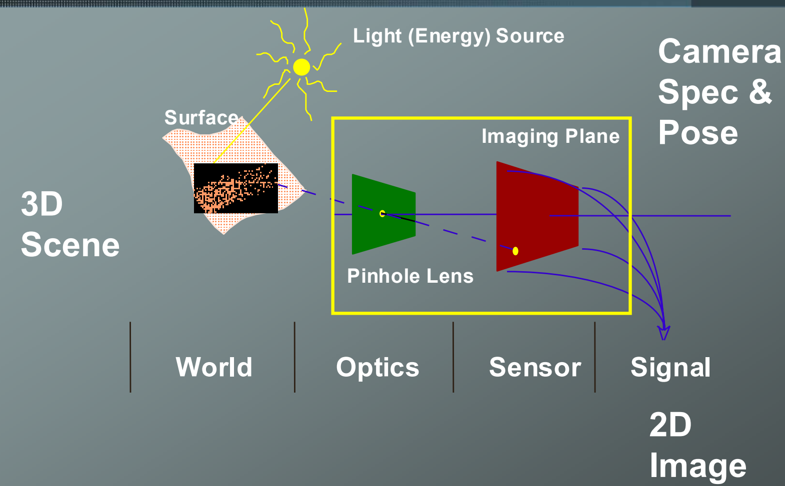
2. Camera Geometry

Computer Vision

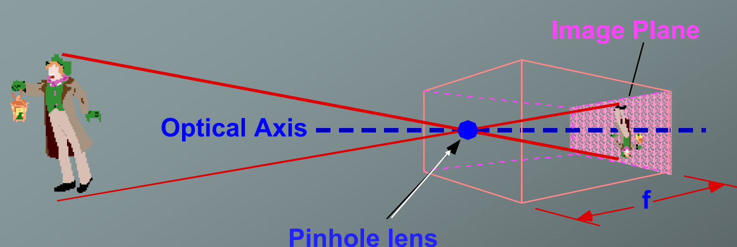
Zoltan Kato

http://www.inf.u-szeged.hu/~kato/

Image Formation



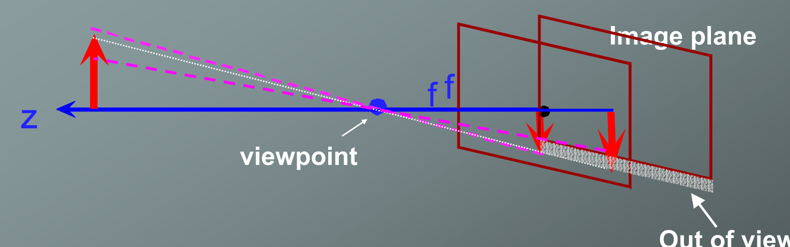
Pinhole Camera Model



- Pin-hole is the basis for most graphics and vision
  - Derived from physical construction of early cameras
  - Mathematics is very straightforward
- 3D World projected to 2D Image
  - Image inverted, size reduced
  - Image is a 2D plane: No direct depth information
- Perspective projection
  - f called the focal length of the lens

Focal Length, Field Of View

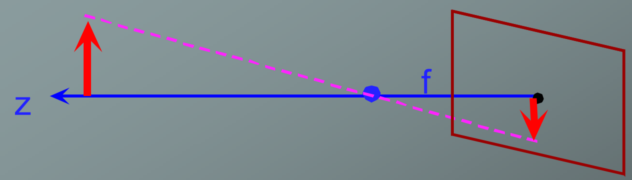
- Consider case with object on the optical axis:



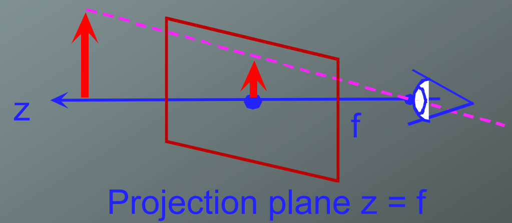
- Optical axis: the direction of imaging
- Image plane: a plane perpendicular to the optical axis
- Center of Projection (pinhole), focal point, viewpoint, nodal point
- Focal length: distance from focal point to the image plane
- FOV : Field of View – viewing angles in horizontal and vertical directions
- Increasing f will enlarge figures, but decrease FOV

Equivalent Geometry

- Consider case with object on the optical axis:



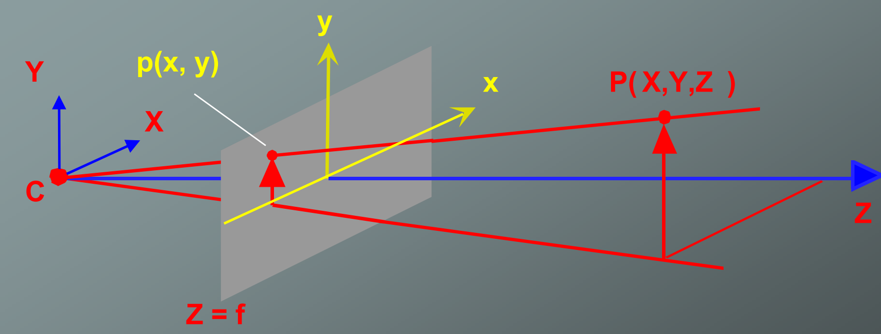
- More convenient with upright image:



- Equivalent mathematically

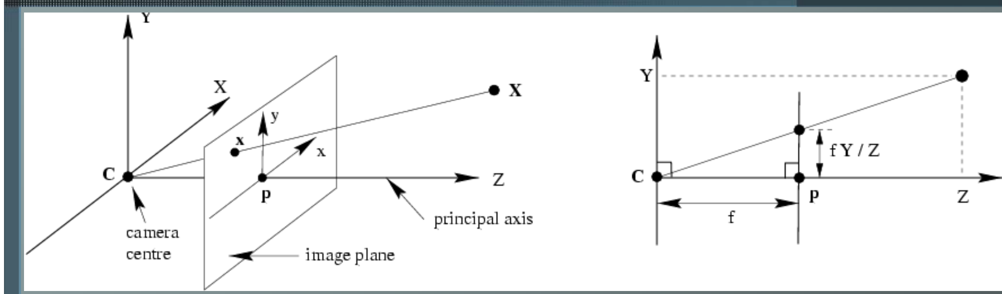
Perspective Projection

- Compute the image coordinates of **p** in terms of the camera coordinates of **P**.



