\ 0 & 0 & 0 & 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & -\frac{1}{.035} \\ 0 & 0 & 0 & 1 \end{bmatrix}$$



in this example

$$G = \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 1 \end{bmatrix}$$

$\uparrow$ all old coordinates have $z' = z - 1$ in new coordinate system so translation is -1.

do pan first

$$R_{\theta,z} = R_{+45^\circ,z} = \begin{bmatrix} \cos\theta & -\sin\theta & 0 & 0 \\ \sin\theta & \cos\theta & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}_{\theta = +45^\circ}$$

$$R_{+45^\circ,z} = \begin{bmatrix} 0.71 & -0.71 & 0 & 0 \\ +0.71 & +0.71 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

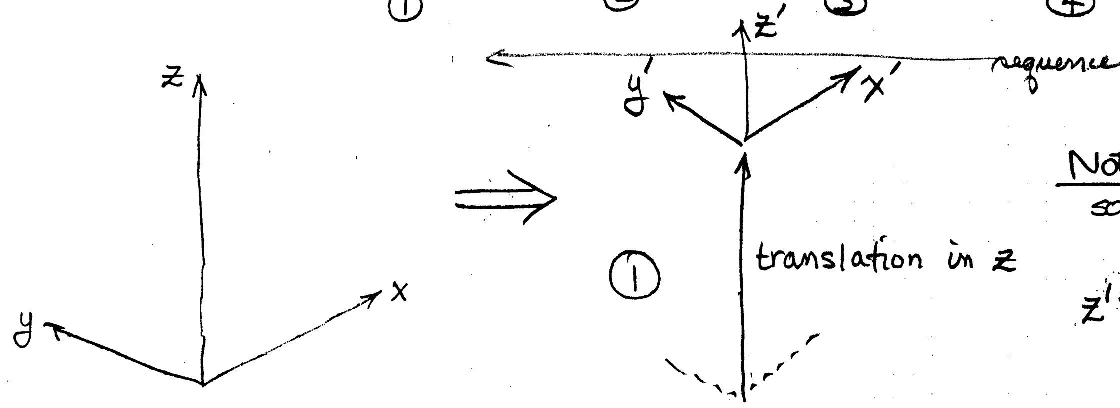
now do tilt

$$R_{+135^\circ,y} = \begin{bmatrix} \cos\alpha & 0 & \sin\alpha & 0 \\ 0 & 1 & 0 & 0 \\ -\sin\alpha & 0 & \cos\alpha & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}_{\alpha = +135^\circ} = \begin{bmatrix} -0.71 & 0 & +0.71 & 0 \\ 0 & 1 & 0 & 0 \\ -0.71 & 0 & -0.71 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

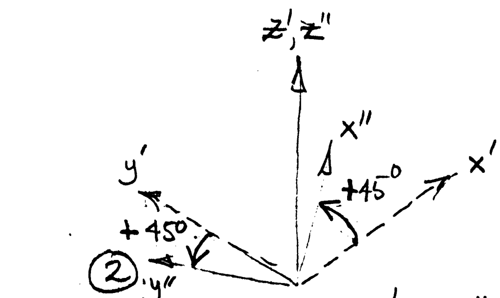
$$R = R_{+45^\circ,z}\, R_{+135^\circ,y} = \begin{bmatrix} +0.71 & -0.71 & 0 & 0 \\ +0.71 & +0.71 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} -0.71 & 0 & +0.71 & 0 \\ 0 & 1 & 0 & 0 \\ -0.71 & 0 & -0.71 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

$$R = \begin{bmatrix} -0.5 & -0.71 & +0.5 & 0 \\ -0.5 & +0.71 & +0.5 & 0 \\ -0.71 & 0 & -.71 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix}$$

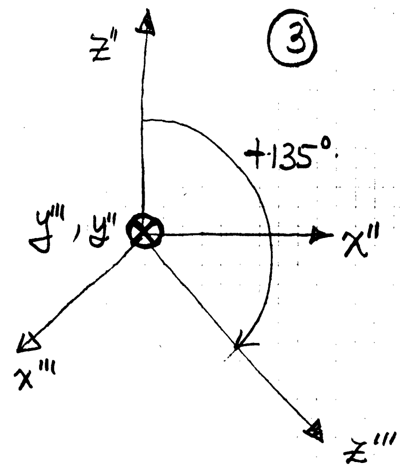
$$\left(\overset{z'}{\text{camera}}\right) = \left(\overset{z}{\text{world}}\right)\underset{①}{\begin{pmatrix}\text{move up}\\ \text{in } z\end{pmatrix}}\underset{②}{\begin{pmatrix}+45^\circ\text{ rot}\\ \text{about } z\end{pmatrix}}\underset{③}{\begin{pmatrix}+135^\circ\text{ rot}\\ \text{about } z\end{pmatrix}}\underset{④}{\begin{pmatrix}\text{move back}\\ \text{in } z\end{pmatrix}}$$



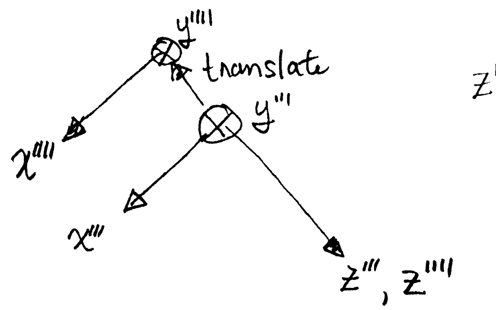
Note $z' = z - 1$
so translation is
-1 from z to z'
z' to z is $+1$.



