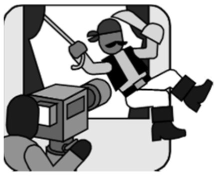
CSC I6716
Spring2011



Topic 1 of Part II
Camera Models

Zhigang Zhu, City College of New York zhu@cs.ccny.cuny.edu

- Closely Related Disciplines
  - Image Processing – images to mages
  - Computer Graphics – models to images
  - Computer Vision – images to models
  - Photogrammetry – obtaining accurate measurements from images
- What is 3-D (three dimensional) Vision?
  - Motivation: making computers see (the 3D world as humans do)
  - Computer Vision: 2D images to 3D structure
  - Applications : robotics / VR /Image-based rendering/ 3D video
- Lectures on 3-D Vision Fundamentals
  - Camera Geometric Models (1 lecture)
  - Camera Calibration (2 lectures)
  - Stereo (2 lectures)
  - Motion (2 lectures)

Lecture Outline

- Geometric Projection of a Camera
  - Pinhole camera model
  - Perspective projection
  - Weak-Perspective Projection
- Camera Parameters
  - Intrinsic Parameters: define mapping from 3D to 2D
  - Extrinsic parameters: define viewpoint and viewing direction
    - Basic Vector and Matrix Operations, Rotation
- Camera Models Revisited
  - Linear Version of the Projection Transformation Equation
    - Perspective Camera Model
    - Weak-Perspective Camera Model
    - Affine Camera Model
    - Camera Model for Planes
- Summary

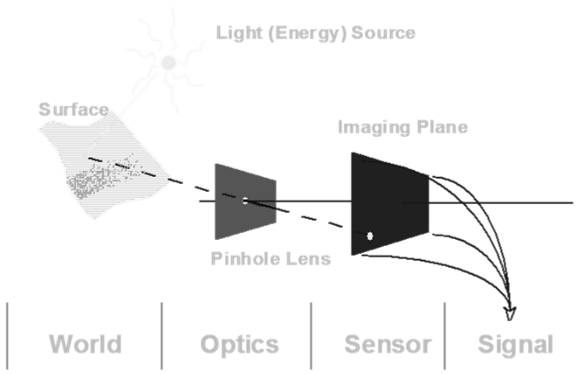
Lecture Assumptions

- Camera **Geometric** Models
  - Knowledge about 2D and 3D geometric transformations
  - Linear algebra (vector, matrix)
  - This lecture is only about geometry
- Goal

Build up relation between 2D images and 3D scenes
-3D Graphics (rendering): from 3D to 2D
-3D Vision (stereo and motion): from 2D to 3D
-Calibration: Determning the parameters for mapping

3D Computer Vision
and Video Computing

Image Formation



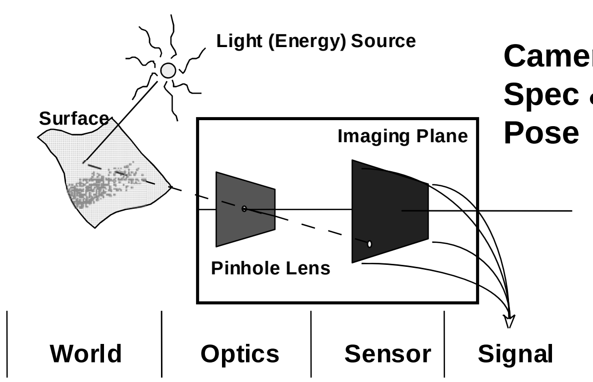
B&W Film	Silver Density
Color Film	Silver density in three color layers
TV Camera	Electrical