- Origin of camera at center of projection
- **Z** axis along optical axis
- Image Plane at **Z = f**; **x || X** and **y||Y**

Pinhole camera model

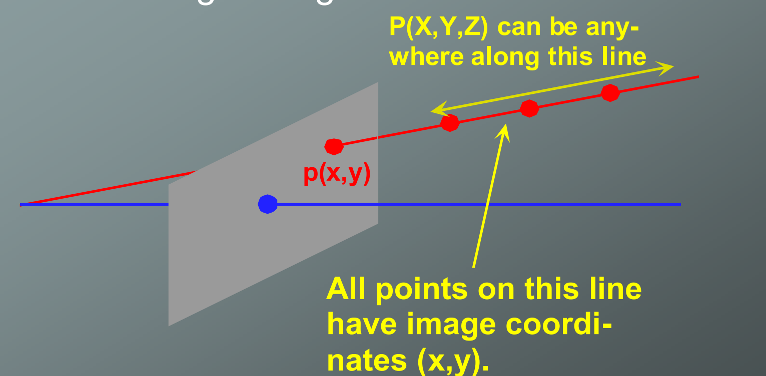


$$\begin{pmatrix}X\\Y\\Z\end{pmatrix}\mapsto\begin{pmatrix}f\dfrac{X}{Z}\\[2mm] f\dfrac{Y}{Z}\end{pmatrix}$$

$$\begin{pmatrix}X\\Y\\Z\\1\end{pmatrix}\mapsto\begin{pmatrix}fX\\fY\\Z\end{pmatrix}=\begin{bmatrix}f&&&0\\&f&&0\\&&1&0\end{bmatrix}\begin{pmatrix}X\\Y\\Z\\1\end{pmatrix}$$

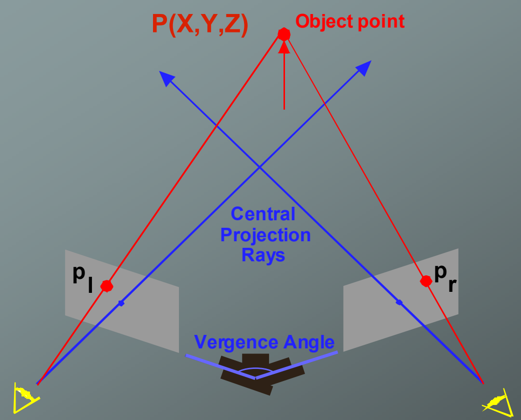
Reverse Projection

- Given a center of projection and image coordinates of a point, it is not possible to recover the 3D depth of the point from a single image.



In general, at least two images of the same point taken from two different locations are required to recover depth.

Stereo Geometry



- Depth obtained by triangulation
- Correspondence problem: p_l and p_r must correspond to the left and right projections of P, respectively.

Pinhole camera image

Amsterdam : what do you see in this picture?

- straight line
- size
- parallelism/angle
- shape
- shape of planes
- depth

Photo by Robert Kosara, robert@kosara.net http://www.kosara.net/gallery/pinholeamsterdam/pic01.html

Pinhole camera image

Amsterdam

- ✓straight line
- size
- parallelism/angle
- shape
- shape of planes
- depth

Photo by Robert Kosara, robert@kosara.net http://www.kosara.net/gallery/pinholeamsterdam/pic01.html

Pinhole camera image

Amsterdam

- ✓straight line
- ×size
- parallelism/angle
- shape
- shape of planes
- depth

Photo by Robert Kosara, robert@kosara.net http://www.kosara.net/gallery/pinholeamsterdam/pic01.html

Pinhole camera image

Amsterdam

- ✓straight line
- ×size
- ×parallelism/angle
- shape
- shape of planes
- depth

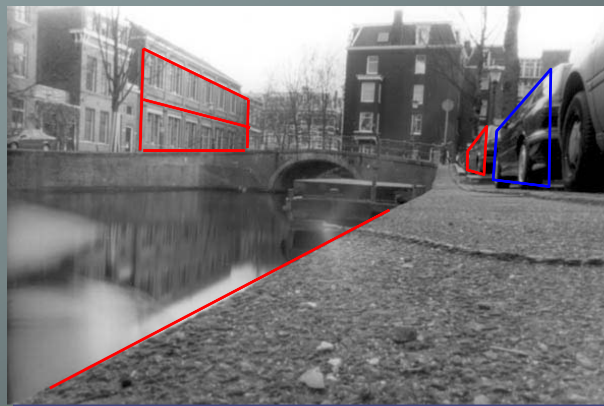


Photo by Robert Kosara, robert@kosara.net
http://www.kosara.net/gallery/pinholeamsterdam/pic01.html

Pinhole camera image

Amsterdam

- ✓straight line
- ×size
- ×parallelism/angle
- ×shape
- shape of planes
- depth

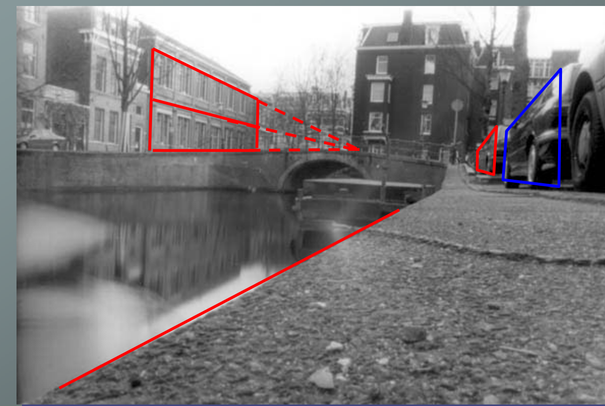


Photo by Robert Kosara, robert@kosara.net
http://www.kosara.net/gallery/pinholeamsterdam/pic01.html