Note: rotation from x' to x'' $+45^\circ$
from x'' to x' -45°



Note: rotation f.
z'' to z''' is +



$z''' = z''$

All the signs are reversed because we are post multiplying rather th pre multiplying.

overall transformation of <u>world coordinates</u> to camera coordinates

$$\underset{\substack{\text{camera}\\ \text{transform}}}{{}^{W_R}\underline{\underline{C}}} = \underset{\substack{\text{world}\\ \text{coordinates}}}{W_R}\ \overbrace{G\ \ R}^{\text{geometry}}\ \overbrace{T\ \ P}^{\text{camera}}$$

P — perspective transformati

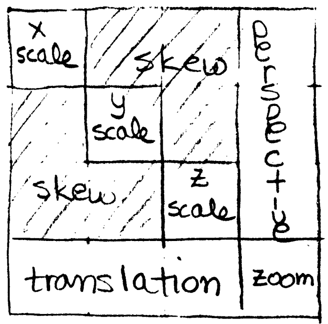
T — displacement of image plane from center of camera (usually use principal point for center of camera

R — pan-tilt rotation
pan - horizontal rotation
tilt - from $+z$ axis

G — translate $O(x,y,z)$ to <u>gimbal</u> center of camera

In general, we can have a 4x4 homogeneous camera transform

$$[x \quad y \quad z \quad 1]$$



$$= [x' \quad y' \quad z' \quad 1]$$

x' y': transverse camera coordinates

z': not of great importance. Should be near the focal plan if lens law is satisified.

$$[x \quad y \quad z \quad 1] \; \underline{\underline{C}} = [\underbrace{u, v,}_{\substack{\text{camera}\\ \text{frame}\\ \text{coordinates}\\ \text{(drop } z)}} t] \qquad (*)$$

this makes C 3x4 not 4x4.

t: not necessarily normalized to 1 (perspective transforms change the 1)

Calibration problem: given (*) above and enough (x,y,z) to (u,v) data determine $\underline{\underline{C}}$

Define normalized camera (transverse) frame coordinates

These are what we would measure:
$$U = \frac{u}{t}, \quad V = \frac{v}{t} \qquad (1)$$

Break (*) into three separate equations

$$\begin{aligned} [x \quad y \quad z \quad 1] \; \underline{C_1} &= u \\ [x \quad y \quad z \quad 1] \; \underline{C_2} &= v \\ [x \quad y \quad z \quad 1] \; \underline{C_3} &= t \end{aligned} \qquad (2)$$

Re-write equations (1) as

$$u - Ut = 0 \qquad v - Vt = 0$$

and substitute (2) into the above expressions.

In general, we can have a 4x4 homogeneous camera transform:

transverse camera coordinates: x', y'

not of great importance $z' \approx f$ if lens law is satisified: z'

$$\begin{bmatrix} x' \\ y' \\ z' \\ 1 \end{bmatrix} = \left[\begin{array}{ccc|c} x\ \text{scale} & & & \\ & y\ \text{scale} & & \text{translation} \\ & & z\ \text{scale} & \\ \hline & \text{perspective} & & \text{zoom} \end{array}\right] \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$

Camera transformation

$$\begin{bmatrix} u \\ v \\ t \end{bmatrix} = \underline{\underline{C}} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \qquad (*)$$

C is now 4x3, not 4x4.

t: not necessarily normalized to 1 (perspective transforms can change the "1").

Calibration Problem: given $(*)$ and enough (x, y, z) to (u, v) calibration data determine $\underline{\underline{C}}$

Define normalized camera (transverse) coordinates

$$U = \frac{u}{t} \qquad V = \frac{v}{t} \qquad (1)$$

U, V are what we would measure.

Break $(*)$ into three separate equations:

$$\left.\begin{aligned} u &= \underline{C_1} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \\ v &= \underline{C_2} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \\ t &= \underline{C_3} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \end{aligned}\right\} \quad (2)$$

Note that $\underline{C_1}, \underline{C_2}, \underline{C_3}$ are row vectors

$$\begin{aligned} \underline{C_1} &\ [c_{11} \ | \ c_{12} \ | \ c_{13} \ | \ c_{14}] \\ \underline{C_2} &\ [\quad \ | \ c_{22} \ | \quad \ | \quad] \\ \underline{C_3} &\ [c_{31} \ | \ c_{32} \ | \ c_{33} \ | \ c_{34}] \end{aligned}$$

Rewrite equations (1) as.

$$u - U t = 0 \qquad v - V t = 0$$

(U, V) are just numbers.

and substitute (2) into the above equations:

to get

$$\underline{c}_1 \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} - U \underline{c}_3 \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = 0 \qquad (3a)$$

$$\underline{c}_2 \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} - V \underline{c}_3 \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} = 0 \qquad (3b)$$

Now expand these equations using elements of $\underline{\underline{C}}$

from (3a) $c_{11}x + c_{12}y + c_{13}z + c_{14} - Uc_{31}x - Uc_{32}y - Uc_{33}z - Uc_{34} = 0$ (4a)

from (3b) $C_{21}x + C_{21}y + C_{23}z + C_{24} - VC_{31}x - VC_{32}y - VC_{33}z - VC_{34} = 0$ (4b)

Equation 4 represents 2 equations in 12 unknowns.

Since these are homogeneous equations pick $C_{34} = 1$, It will be our scaling factor. Represent (4) as a matrix:

$$
\underbrace{\begin{bmatrix}
x^{(1)} & y^{(1)} & z^{(1)} & 1 & 0 & 0 & 0 & 0 & -u^{(1)}x^{(1)} & -u^{(1)}y^{(1)} & -u^{(1)}z^{(1)} \\
0 & 0 & 0 & 0 & x^{(1)} & y^{(1)} & z^{(1)} & 1 & -v^{(1)}x^{(1)} & -v^{(1)}y^{(1)} & -v^{(1)}z^{(1)} \\
x^{(2)} & y^{(2)} & z^{(2)} & 1 & 0 & 0 & 0 & 0 & -u^{(2)}x^{(2)} & -u^{(2)}y^{(2)} & -u^{(2)}z^{(2)} \\
0 & 0 & 0 & 0 & x^{(2)} & y^{(2)} & z^{(2)} & 1 & -v^{(2)}x^{(2)} & -v^{(2)}y^{(2)} & -v^{(2)}z^{(2)} \\
 & & & & & \vdots & & & & &
\end{bmatrix}}_{\text{measured points in space}}
\underbrace{\begin{bmatrix}
c_{11} \\ c_{12} \\ c_{13} \\ c_{14} \\ c_{21} \\ c_{22} \\ c_{23} \\ c_{24} \\ c_{31} \\ c_{32} \\ c_{33} \\ c_{34} \to 1
\end{bmatrix}}_{\text{unknowns}}
=
\underbrace{\begin{bmatrix}
u^{(1)} \\ v^{(1)} \\ u^{(2)} \\ v^{(2)} \\ \vdots \\ u^{(n)} \\ v^{(n)}
\end{bmatrix}}_{\text{measured camera coordinates}}
$$