3D Computer Vision
and Video Computing

Image Formation

Camera: Spec & Pose

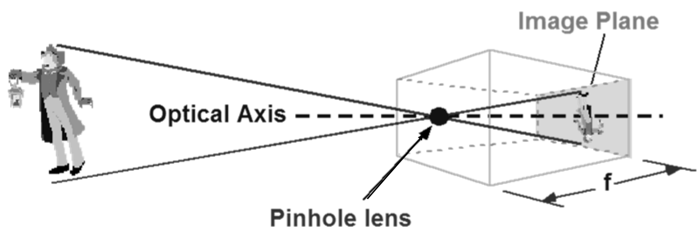
3D Scene



2D Image

3D Computer Vision
and Video Computing

Pinhole Camera Model

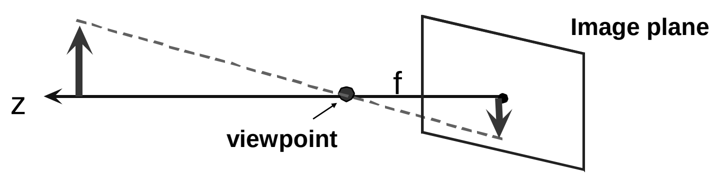


- Pin-hole is the basis for most graphics and vision
  - Derived from physical construction of early cameras
  - Mathematics is very straightforward
- 3D World projected to 2D Image
  - Image inverted, size reduced
  - Image is a 2D plane: No direct depth information
- Perspective projection
  - f called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

3D Computer Vision
and Video Computing

Focal Length, FOV

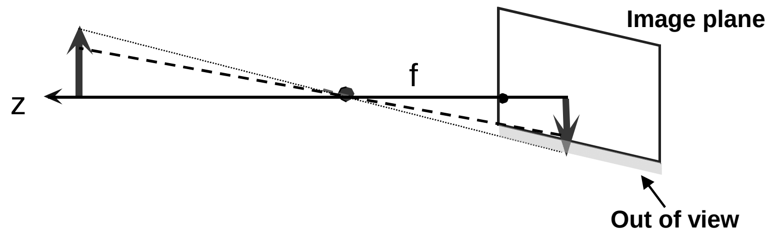
- Consider case with object on the optical axis:



- 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

Focal Length, FOV

- Consider case with object on the optical axis:

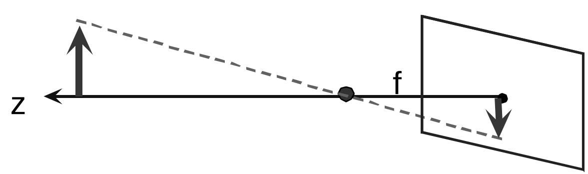


  - Optical axis: the direction of imaging
  - Image plane: a plane perpendicular to the optical axis
  - Center of Projection (pinhole), focal point, viewpoint, , nodal point
  - Focal length: distance from focal point to the image plane
  - FOV : Field of View – viewing angles in horizontal and vertical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

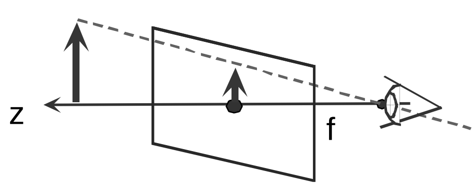
  - Increasing f will enlarge figures, but decrease FOV

Equivalent Geometry

- Consider case with object on the optical axis:



- More convenient with upright image:

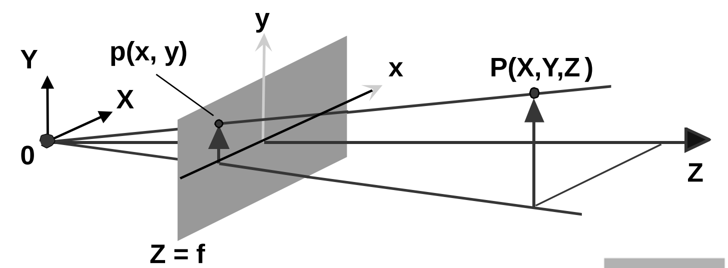


Projection plane z = f

- Equivalent mathematically

Perspective Projection

- Compute the image coordinates of p in terms of the world (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

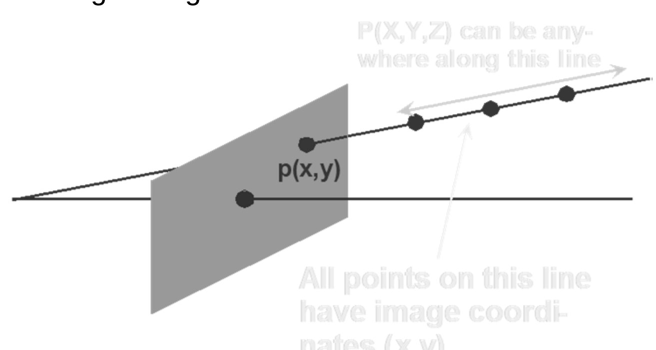
$$x = f\frac{X}{Z}$$

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

3D Computer Vision
and Video Computing

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.