Pinhole camera image

Amsterdam

- ✓straight line
- ×size
- ×parallelism/angle
- ×shape
- shape of planes
- ✓ parallel to image
- depth

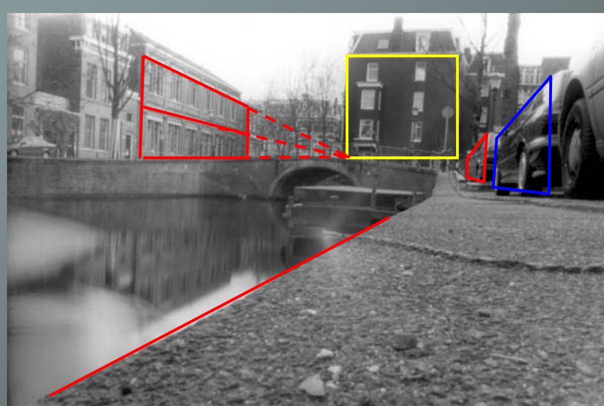


Photo by Robert Kosara, robert@kosara.net
http://www.kosara.net/gallery/pinholeamsterdam/pic01.html

Pinhole camera image

Amsterdam: what do you see?

- ✓straight line
- ×size
- ×parallelism/angle
- ×shape
- shape of planes
- ✓ parallel to image
- Depth ?
  - stereo
  - motion
  - size
  - structure …

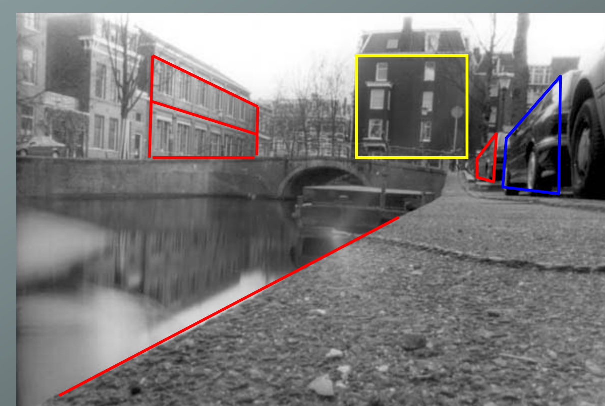
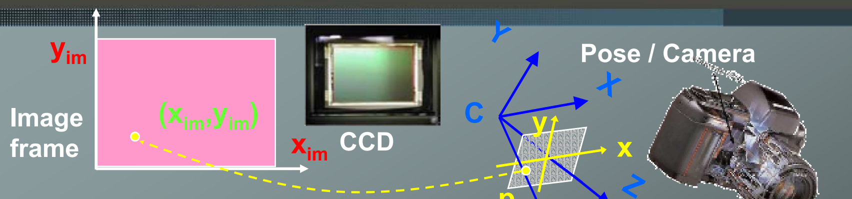
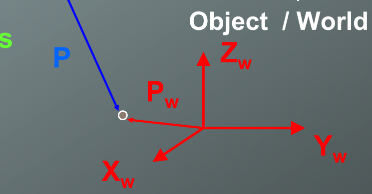


Photo by Robert Kosara, robert@kosara.net
http://www.kosara.net/gallery/pinholeamsterdam/pic01.html

- We see spatial shapes rather than individual pixels
- Knowledge: top-down vision belongs to human
- Stereo &Motion most successful in 3D CV & application
- You can see it but you don't know how…

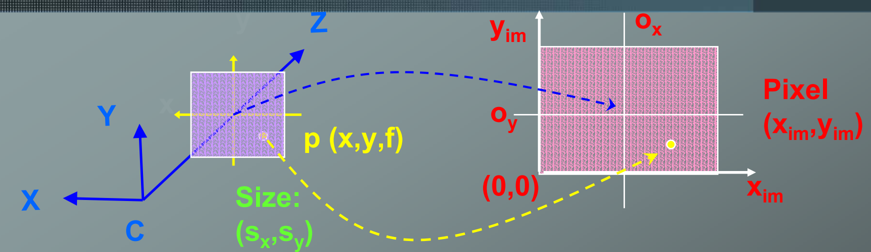
Camera Parameters





- Coordinate Systems
  - Frame coordinates (x_{im}, y_{im}) pixels
  - Image coordinates (x,y)
  - Camera coordinates (X,Y,Z)
  - World coordinates (X_w,Y_w,Z_w)
- Camera Parameters
  - **Intrinsic Parameters** (of the camera and the frame grabber): link the **frame coordinates** of an image point with its corresponding **camera coordinates**
  - **Extrinsic parameters**: define the location and orientation of the **camera coordinate system** with respect to the **world coordinate system**

Intrinsic Parameters (1)



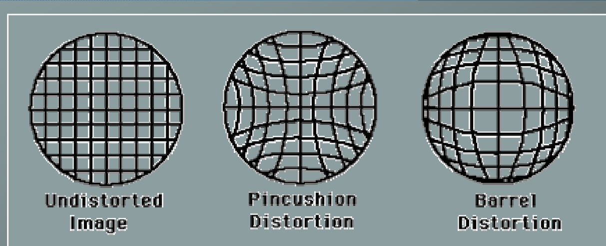
$$\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} fX + Zo_x \\ fY + Zo_y \\ Z \end{pmatrix} = \begin{bmatrix} fs_x & & o_x & 0 \\ & fs_y & o_y & 0 \\ & & 1 & 0 \end{bmatrix} \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

- From frame to image
  - Image center
  - Directions of axes
  - Pixel size
- Intrinsic Parameters
  - (o_x,o_y): principal point (image center)
  - (s_x,s_y): effective size of the pixel
  - f: focal length

Intrinsic Parameters (2)