where

$$
\begin{aligned}
x^{(1)}\ y^{(1)}\ z^{(1)} &\to u^{(1)}\ v^{(1)} \\
x^{(2)}\ y^{(2)}\ z^{(2)} &\to u^{(2)}\ v^{(2)} \\
\text{etc.} &\quad\ \ \text{etc.}
\end{aligned}
$$

to get

$$u - Ut = 0 \qquad\qquad v - Vt = 0$$

$$[x\ y\ z\ 1]\,\underline{c_1} - U\,[x\ y\ z\ 1]\,\underline{c_3} = 0 \tag{3a}$$

$$[x\ y\ z\ 1]\,\underline{c_2} - V\,[x\ y\ z\ 1]\,\underline{c_3} = 0 \tag{3b}$$

Now expand these equations using elements of $\underline{\underline{C}}$

(3a): $x c_{11} + y c_{21} + z c_{31} + c_{41} - Ux c_{13} - Uy c_{23} - Uz c_{33} - U c_{43} = 0$ (

(3b): $x c_{12} + y c_{22} + z c_{32} + c_{42} - Vx c_{13} - Vy c_{23} - Vz c_{33} - V c_{43} = 0$ (

Equation (4) represents 2 equations in 12 unknowns, c_{11}, c_{42}, c_{13}

$c_{21}\ c_{22}\ c_{23}$

$c_{31}\ c_{32}\ c_{33}$

$c_{41}\ c_{42}\ c_{43}$

Since these are homogeneous coordinates, pick $c_{43} = 1$. It will be our scaling factor. Represent (4) as a matrix: ← This really doesn't get u a lot.

$$\underbrace{\begin{bmatrix} x^{(1)} & y^{(1)} & z^{(1)} & 1 & 0 & 0 & 0 & 0 & -U^{(1)}x^{(1)} & -U^{(1)}y^{(1)} & -U^{(1)}z^{(1)} \\ 0 & 0 & 0 & 0 & x^{(1)} & y^{(1)} & z^{(1)} & 1 & -V^{(1)}x^{(1)} & -V^{(1)}y^{(1)} & -V^{(1)}z^{(1)} \\ \hline x^{(2)} & y^{(2)} & z^{(2)} & 1 & 0 & 0 & 0 & 0 & -U^{(2)}x^{(2)} & -U^{(2)}y^{(2)} & -U^{(2)}z^{(2)} \\ 0 & 0 & 0 & 0 & x^{(2)} & y^{(2)} & z^{(2)} & 1 & -V^{(2)}x^{(2)} & -V^{(2)}y^{(2)} & -V^{(2)}z^{(2)} \\ \hline & & & & & \vdots & & & & & \end{bmatrix}}_{\text{known coordinates inspace!}} \underbrace{\begin{bmatrix} c_{11} \\ c_{21} \\ c_{31} \\ c_{41} \\ c_{12} \\ c_{22} \\ c_{32} \\ c_{42} \\ c_{13} \\ c_{23} \\ c_{33} \\ \cancel{c_{43}} \to 1 \end{bmatrix}}_{\text{unknowns}} = \underbrace{\begin{bmatrix} u^{(1)} \\ v^{(1)} \\ U^{(2)} \\ V^{(2)} \\ \vdots \\ u^{(n)} \\ v^{(n)} \end{bmatrix}}_{\text{measu camera coordinat}}$$

went to one.

where $[x^{(1)}\ y^{(1)}\ z^{(1)}\ 1] \longrightarrow [u^{(1)}\ v^{(1)}\ \ ]$

$[x^{(2)}\ y^{(2)}\ z^{(2)}\ 1] \longrightarrow [u^{(2)}\ v^{(2)}\ -\ ]$

etc. etc.

This can be solved <u>exactly</u> if we have 6 calibration pairs.

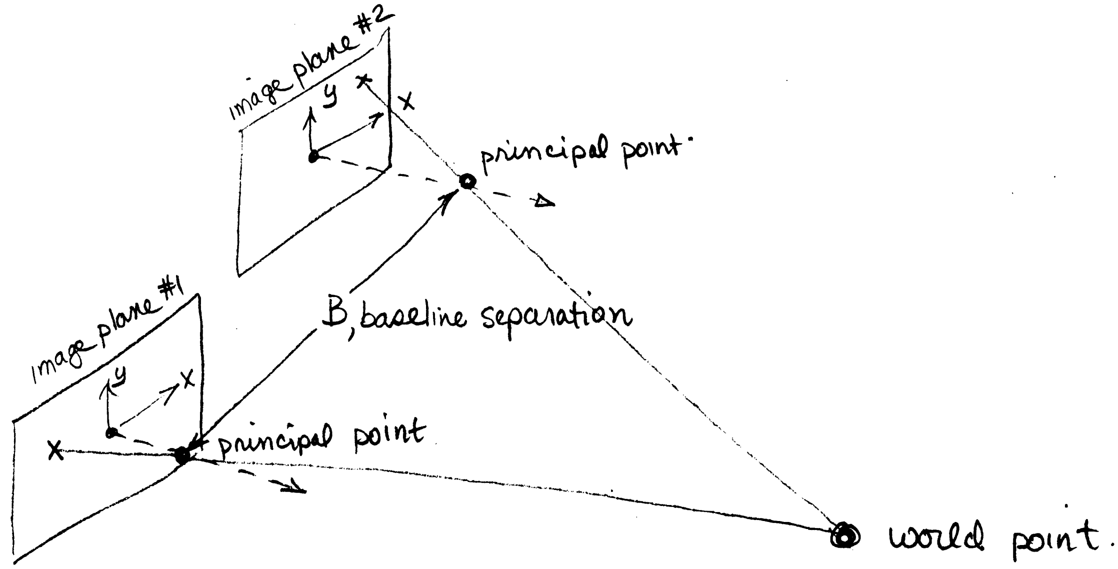
Typically, we have $n > 6$ and use statistical methods to solve for $\underline{\underline{C}}$