3D Computer Vision
and Video Computing

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

3D Computer Vision
and Video Computing

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

3D Computer Vision
and Video Computing

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

3D Computer Vision
and Video Computing

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 …

- 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…

Yet other pinhole camera images

Rabbit or Man?

Markus Raetz, *Metamorphose II*, 1991-92, cast iron, 15 1/4 x 12 x 12 inches
Fine Art Center University Gallery, Sep 15 – Oct 26

Yet other pinhole camera images

2D projections are not the “same” as the real object as we usually see everyday!

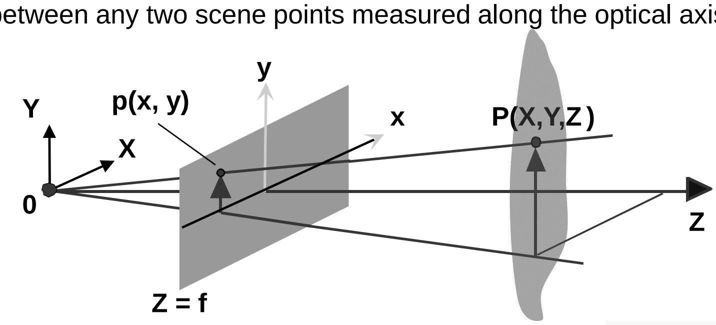
Markus Raetz, *Metamorphose II*, 1991-92, cast iron, 15 1/4 x 12 x 12 inches
Fine Art Center University Gallery, Sep 15 – Oct 26

It’s real!

3D Computer Vision
and Video Computing

Weak Perspective Projection

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



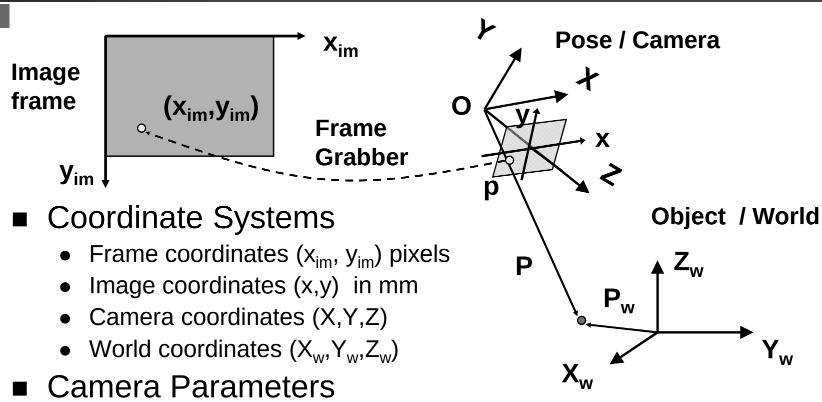
- A sequence of two transformations
  - Orthographic projection : parallel rays
  - Isotropic scaling : $f/\bar{Z}$
- Linear Model
  - Preserve angles and shapes

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

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

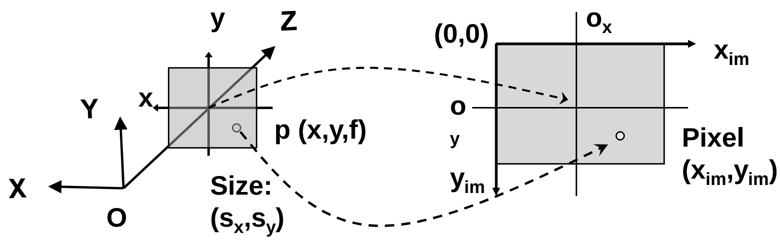
3D Computer Vision
and Video Computing

Camera Parameters



- Coordinate Systems
  - Frame coordinates (x_{im}, y_{im}) pixels
  - Image coordinates (x,y) in mm
  - 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 (I)



- From image to frame
  - Image center
  - Directions of axes
  - Pixel size
- From 3D to 2D
  - Perspective projection
- Intrinsic Parameters
  - (ox ,oy) : image center (in pixels)
  - (sx ,sy) : effective size of the pixel (in mm)
  - f: focal length