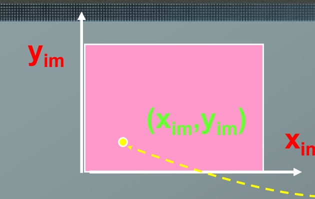
$(x, y) \xleftarrow{k_1, k_2} (xd, yd)$

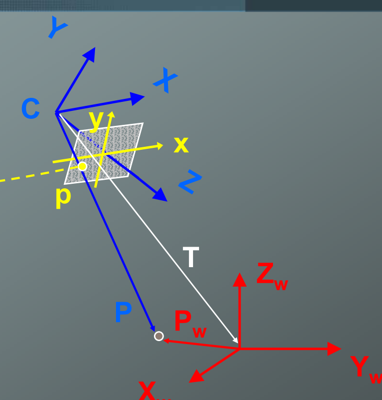
$$x = x_d(1 + k_1 r^2 + k_2 r^4)$$
$$y = y_d(1 + k_1 r^2 + k_2 r^4)$$



- Lens Distortions
  - Modeled as simple radial distortions
    - $r^2 = x_d^2 + y_d^2$
    - (x_d, y_d) distorted points
    - k_1, k_2: distortion coefficients
  - A model with $k_2 = 0$ is still accurate for a CCD sensor of 500x500 with ~5 pixels distortion on the outer boundary

Camera rotation and translation





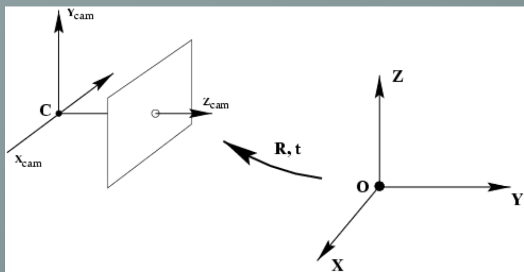
- From World to Camera

$$\mathbf{P} = \mathbf{R}\,\mathbf{P_w} + \mathbf{T}$$

- Extrinsic Parameters
  - A 3D translation vector, **T**, describing the relative locations of the origins of the two coordinate systems
  - A 3x3 rotation matrix, **R**, an orthogonal matrix that brings the corresponding axes of the two systems onto each other

$$\mathbf{R}^{-1} = \mathbf{R}^T, i.e.\ \mathbf{R}\mathbf{R}^T = \mathbf{R}^T\mathbf{R} = \mathbf{I}$$

Extrinsic Parameters: [R | T]



$$\tilde{X}_{cam} = R(\tilde{X} - \tilde{C})$$
$$x = K[I|0]X_{cam}$$
$$x = KR[I|-\tilde{C}]X$$

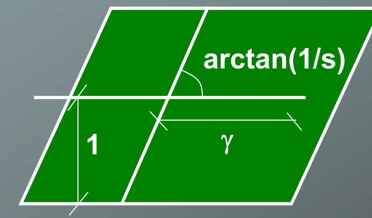
$$X_{cam} = \begin{bmatrix} R & -R\tilde{C} \\ 0 & 1 \end{bmatrix}\begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix} = \begin{bmatrix} R & -RC \\ 0 & 1 \end{bmatrix} X$$

$\tilde{C}$ ➔ Projection center in world coordinate frame.

$x = PX$ $\quad P = K[R|T]$ $\quad T = -R\tilde{C}$

Finite projective camera

$$K = \begin{bmatrix} f_x & s & p_x \\ & f_y & p_y \\ & & 1 \end{bmatrix}$$



s ➔ skew
for CCD/CMOS, always **s=0**

$P = KR[I|-\tilde{C}]$ 11 degree of freedom (5+3+3)

non-singular

decompose **P** in **K**,**R**,**C**?

$P = [M|p_4]$ $\quad [K,R] = RQ(M)$ $\quad \tilde{C} = -M^{-1}p_4$

$\tilde{C}$ ➔ Projection center in world coordinate frame.

{finite cameras}={P_{4x3} | det M≠0}

If **rank P=3**, but **rank M<3**, then camera is at infinity

Camera matrix decomposition

Finding the camera center

$PC = 0$ (use SVD to find null-space)

Algebraically, **C** may be obtained as:

$X = \det([p_2, p_3, p_4])$ $\quad Y = -\det([p_1, p_3, p_4])$

$Z = \det([p_1, p_2, p_4])$ $\quad W = -\det([p_1, p_2, p_3])$

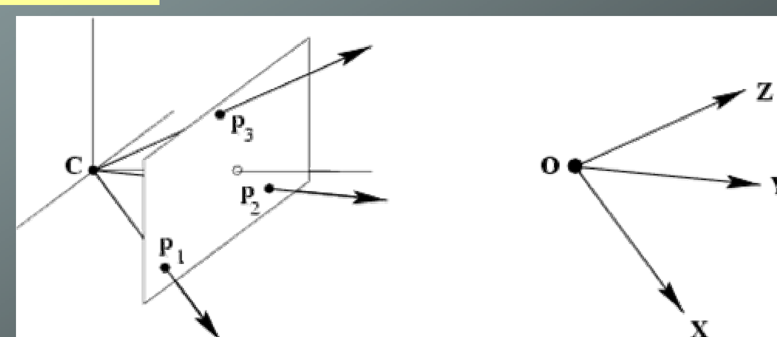
Finding the camera orientation and internal parameters

$M = KR$ (use RQ decomposition ~QR)
(if only QR, invert)

Column vectors

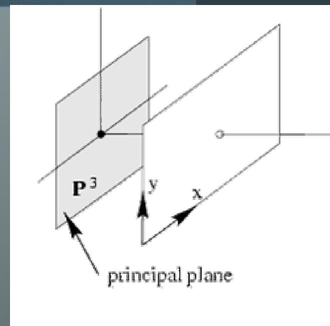
$$[p_2] = [p_1 p_2 p_3 p_4]\begin{bmatrix} 0 \\ 1 \\ 0 \\ 0 \end{bmatrix}$$

Image points corresponding to **X**,**Y**,**Z** directions and origin (**p_4**)

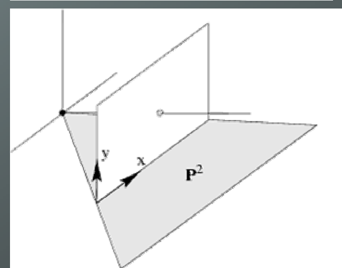


Row vectors

$$\begin{bmatrix} x \\ y \\ 0 \end{bmatrix} = \begin{bmatrix} \mathrm{p}^{1\mathrm{T}} \\ \mathrm{p}^{2\mathrm{T}} \\ \mathrm{p}^{3\mathrm{T}} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