In general, pick 3 points on each of two different planes (so-called two plane calibration approach)

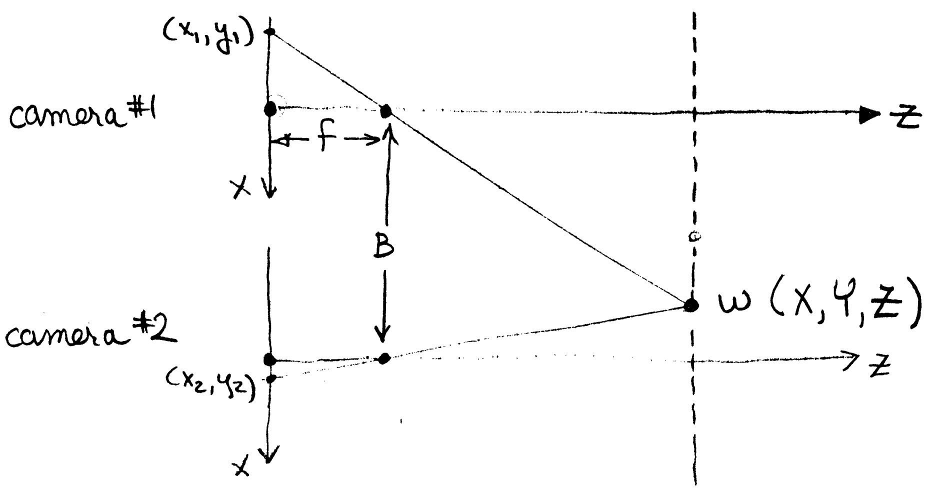
Stereo imaging:

single camera: many → one transformation

use two cameras to recover 3-D information



simple coordinate system:



two coordinate system

(X_1, Y_1, Z_1)

(X_2, Y_2, Z_2)

if cameras parallel Z will be same as well as $y = 0$.

simple geometry for camera #1: $x_1 = X_1 \dfrac{f}{f - Z_1}$

camera #2 $x_2 = X_2 \dfrac{f}{f - Z_2}$

assume cameras are identical

Then: $X_1 = \dfrac{f - Z_1}{f} x_1$ $\quad Z$

$X_2 = \dfrac{f - Z_2}{f} x_2$ $\quad Z$

making some simple observations :

$$Z_1 = Z_2 = Z$$

$$X_2 = X_1 + B$$

Then

$$\frac{f-Z}{f} x_2 = \frac{f-Z}{f} x_1 + B$$

$$(f-Z)(x_2 - x_1) = fB$$

$$f - Z = \frac{fB}{(x_2 - x_1)}$$

this is called the baseline

$$Z = f - \frac{fB}{(x_2 - x_1)}$$

measure in camera coordinate

Problems : ① finding corresponding points

② knowing coordinates exactly.

HOMEWORK ASSIGNMENT #2

Problem 1.

(a) Referring to Table 1 (below), determine the calibration parameters for the two cameras using the nine data points labeled 1, 3, 4, 5, 6, 8, 9, 10 and 11. Set $A_{134}=A_{234}=1.0$

(b) Using your calibration results from (a) and the image plane data for points 2 and 7, find the object point locations for these points.

	camera #1 image plane coordinates (pixels)		camera #2 image plane coordinates (pixels)		object coordinates in world coordinate system (inches)		
Point #	x_{i1}	y_{i1}	x_{i2}	y_{i2}	x_o	y_o	z_o
1	48	48	173	217	0.00	0.00	0.00
2	36	36	247	278			
3	36	48	223	283	1.72	1.95	0.00
4	48	36	217	223	0.00	0.05	0.24
5	36	24	280	275	0.64	3.04	0.00
6	24	24	339	356	2.76	5.88	0.00
7	48	24	257	232			
8	113	36	239	89	0.00	1.22	4.00
9	36	92	119	294	3.57	0.26	0.00
10	92	92	136	172	3.38	0.00	3.44
11	92	36	233	137	0.00	1.16	2.84

Solution : Let $\bar{x}_{ik} = A_k \bar{X}_o$ where k=1,2 and "i" denotes the image plane and "o" denotes the object plane (1)

where
$$A_{1} = \begin{bmatrix} a_{111} & a_{112} & a_{113} & a_{114} \\ a_{121} & a_{122} & a_{123} & a_{124} \\ a_{131} & a_{132} & a_{133} & a_{134} \end{bmatrix}$$
— a_{134}: set to 1, according to hint

The point $(w_i x_{i1}, w_i y_{i1}, w_i)$ in camera coordinates corresponds to $(x_o, y_o, z_o, 1)$ in world coordinates so (1) can be written

$$\begin{bmatrix} w_i x_{i1} \\ w_i y_{i1} \\ w_i \end{bmatrix} = \begin{bmatrix} a_{111} & a_{112} & a_{113} & a_{114} \\ a_{121} & a_{122} & a_{123} & a_{124} \\ a_{131} & a_{132} & a_{133} & a_{134} \end{bmatrix} \begin{bmatrix} x_o \\ y_o \\ z_o \\ 1 \end{bmatrix} \quad (2)$$

writing out and solving for x_i and y_i

$$w_i x_{i1} = a_{111} x_o + a_{112} y_o + a_{113} z_o + a_{114}$$
$$w_i y_{i1} = a_{121} x_o + a_{122} y_o + a_{123} z_o + a_{124} \quad (3)$$
$$w_i = a_{131} x_o + a_{132} y_o + a_{133} z_o + a_{134}$$