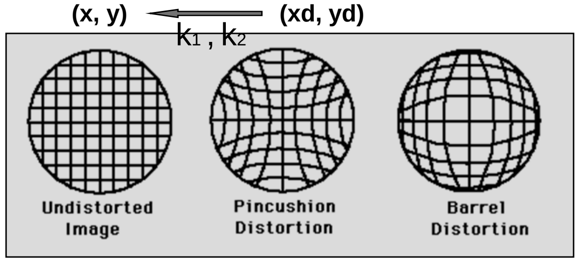
$$x = -(x_{im} - o_x)s_x$$
$$y = -(y_{im} - o_y)s_y$$

$$x = f\frac{X}{Z}$$
$$y = f\frac{Y}{Z}$$

Intrinsic Parameters (II)

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

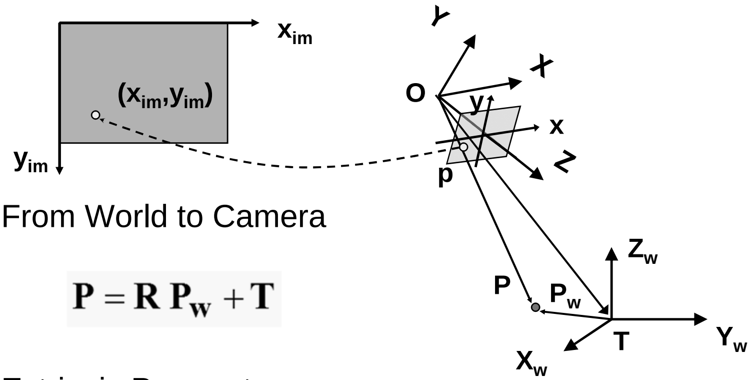
- Lens Distortions



- 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

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

Extrinsic Parameters



- From World to Camera

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

- Extrinsic Parameters
  - A 3-D translation vector, T, describing the relative locations of the origins of the two coordinate systems (what's it?)
  - A 3x3 rotation matrix, R, an orthogonal matrix that brings the corresponding axes of the two systems onto each other

Linear Algebra: Vector and Matrix

T: Transpose

- A point as a 2D/ 3D vector
  - Image point: 2D vector $\mathbf{p} = \begin{pmatrix} x \\ y \end{pmatrix} = (x, y)^T$
  - Scene point: 3D vector $\mathbf{P} = (X, Y, Z)^T$
  - Translation: 3D vector $\mathbf{T} = (T_x, T_y, T_z)^T$
- Vector Operations
  - Addition: $\mathbf{P} = \mathbf{Pw} + \mathbf{T} = (X_w + T_x, Y_w + T_y, Z_w + T_z)^T$
    - Translation of a 3D vector
  - Dot product (a scalar): $c = \mathbf{a} \bullet \mathbf{b} = \mathbf{a}^T \mathbf{b}$
    - a.b = |a||b|cosθ
  - Cross product (a vector) $\mathbf{c} = \mathbf{a} \times \mathbf{b}$
    - Generates a new vector that is orthogonal to both of them

$\underline{\mathbf{a}} \times \underline{\mathbf{b}} = (a_2b_3 - a_3b_2)\underline{\mathbf{i}} + (a_3b_1 - a_1b_3)\underline{\mathbf{j}} + (a_1b_2 - a_2b_1)\underline{\mathbf{k}}$

3D Computer Vision
and Video Computing

Linear Algebra: Vector and Matrix

$$\mathbf{R} = \left(r_{ij}\right)_{3\times 3} = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix} = \begin{bmatrix} \mathbf{R}_1^T \\ \mathbf{R}_2^T \\ \mathbf{R}_3^T \end{bmatrix}$$

- Rotation: 3x3 matrix
  - Orthogonal :

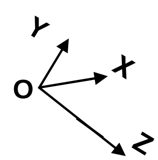
    $\mathbf{R}^{-1} = \mathbf{R}^T$, *i.e.* $\mathbf{R}\mathbf{R}^T = \mathbf{R}^T\mathbf{R} = \mathbf{I}$