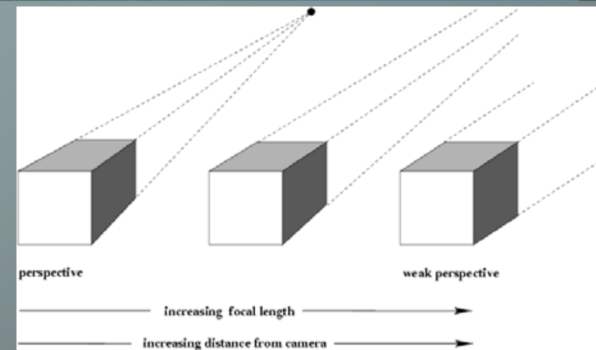


$$\begin{bmatrix} 0 \\ y \\ w \end{bmatrix} = \begin{bmatrix} \mathrm{p}^{1\mathrm{T}} \\ \mathrm{p}^{2\mathrm{T}} \\ \mathrm{p}^{3\mathrm{T}} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$



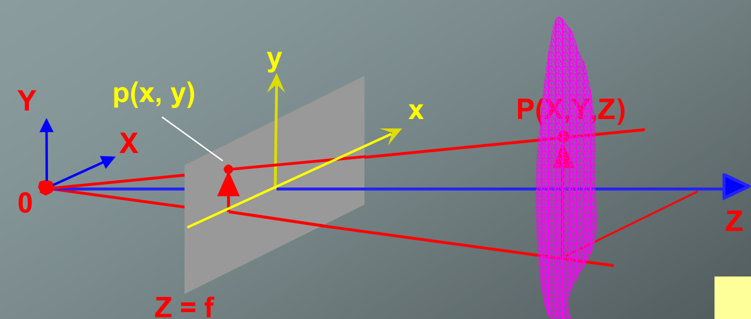
note: $\mathbf{p}^1$,$\mathbf{p}^2$ dependent on image reparametrization

Affine cameras



Weak Perspective Projection

- Average depth $\bar{Z}$ is much larger than the relative distance between any two scene points measured along the optical axis



- A sequence of two transformations
  - Orthographic projection : parallel rays
  - Isotropic scaling : $f/\bar{Z}$
- Linear Model
  - Preserve angles and shapes

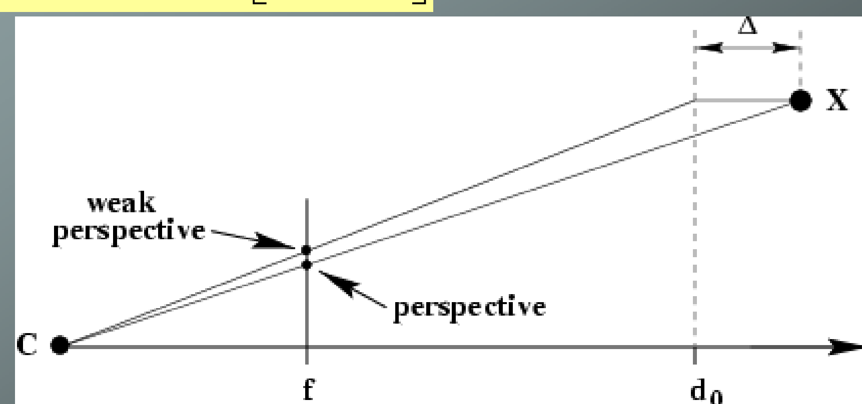
$$x = \frac{f}{\bar{Z}} X$$

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

Weak perspective projection

$$\mathrm{P}_{\infty} = \begin{bmatrix} \alpha_x & & \\ & \alpha_y & \\ & & 1 \end{bmatrix} \begin{bmatrix} \mathrm{r}^{1\mathrm{T}} & t_1 \\ \mathrm{r}^{2\mathrm{T}} & t_2 \\ 0 & 1/k \end{bmatrix}$$

(7 degrees of freedom)



Affine camera

$$P_A = \begin{bmatrix} \alpha_x & s & \\ & \alpha_y & \\ & & 1 \end{bmatrix} \begin{bmatrix} r^{1T} & t_1 \\ r^{1T} & t_2 \\ 0 & 1/k \end{bmatrix} \qquad P_A = \begin{bmatrix} m_{11} & m_{12} & m_{13} & t_1 \\ m_{21} & m_{22} & m_{23} & t_2 \\ 0 & 0 & 0 & 1 \end{bmatrix} \quad \text{(8dof)}$$

$$P_A = [3 \times 3 \text{ affine}] \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} [4 \times 4 \text{ affine}]$$

1. Affine camera=camera with principal plane coinciding with the plan at infinity Π_∞
2. Affine camera maps parallel lines to parallel lines
3. No center of projection, but direction of projection $P_A D=0$ (point on Π_∞)

Slides adopted from:

CS 395/495-25: Spring 2004

IBMR:

$R^3 \rightarrow R^2$ and $P^3 \rightarrow P^2$:

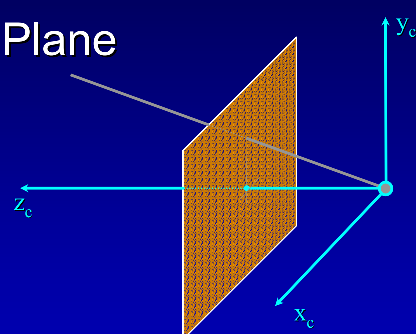
The Projective Camera Matrix

Jack Tumblin
jet@cs.northwestern.edu

Cameras Revisited

Plenty of Terminology:

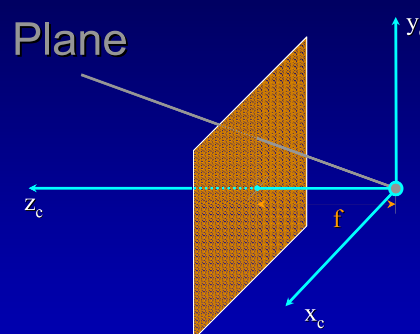
- Image Plane or Focal Plane
- Focal Distance f
- Camera Center C
- Principal Point
- Principal Axis
- Principal Plane
- Camera Coords (x_c,y_c,z_c)
- Image Coords (x',y')



Cameras Revisited

Plenty of Terminology:

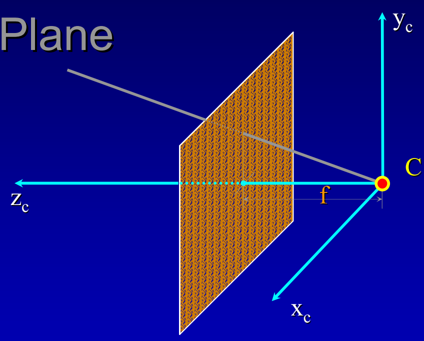
- Image Plane or Focal Plane
- Focal Distance f
- Camera Center C
- Principal Point
- Principal Axis
- Principal Plane
- Camera Coords (x_c,y_c,z_c)
- Image Coords (x',y')



Cameras Revisited

Plenty of Terminology:

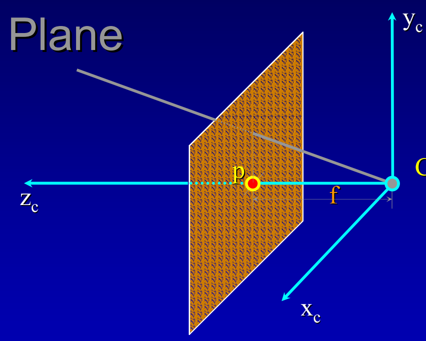
- Image Plane or Focal Plane
- Focal Distance f
- Camera Center C
- Principal Point
- Principal Axis
- Principal Plane
- Camera Coords (x_c, y_c, z_c)
- Image Coords (x', y')



Cameras Revisited

Plenty of Terminology:

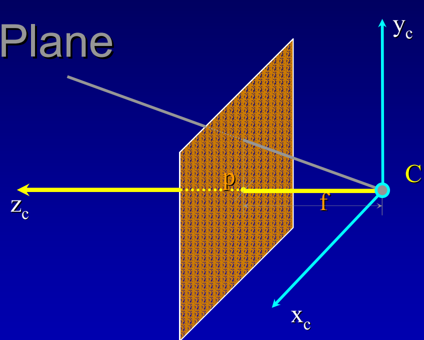
- Image Plane or Focal Plane
- Focal Distance f
- Camera Center C
- Principal Point p
- Principal Axis
- Principal Plane
- Camera Coords (x_c, y_c, z_c)
- Image Coords (x', y')



Cameras Revisited

Plenty of Terminology:

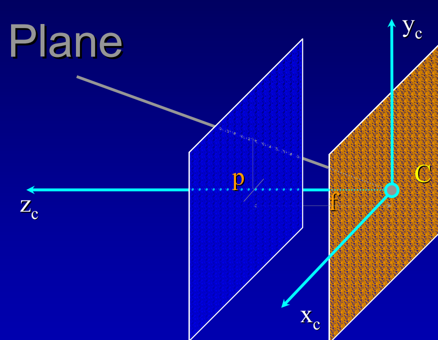
- Image Plane or Focal Plane
- Focal Distance f
- Camera Center C
- Principal Point p
- Principal Axis
- Principal Plane
- Camera Coords (x_c, y_c, z_c)
- Image Coords (x', y')



Cameras Revisited

Plenty of Terminology:

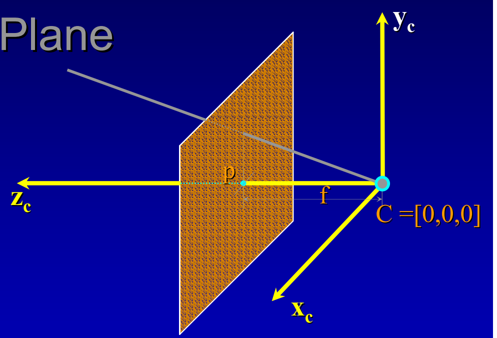
- Image Plane or Focal Plane
- Focal Distance f
- Camera Center C
- Principal Point p
- Principal Axis
- Principal Plane
- Camera Coords (x_c, y_c, z_c)
- Image Coords (x', y')



Cameras Revisited

Plenty of Terminology:

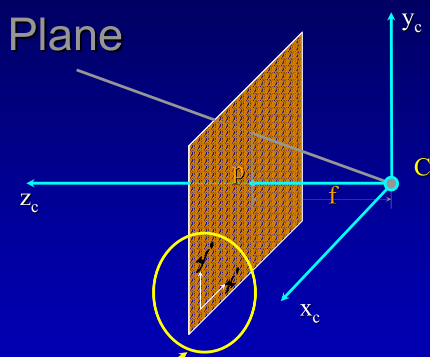
- Image Plane or Focal Plane
- Focal Distance f
- Camera Center C
- Principal Point p
- Principal Axis
- Principal Plane
- Camera Coords (x_c, y_c, z_c)
- Image Coords (x',y')



Cameras Revisited

Plenty of Terminology:

- Image Plane or Focal Plane
- Focal Distance f
- Camera Center C
- Principal Point p
- Principal Axis
- Principal Plane
- Camera Coords (x_c, y_c, z_c)
- Image Coords (x',y')

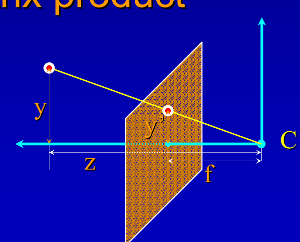


Cameras Revisited

- Goal: Formalize projective 3D→2D mapping
- Homogeneous coords handles infinities well:
  - Projective cameras (convergent 'eye' rays)
  - Affine cameras (parallel 'eye' rays)
  - Composed, controlled as matrix product
- Recall Euclidian $R^3 \rightarrow R^2$:

$$x' = f\,x / z$$
$$y' = f\,y / z$$

(Much Better: $P^3 \rightarrow P^2$)