Solving for x_i and y_i, we can eliminate w_i

$$x_{i1} = \frac{a_{111} x_0 + a_{112} y_0 + a_{113} z_0 + a_{114}}{a_{131} x_0 + a_{132} y_0 + a_{133} z_0 + a_{134} \to 1}$$

$$y_{i1} = \frac{a_{121} x_0 + a_{122} y_0 + a_{123} z_0 + a_{124}}{a_{131} x_0 + a_{132} y_0 + a_{133} z_0 + a_{134} \to 1}$$

This can be written in the manner of the notes as a matrix equation using the coefficients a_{ijk} as variables

$$a_{111} x_0 + a_{112} y_0 + a_{113} z_0 + a_{114} - a_{131} x_{i1} x_0 - a_{132} x_{i1} y_0 - a_{133} x_{i1} z_0 - x_{i1} =$$

$$a_{121} x_0 + a_{122} y_0 + a_{123} z_0 + a_{124} - a_{131} y_{i1} x_0 - a_{132} y_{i1} y_0 - a_{133} y_{i1} z_0 - y_{i1}$$

$$\begin{bmatrix} x_0 & y_0 & z_0 & 1 & 0 & 0 & 0 & 0 & -x_{i1} x_0 & -x_{i1} y_0 & -x_{i1} z_0 \\ 0 & 0 & 0 & 0 & x_0 & y_0 & z_0 & 1 & -y_{i1} x_0 & -y_{i1} y_0 & -y_{i1} z_0 \end{bmatrix} \begin{bmatrix} a_{111} \\ a_{112} \\ a_{113} \\ a_{114} \\ a_{121} \\ a_{122} \\ a_{123} \\ a_{124} \\ a_{131} \\ a_{132} \\ a_{133} \end{bmatrix} = \begin{bmatrix} x_{i1} \\ y_{i1} \end{bmatrix}$$

This is 2 equations in 11 unknowns. We will need at least 6 points to solve this equation.

The $\underline{U}$ matrix for camera #1 is then

0	0	0	1.0000	0	0	0	0	0	0	0
0	0	0	0	0	0	0	1.0000	0	0	0
1.7200	1.9500	0	1.0000	0	0	0	0	-61.9200	-70.2000	0
0	0	0	0	1.7200	1.9500	0	1.0000	-82.5600	-93.6000	0
0	0.0500	0.2400	1.0000	0	0	0	0	0	-2.4000	-11.5200
0	0	0	0	0	0.0500	0.2400	1.0000	0	-1.8000	-8.6400
0.6400	3.0400	0	1.0000	0	0	0	0	-23.0400	-109.4400	0
0	0	0	0	0.6400	3.0400	0	1.0000	-15.3600	-72.9600	0
2.7600	5.8800	0	1.0000	0	0	0	0	-66.2400	-141.1200	0
0	0	0	0	2.7600	5.8800	0	1.0000	-66.2400	-141.1200	0
0	1.2200	4.0000	1.0000	0	0	0	0	0	-137.8600	-452.0000
0	0	0	0	0	1.2200	4.0000	1.0000	0	-43.9200	-144.0000
3.5700	0.2600	0	1.0000	0	0	0	0	-128.5200	-9.3600	0
0	0	0	0	3.5700	0.2600	0	1.0000	-328.4400	-23.9200	0
3.3800	0	3.4400	1.0000	0	0	0	0	-311.9740	0	-316.4800
0	0	0	0	3.3800	0	3.4400	1.0000	-310.9600	0	-316.4800
0	1.1600	2.8400	1.0000	0	0	0	0	0	-106.7200	-261.2800
0	0	0	0	0	1.1600	2.8400	1.0000	0	-41.7600	-102.2400

The X matrix is

48
48
36
48
48
36
36
24
24
24
113
36
36
92
92
92
92
36

Using the matlab function $\underline{a} = \underline{\underline{U}} \backslash \underline{x}$

The camera parameters are:

$-2.4721 = a_{111}$

$-3.1257 = a_{112}$

$17.4912 = a_{113}$

$45.7472 = a_{114}$

$14.6739 = a_{121}$

$-10.4759 = a_{122}$

$1.3911 = a_{123}$

$42.9902 = a_{124}$

$0.0081 = a_{131}$

$-0.0239 = a_{132}$

$0.0072 = a_{133}$ and $a_{134} = 1$

The U matrix for camera #2 is

differences between Thomas & mine

								all +	all +	all +
0	0	0	0.0010	0	0	0	0	0	0	0
0	0	0	0	0	0	0	0.0010	0	0	0
0.0017	0.0019 20	0	0.0010	0	0	0	0	−0.3836	−0.4348	0
0	0	0	0	0.0017	0.0019 20	0	0.0010	−0.4868	−0.5518	0
0	0.0000	0.0002	0.0010	0	0	0	0	0	−0.0108	−0.0521
0	0	0	0	0	0.0000	0.0002	0.0010	0	−0.0111	−0.0535
0.0006	0.0030	0	0.0010	0	0	0	0	−0.1792	−0.8512	0
0	0	0	0	0.0006	0.0030	0	0.0010	−0.1760	−0.8360	0
0.0028	0.0059	0	0.0010	0	0	0	0	−0.9356	−1.9933	0
0	0	0	0	0.0028	0.0059	0	0.0010	−0.9826	−2.0933	0
0	0.0012	0.0040	0.0010	0	0	0	0	0	−0.2916	−0.9560
0	0	0	0	0	0.0012	0.0040	0.0010	0	−0.1086	−0.3560
0.0036	0.0003	0	0.0010	0	0	0	0	−0.4248	−0.0309	0
0	0	0	0	0.0036	0.0003	0	0.0010	−1.0496	−0.0764	0
0.0034	0	0.0034	0.0010	0	0	0	0	−0.4597	0	−0.4678
0	0	0	0	0.0034	0	0.0034	0.0010	−0.5814	0	−0.5917
0	0.0012	0.0028	0.0010	0	0	0	0	0	−0.2703	−0.6617
0	0	0	0	0	0.0012	0.0028	0.0010	0	−0.1589	−0.3891