    - 9 elements => 3+3 constraints (orthogonal/cross) => 2+2 constraints (unit vectors) => 3 DOF ? (degrees of freedom, orthogonal/dot)
  - How to generate R from three angles? (next few slides)
- Matrix Operations
  - R P_w +T= ? - Points in the World are projected on three new axes (of the camera system) and translated to a new origin

$$\mathbf{P} = \mathbf{R}\mathbf{P_w} + \mathbf{T} = \begin{pmatrix} r_{11}X_w + r_{12}Y_w + r_{13}Z_w + T_x \\ r_{21}X_w + r_{22}Y_w + r_{23}Z_w + T_y \\ r_{31}X_w + r_{32}Y_w + r_{33}Z_w + T_z \end{pmatrix} = \begin{bmatrix} \mathbf{R}_1^T\mathbf{P_w} + T_x \\ \mathbf{R}_2^T\mathbf{P_w} + T_y \\ \mathbf{R}_3^T\mathbf{P_w} + T_z \end{bmatrix}$$

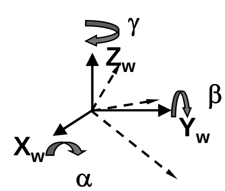
3D Computer Vision
and Video Computing

Rotation: from Angles to Matrix



- Rotation around the Axes
  - Result of three consecutive rotations around the coordinate axes

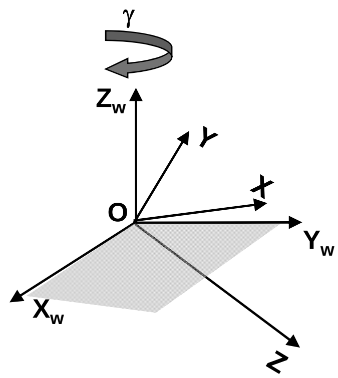
$$\mathbf{R} = \mathbf{R}_\alpha \mathbf{R}_\beta \mathbf{R}_\gamma$$



- Notes:
  - Only three rotations
  - Every time around one axis
  - Bring corresponding axes to each other
    - Xw = X, Yw = Y, Zw = Z
  - First step (e.g.) Bring Xw to X

Rotation: from Angles to Matrix

$$\mathbf{R}_{\gamma} = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

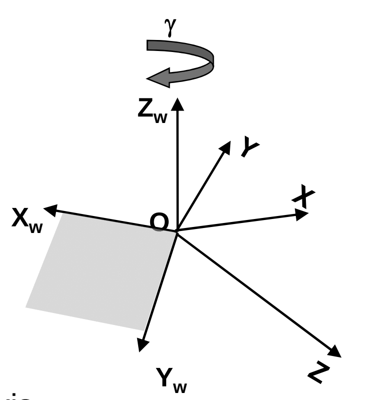


- Rotation γ around the Z_w Axis
  - Rotate in X_wOY_w plane
  - Goal: Bring X_w to X
  - But X is not in XwOYw

  - $Y_w \perp X \Rightarrow X$ in X_wOZ_w ($\Leftarrow Y_w \perp X_wOZ_w$) $\Rightarrow Y_w$ in YOZ ($\Leftarrow X \perp$ YOZ)
- Next time rotation around Y_w

Rotation: from Angles to Matrix

$$\mathbf{R}_{\gamma} = \begin{bmatrix} \cos\gamma & -\sin\gamma & 0 \\ \sin\gamma & \cos\gamma & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

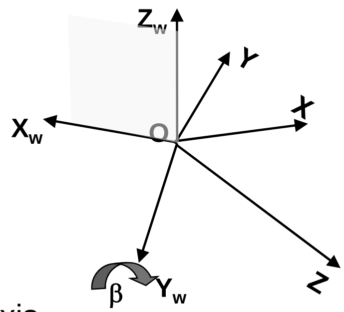


- Rotation γ around the Z_w Axis
  - Rotate in X_wOY_w plane so that
  - $Y_w \perp X \Rightarrow X$ in X_wOZ_w ($\Leftarrow Y_w \perp X_wOZ_w$) $\Rightarrow Y_w$ in YOZ ($\Leftarrow X \perp$ YOZ)
- Z_w does not change