Basic Camera P_0: $P^3 \rightarrow P^2$ (or camera R^3)

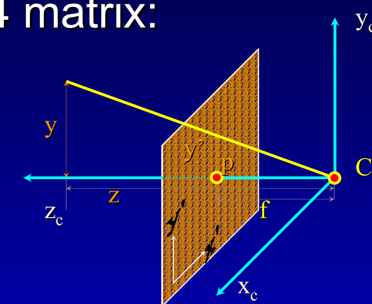
- Basic Camera P_0 is a 3x4 matrix:

$$\begin{bmatrix}\alpha_x f & s & p_x & 0\\ 0 & \alpha_y f & p_y & 0\\ 0 & 0 & 1 & 0\end{bmatrix}\begin{bmatrix}x_c\\ y_c\\ z_c\\ 1\end{bmatrix} = \begin{bmatrix}x\\ y\\ z\end{bmatrix}$$

K (3x3 submatrix)

$$P_0\,X = x$$

$$\boxed{[K \,|\, 0] = P_0}$$



- Non-square pixels? change scaling (α_x, α_y)
- Parallelogram pixels? set nonzero skew **s**

K matrix: "(internal) camera calib. matrix"

Complete Camera Matrix P

- K matrix: "internal camera calib. matrix"
- R·T matrix: "external camera calib. matrix"
  - T matrix: Translate world to cam. origin
  - R matrix: 3D rotate world to fit cam. axes

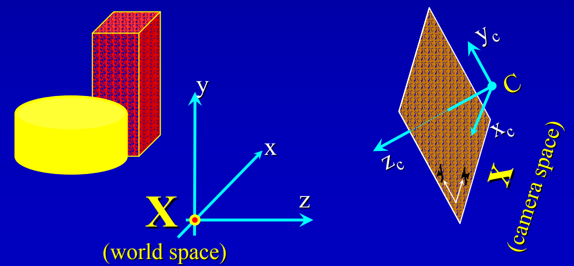
Input: **X** 3D World Space → Output: **x** 2D Camera Image

Combine: write

$(P_0 \cdot R \cdot T)\cdot X = x$

as

$\boxed{P\cdot X = x}$

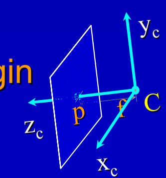


The Pieces of Camera Matrix P

$$P\cdot X = x,\ \text{or}\ \begin{bmatrix} \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot\ P & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{bmatrix}\begin{bmatrix} x_w \\ y_w \\ z_w \\ t_w \end{bmatrix} = \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} \qquad P = \begin{bmatrix} p_1 & p_2 & p_3 & p_4 \end{bmatrix}$$

Columns of P matrix = image of world-space axes:

- p_1, p_2, p_3 == image of x,y,z axis vanishing points
  - Direction D = $[1\ 0\ 0\ 0]^T$ = point on P^3's x_1 axis, at inifinity
  - PD = 1st column of P = P^1. Repeat for y and z axes.
- p_4 == image of the world-space origin pt.
  - Proof: let **D** = [0 0 0 1]T = direction to world origin
  - **PD** = 4th column of **P** = image of origin pt.

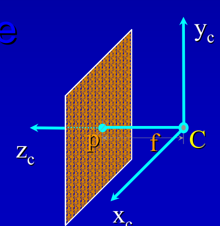


The Pieces of Camera Matrix P

$$P\cdot X = x,\ \text{or}\ \begin{bmatrix} \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot\ P & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{bmatrix}\begin{bmatrix} x_w \\ y_w \\ z_w \\ t_w \end{bmatrix} = \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} \qquad P = \begin{bmatrix} P^{1T} & \cdot & \cdot \\ \hline P^{2T} & \cdot & \cdot \\ \hline P^{3T} & \cdot & \cdot \end{bmatrix}$$

Rows of P matrix: camera planes in world space

- row 1 = **P**^{1T} = image x-axis plane
- row 2 = **P**^{2T} = image y-axis plane
- row 1 = **P**^{3T} = camera's principal plane



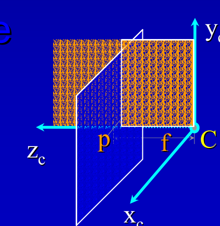
The Pieces of Camera Matrix P

$$P\cdot X = x,\ \text{or}\ \begin{bmatrix} \cdot & \cdot & \cdot & \cdot \\ \cdot & \cdot\ P & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{bmatrix}\begin{bmatrix} x_w \\ y_w \\ z_w \\ t_w \end{bmatrix} = \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix} \qquad P = \begin{bmatrix} P^{1T} & \cdot & \cdot \\ \hline P^{2T} & \cdot & \cdot \\ \hline P^{3T} & \cdot & \cdot \end{bmatrix}$$

Rows of P matrix: planes in world space

- row 1 = **P**^{1} = world plane whose image is x=0
- row 2 = **P**^{2} = world plane whose image is y=0
- row 3 = **P**^{3} = camera's principal plane

Why? Recall that in P^3, point X is on plane π if and only if $\pi^T\cdot X = 0$, so…

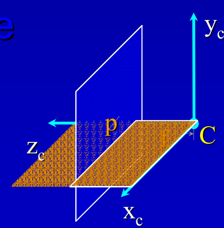


The Pieces of Camera Matrix P

$\mathbf{P} \cdot \mathbf{X} = \mathbf{x}$, or $\left[\; \mathbf{P} \; \right] \begin{bmatrix} x_w \\ y_w \\ z_w \\ t_w \end{bmatrix} = \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix}$ $\quad \mathbf{P} = \begin{bmatrix} \mathbf{P}^{1T} \\ \mathbf{P}^{2T} \\ \mathbf{P}^{3T} \end{bmatrix}$

Rows of P matrix: planes in world space

- row 1 = $\mathbf{P}^1$ = world plane whose image is x=0
- row 2 = $\mathbf{P}^2$ = world plane whose image is y=0
- row 3 = $\mathbf{P}^3$ = camera's principal plane