(all × 10^{+3})

The X matrix is

173
217
223
283
217
223
280
275
339
356
239
89
119
294
136
172
233
137

The camera parameters are then

$-19.7590 = a_{211}$

$27.3031 = a_{212}$

$5.3775 = a_{213}$

$191.8194 = a_{214}$

$22.1813 = a_{221}$

$4.8661 = a_{222}$

$-33.6712 = a_{223}$

$224.7052 = a_{224}$

$0.0103 = a_{231}$

$-0.0252 = a_{232}$

$0.0137 = a_{233}$ and $a_{234} = 1$.

To find points in three-dimensional space requires a simultaneous solution of both sets of equations as provided on page 1a.

For $x_{i1} = 48$, $y_{i1} = 24$, $x_{i2} = 257$ and $y_{i2} = 232$ ← point #7

```
»P1=[X1(1)-X1(9)*48,X1(2)-X1(10)*48,X1(3)-X1(11)*48;
X1(5)-X1(9)*24,X1(6)-X1(10)*24,X1(7)-X1(11)*24;
X2(1)-X2(9)*257,X2(2)-X2(10)*257,X2(3)-X2(11)*257;
X2(5)-X2(9)*232,X2(6)-X2(10)*232,X2(7)-X2(11)*232;]
```

} eqn (3), pg 1a

```
P1 =

   -2.8608   -1.9762   17.1437
   14.4796   -9.9011    1.2173
  -22.4002   33.7895    1.8671
   19.7970   10.7215  -36.8401

»F1=[48-X1(4);24-X1(8);257-X2(4);232-X2(8);]
```

} eqn (5), pg 1a

```
F1 =

    2.2528
  -18.9902
   65.1806
    7.2948

»X=inv(P1'*P1)*P1'*F1

X =

   -0.0385
    1.8868
    0.3324
```

} point #7 coordinates

NOTE:

X1(1) = A_{111}	X2(1) = A_{211}
X1(2) = A_{112}	X2(2) = A_{212}
X1(3) = A_{113}	X2(3) = A_{213}
X1(4) = A_{114}	X2(4) = A_{214}
etc.	etc.

For point #2 $\quad x_{i_1} = 36$

$y_{i_1} = 36$

$x_{i_2} = 247$

$y_{i_2} = 278$

```
»P2=[X1(1)-X1(9)*36,X1(2)-X1(10)*36,X1(3)-X1(11)*36;
X1(5)-X1(9)*36,X1(6)-X1(10)*36,X1(7)-X1(11)*36;
X2(1)-X2(9)*247,X2(2)-X2(10)*247,X2(3)-X2(11)*247;
X2(5)-X2(9)*278,X2(6)-X2(10)*278,X2(7)-X2(11)*278;]

P2 =

   -2.7636   -2.2636   17.2306
   14.3824   -9.6138    1.1305
  -22.2974   33.5371    2.0037
   19.3243   11.8825  -37.4684

»F2=[36-X1(4);36-X1(8);247-X2(4);278-X2(8);]

F2 =

   -9.7472
   -6.9902
   55.1806
   53.2948

»X=inv(P2'*P2)*P2'*F2

X =

    1.1335
    2.4044
   -0.0741
```

point #2's coordinates

and, as a check on our solution, let's do point #5

```
»P3=[X1(1)-X1(9)*36,X1(2)-X1(10)*36,X1(3)-X1(11)*36;
X1(5)-X1(9)*24,X1(6)-X1(10)*24,X1(7)-X1(11)*24;
X2(1)-X2(9)*280,X2(2)-X2(10)*280,X2(3)-X2(11)*280;
X2(5)-X2(9)*275,X2(6)-X2(10)*275,X2(7)-X2(11)*275;]
```

```
P3 =

   -2.7636   -2.2636   17.2306
   14.4796   -9.9011    1.2173
  -22.6366   34.3700    1.5530
   19.3551   11.8068  -37.4274

»F3=[36-X1(4);24-X1(8);280-X2(4);275-X2(8);]

F3 =

   -9.7472
  -18.9902
   88.1806
   50.2948
```

```
»X=inv(P3'*P3)*P3'*F3
```

X =	actual	max	error
0.7316	0.64	3.57	(.73−.64)/3.57 ≅ 3%
3.0420	3.04	5.88	(3.042−3.04)/5.88 ≅ .03%
−0.0127	0.00	4.00	(−0.0127−0)/4.00 ≅ 0.3%

»

in terms of the maximum distance in each dimension, the errors are.

The error (for this point) in y and z is outstanding, but the error in x is more typical of a vision system. Quantization error would be on the order of 0.3% (256 pixels).

This gives a set of equations of the form

$$\underline{\underline{U}}\,\underline{a} = \underline{x} \tag{4}$$

where $\underline{\underline{U}}$ = $2n \times 11$ matrix

$\underline{a}$ = vector of unknown calibration parameters (11×1)

$\underline{x}$ = image plane coordinate vector $(2n \times 1)$

where $n = 9$ for this example.