3D Computer Vision
and Video Computing

Rotation: from Angles to Matrix

$$\mathbf{R}_\gamma = \begin{bmatrix} \cos\beta & 0 & -\sin\beta \\ 0 & 1 & 0 \\ \sin\beta & 0 & \cos\beta \end{bmatrix}$$

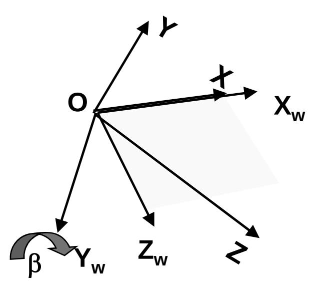


- Rotation β around the Y_w Axis
  - Rotate in X_wOZ_w plane so that
  - $X_w = X \Rightarrow Z_w$ in YOZ (& Y_w in YOZ)
- Y_w does not change

3D Computer Vision
and Video Computing

Rotation: from Angles to Matrix

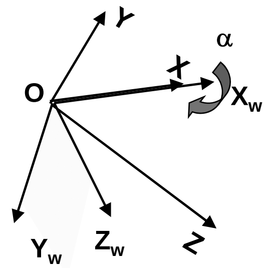
$$\mathbf{R}_\beta = \begin{bmatrix} \cos\beta & 0 & -\sin\beta \\ 0 & 1 & 0 \\ \sin\beta & 0 & \cos\beta \end{bmatrix}$$



- Rotation β around the Y_w Axis
  - Rotate in X_wOZ_w plane so that
  - $X_w = X \Rightarrow Z_w$ in YOZ (& Y_w in YOZ)
- Y_w does not change

Rotation: from Angles to Matrix

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

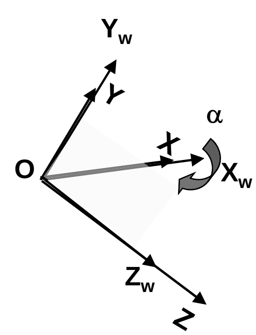


- Rotation α around the X_w(X) Axis
  - Rotate in Y_wOZ_w plane so that
  - $Y_w = Y$, $Z_w = Z$ (& $X_w = X$)
- X_w does not change

3D Computer Vision
and Video Computing

Rotation: from Angles to Matrix

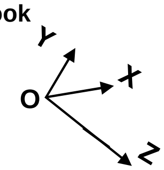
$$\mathbf{R}_\alpha = \begin{bmatrix} 1 & 0 & 0 \\ 0 & \cos\alpha & -\sin\alpha \\ 0 & \sin\alpha & \cos\alpha \end{bmatrix}$$



- Rotation α around the X_w(X) Axis
  - Rotate in Y_wOZ_w plane so that
  - $Y_w = Y$, $Z_w = Z$ (& $X_w = X$)
- X_w does not change

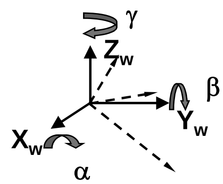
Rotation: from Angles to Matrix

Appendix A.9 of the textbook



- Rotation around the Axes
  - Result of three consecutive rotations around the coordinate axes

$$\mathbf{R} = \mathbf{R}_\alpha \mathbf{R}_\beta \mathbf{R}_\gamma$$



- Notes:
  - Rotation directions
  - The order of multiplications matters: γ,β,α
  - Same R, 6 different sets of α,β,γ
  - **R Non-linear function of** α,β,γ
  - **R is orthogonal**
  - **It's easy to compute angles from R**

$$\mathbf{R} = \begin{bmatrix} \cos\beta\cos\gamma & -\cos\beta\sin\gamma & -\sin\beta \\ -\sin\alpha\sin\beta\cos\gamma + \cos\alpha\sin\gamma & \sin\alpha\sin\beta\sin\gamma + \cos\alpha\cos\gamma & -\sin\alpha\cos\beta \\ \cos\alpha\sin\beta\cos\gamma + \sin\alpha\sin\gamma & -\cos\alpha\sin\beta\sin\gamma + \sin\alpha\cos\gamma & \cos\alpha\cos\gamma \end{bmatrix}$$

Rotation- Axis and Angle

Appendix A.9 of the textbook

- According to Euler's Theorem, any 3D rotation can be described by a rotating angle, θ, around an axis defined by an unit vector $\mathbf{n} = [n_1, n_2, n_3]^T$.
- Three degrees of freedom – why?