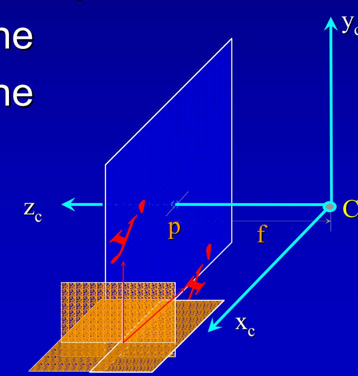


The Pieces of Camera Matrix P

$\mathbf{P} \cdot \mathbf{X} = \mathbf{x}$, or $\left[\; \mathbf{P} \; \right] \begin{bmatrix} x_w \\ y_w \\ z_w \\ t_w \end{bmatrix} = \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix}$ $\quad \mathbf{P} = \begin{bmatrix} \mathbf{P}^{1T} \\ \mathbf{P}^{2T} \\ \mathbf{P}^{3T} \end{bmatrix}$

Rows of P matrix: planes in world space

- row 1 = $\mathbf{P}^1$ = image's x=0 plane
- row 2 = $\mathbf{P}^2$ = image's y=0 plane
- Careful! Shifting the image origin by p_x, p_y shifts the x=0,y=0 planes!

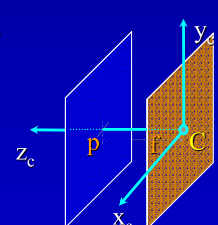


The Pieces of Camera Matrix P

$\mathbf{P} \cdot \mathbf{X} = \mathbf{x}$, or $\left[\; \mathbf{P} \; \right] \begin{bmatrix} x_w \\ y_w \\ z_w \\ t_w \end{bmatrix} = \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix}$ $\quad \mathbf{P} = \begin{bmatrix} \mathbf{P}^{1T} \\ \mathbf{P}^{2T} \\ \mathbf{P}^{3T} \end{bmatrix}$

Rows of P matrix: planes in world space

- row 1 = $\mathbf{P}^1$ = image's x=0 plane
- row 2 = $\mathbf{P}^2$ = image's y=0 plane
- row 3 = $\mathbf{P}^3$ = camera's principal plane



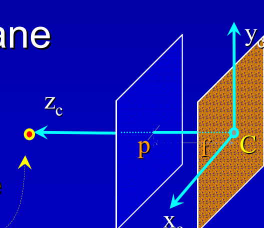
The Pieces of Camera Matrix P

$\mathbf{P} \cdot \mathbf{X} = \mathbf{x}$, or $\left[\; \mathbf{P} \; \right] \begin{bmatrix} x_w \\ y_w \\ z_w \\ t_w \end{bmatrix} = \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix}$ $\quad \mathbf{P} = \begin{bmatrix} \mathbf{P}^{1T} \\ \mathbf{P}^{2T} \\ \mathbf{P}^{3T} \end{bmatrix}$

Rows of P matrix: planes in world space

- row 1 = $\mathbf{P}^1$ = image x-axis plane
- row 2 = $\mathbf{P}^2$ = image y-axis plane
- row 3 = $\mathbf{P}^3$ = camera's principal plane
  - princip. plane $\mathbf{P}^3 = [p_{31}\ p_{32}\ p_{33}\ p_{34}]^T$
  - its normal direction: $[p_{31}\ p_{32}\ p_{33}\ 0\,]^T$
  - Why is it normal? It's the world-space $\mathbf{P}^3$ direction of the z_c axis

Principal plane $\mathbf{P}^3$



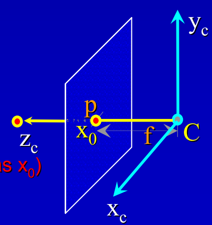
The Pieces of Camera Matrix P

$\mathbf{P \cdot X = x}$, or $\begin{bmatrix} \cdot & \cdot & \cdot & \cdot \\ \cdot & \mathbf{P} & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ t_w \end{bmatrix} = \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix}$ $\quad \mathbf{P} = \left[\begin{array}{c|c} \mathbf{m}^{1T} & \\ \mathbf{m}^{2T} & \mathbf{p}_4 \\ \mathbf{m}^{3T} & \end{array} \right]$

M

- Principal Axis Vector (z_c) in **world** space:
  - Normal of principal plane: = m^3 = $[p_{31}\ p_{32}\ p_{33}]^T$
  - P^3 Scaling → Ambiguous direction!! +/- m^3 ?
  - Solution: use det(M)·m^3 as front of camera
- Principal Point p in **image** space:
  - image of (infinity point on z_c axis = m^3)

  $\mathbf{M} \cdot m^3 = p = \mathbf{x_0}$ (Zisserman book renames p as x_0)



The Pieces of Camera Matrix P

$\mathbf{P \cdot X = x}$, or $\begin{bmatrix} \cdot & \cdot & \cdot & \cdot \\ \cdot & \mathbf{P} & \cdot & \cdot \\ \cdot & \cdot & \cdot & \cdot \end{bmatrix} \begin{bmatrix} x_w \\ y_w \\ z_w \\ t_w \end{bmatrix} = \begin{bmatrix} x_c \\ y_c \\ z_c \end{bmatrix}$ $\quad \mathbf{P} = \left[\begin{array}{c|c} \mathbf{m}^{1T} & \\ \mathbf{m}^{2T} & \mathbf{p}_4 \\ \mathbf{m}^{3T} & \end{array} \right]$

M

Where is camera in world space? at $\tilde{C}$:

- Camera center C is at *camera* origin(x_c,y_c,z_c)=0=C
- Camera P transforms world-space point $\tilde{C}$ to C=0
- But how do we find $\tilde{C}$? It is the Null Space of P :

  $P\,\tilde{C} = C = 0$ (solve for $\tilde{C}$. SVD works, but here's an easier way:)

Camera's Position in the World: $\tilde{C} = \begin{bmatrix} -\mathbf{M}^{-1} \cdot \mathbf{p}_4 \\ 1 \end{bmatrix}$