This system (9 points) is overdetermined and can be solved using a pseudoinverse

$$\underline{a} = \underbrace{\left(\underline{\underline{U}}^T \underline{\underline{U}}\right)}_{\text{pseudo inverse}}{}^{-1}\, \underline{\underline{U}}\,\underline{x} \tag{5}$$

or other method (In matlab $a = U \backslash x$)

The matrix U is given by

point #1	0	0	0	1	0	0	0	0	0	0	0
	0	0	0	0	0	0	0	1	0	0	0
point #3	1.72	1.95	0	1	0	0	0	0	−(36)(1.72)	−(36)(1.95)	0
	0	0	0	0	1.72	1.95	0	1	−(48)(1.72)	−(48)(1.95)	0
point #4	0	0.05	0.24	1	0	0	0	0	0	−(48)(.05)	−(48)(.2
	0	0	0	0	0	.05	.24	1	0	−(36)(.05)	−(36)(.
point #5	0.64	3.04	0	1	0	0	0	0	−(36)(.64)	−(36)(3.04)	0
	0	0	0	0	.64	3.04	0	1	−(24)(.64)	−(24)(3.04)	0
point #6	2.76	5.88	0	1	0	0	0	0	−(24)(2.76)	−(24)(5.88)	0
	0	0	0	0	2.76	5.88	0	1	−(24)(2.76)	−(24)(5.88)	0
pant #8	0	1.22	4	1	0	0	0	0	0	−(113)(1.22)	−(113)
	0	0	0	0	0	1.22	4	1	0	−(36)(1.22)	−(36)
~~point~~ #9	3.57	0.26	0	1	0	0	0	0	−(36)(3.57)	−(36)(.26)	0
	0	0	0	0	3.57	0.26	0	1	−(92)(3.57)	−(92)(0.26)	0
point #10	3.38	0	3.44	1	0	0	0	0	−(92)(3.38)	0	−(92)(3.
	0	0	0	0	3.38	0	3.44	1	−(92)(3.38)	0	−(92)(3.4
point #11	0	1.16	2.84	1	0	0	0	0	0	−(92)(1.16)	−(92)(2.
	0	0	0	0	0	1.16	2.84	1	0	−(36)(1.16)	−(36)(2.8

Note:

The general arrangement of two camera may be described using an extension of the camera calibration equations developed in class. Specifically, assume each camera is characterized by a 3×4 calibration matrix A_k where k=1,2 (one for each camera) and the elements of the matrix A_k are denoted by a_{ijk}. The relationship between the image plane and camera coordinates may then be written as

$$\hat{\mathbf{x}}_{ik} = A_k\hat{\mathbf{x}}_o, \; k=1,2 \qquad (1)$$

where $\hat{\mathbf{x}}_o$ is the vector of object coordinates and $\hat{\mathbf{x}}_{ik}$ is the vector of image plane coordinates. A vector is identified by bold type, the carats in Eqn. 1 denote that the vectors are specified in homogeneous coordinates. Since each matrix equation in (1) represents two equations in physical coordinatyes, we have, using two cameras, the case of four equations and three unknowns (x_o, y_o, z_o). Given two image points (one in each image plane) corresponding to the same physical point, expand each of the four equations in Eqn.1 in physical coordinates. Regrouping terms as coefficients of x_o, y_o, and z_o, a set of four equations in three unknowns can be written in matrix form as

$$P\hat{\mathbf{x}}_o = \mathbf{F} \qquad (2)$$

where the 4×3 matrix P is

$$P = \begin{bmatrix} a_{111} - a_{131}x_{i1} & a_{112} - a_{132}x_{i1} & a_{113} - a_{133}x_{i1} \\ a_{121} - a_{131}y_{i1} & a_{122} - a_{132}y_{i1} & a_{123} - a_{133}y_{i1} \\ a_{211} - a_{231}x_{i2} & a_{212} - a_{232}x_{i2} & a_{213} - a_{233}x_{i2} \\ a_{221} - a_{231}y_{i2} & a_{222} - a_{232}y_{i2} & a_{223} - a_{233}y_{i2} \end{bmatrix} \qquad (3)$$

and, in physical coordinates, the 3×1 object point that corresponds to image plane coordinate vectors $\mathbf{x}_{i1}$ and $\mathbf{y}_{i1}$ is

$$\mathbf{x}_o = \begin{bmatrix} x_o \\ y_o \\ z_o \end{bmatrix} \qquad (4)$$

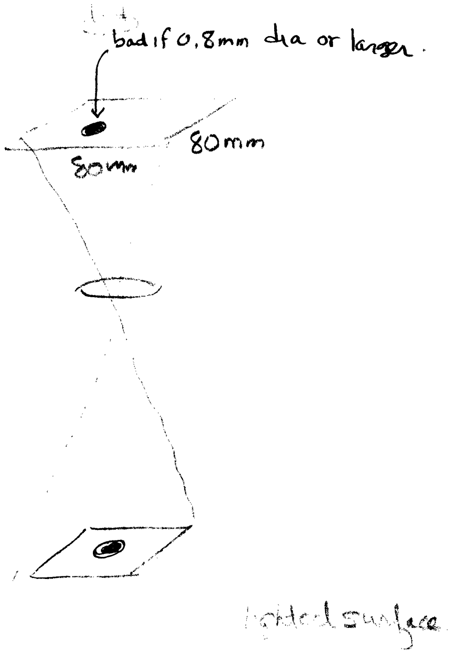
The 4×1 vector **F** is

$$\mathbf{F} = \begin{bmatrix} a_{134}x_{i1} - a_{114} \\ a_{134}y_{i1} - a_{124} \\ a_{234}x_{i2} - a_{214} \\ a_{234}y_{i2} - a_{224} \end{bmatrix} \qquad (5)$$

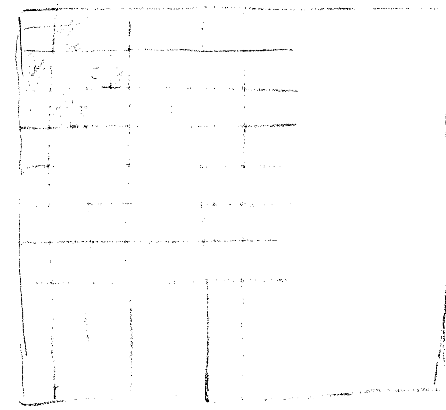
Equation 5 may be directly solved for a least squares estimate of $\mathbf{x}_o$, denoted as $\mathbf{x}_o^e$, by forming the pseudoinverse of P, denoted P^+, yielding the object point estimate:

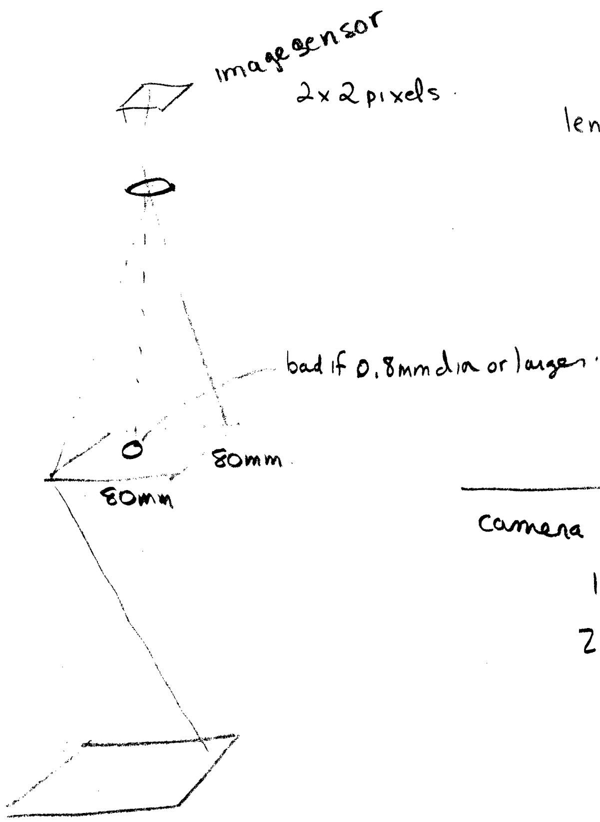
$$\mathbf{x}_o^e = P^+\mathbf{F} \qquad (6)$$

Alternately, the use of the Q-R decomposition approach or the elimination of one equation and a single matrix inversion may be employed. Note that although we have assumed the knowledge of corresponding image point locations, not all of this information is required to estimate $\mathbf{x}_o$. You may be able to determine $\mathbf{x}_o$ using only three of these four equations, assuming that the remaining three equations are linearly independent.



checkerboard ?

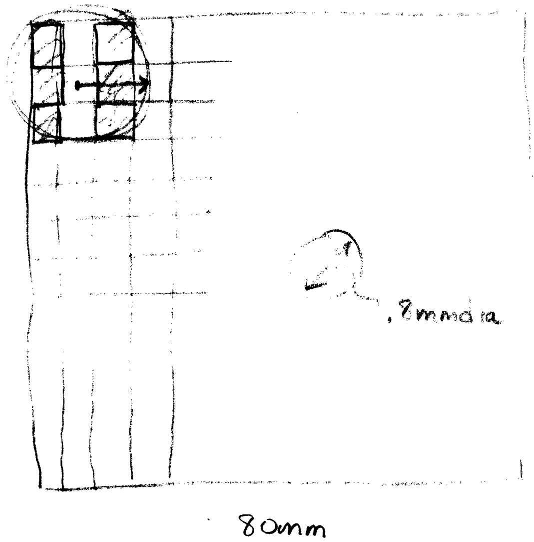


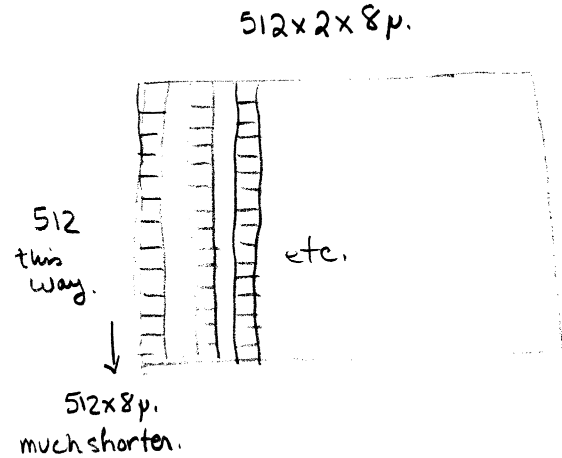


lenses		
	25	35
	50	70
	75	105
	100	140
	125	175
	150	
	175	
	200	

---

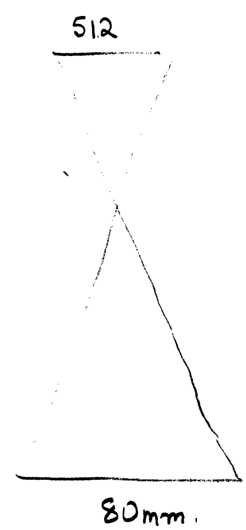
camera 512x512
1024 x 1024
2048 x 2048





want .8mm dia to correspond to at least $3 \times 8 = 24 \mu m$.

gives magnification $\frac{1}{m} = \frac{.8mm \text{ physical}}{.024mm \text{ image plane}}$



$$m = \frac{.024}{.800} = .03$$

this gives for the required camera.

$$\frac{3 \times \cancel{8\mu}}{N \times \cancel{8\mu}} = \frac{.8mm}{80mm}$$

$$\frac{3}{N} = \frac{.8}{80}$$

$$N = \frac{80 \cdot 3}{0.8} = 300.$$

$\Rightarrow$ since I need 3, 8μ per .8mm dia blob
to see 80 mm I need at least 100 times this or 300 squares across.
This implies That I can get by with lowest resolution camera 512x512 with 50% empty space.