$$\mathbf{R} = I\cos\theta + \begin{bmatrix} n_1^2 & n_1 n_2 & n_1 n_3 \\ n_2 n_1 & n_2^2 & n_2 n_3 \\ n_3 n_1 & n_3 n_2 & n_3^2 \end{bmatrix}(1-\cos\theta) + \begin{bmatrix} 0 & -n_3 & n_2 \\ n_3 & 0 & -n_1 \\ -n_2 & n_1 & 0 \end{bmatrix}\sin\theta$$

Linear Version of Perspective Projection

- World to Camera
  - Camera: $P = (X,Y,Z)^T$
  - World: $P_w = (X_w,Y_w,Z_w)^T$
  - Transform: R, T
- Camera to Image
  - Camera: $P = (X,Y,Z)^T$
  - Image: $p = (x,y)^T$
  - Not linear equations
- Image to Frame
  - Neglecting distortion
  - Frame $(x_{im}, y_{im})^T$
- World to Frame
  - $(X_w,Y_w,Z_w)^T \rightarrow (x_{im}, y_{im})^T$
  - Effective focal lengths
    - $f_x = f/s_x$, $f_y=f/s_y$
    - Three are not independent

$$\mathbf{P} = \mathbf{R}\mathbf{P_w} + \mathbf{T} = \begin{pmatrix} r_{11}X_w + r_{12}Y_w + r_{13}Z_w + T_x \\ r_{21}X_w + r_{22}Y_w + r_{23}Z_w + T_y \\ r_{31}X_w + r_{32}Y_w + r_{33}Z_w + T_z \end{pmatrix} = \begin{bmatrix} \mathbf{R}_1^T\mathbf{P_w} + T_x \\ \mathbf{R}_2^T\mathbf{P_w} + T_y \\ \mathbf{R}_3^T\mathbf{P_w} + T_z \end{bmatrix}$$

$$(x, y) = (f\frac{X}{Z},\ f\frac{Y}{Z})$$

$$x = -(x_{im} - o_x)s_x$$
$$y = -(y_{im} - o_y)s_y$$

$$x_{im} - o_x = -f_x \frac{r_{11}X_w + r_{12}Y_w + r_{13}Z_w + T_x}{r_{31}X_w + r_{32}Y_w + r_{33}Z_w + T_z}$$

$$y_{im} - o_y = -f_y \frac{r_{21}X_w + r_{22}Y_w + r_{23}Z_w + T_y}{r_{31}X_w + r_{32}Y_w + r_{33}Z_w + T_z}$$

Linear Matrix Equation of perspective projection

- Projective Space
  - Add fourth coordinate
    - $P_w = (X_w,Y_w,Z_w, 1)^T$
  - Define $(x_1,x_2,x_3)^T$ such that
    - $x_1/x_3 = x_{im}$, $x_2/x_3 = y_{im}$
- 3x4 Matrix M_{ext}
  - Only extrinsic parameters
  - World to camera
- 3x3 Matrix M_{int}
  - Only intrinsic parameters
  - Camera to frame

$$\begin{pmatrix} x_{im} \\ y_{im} \end{pmatrix} = \begin{pmatrix} x_1/x_3 \\ x_2/x_3 \end{pmatrix}$$

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \mathbf{M_{int}}\mathbf{M_{ext}} \begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix}$$

$$\mathbf{M}_{ext} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & T_x \\ r_{21} & r_{22} & r_{23} & T_y \\ r_{31} & r_{32} & r_{33} & T_z \end{bmatrix} = \begin{bmatrix} \mathbf{R}_1^T & T_x \\ \mathbf{R}_2^T & T_y \\ \mathbf{R}_3^T & T_z \end{bmatrix}$$

$$\mathbf{M}_{\mathrm{int}} = \begin{bmatrix} -f_x & 0 & o_x \\ 0 & -f_y & o_y \\ 0 & 0 & 1 \end{bmatrix}$$

- Simple Matrix Product! Projective Matrix $M = M_{int}M_{ext}$
  - $(X_w,Y_w,Z_w)^T \rightarrow (x_{im}, y_{im})^T$
  - Linear Transform from projective space to projective plane
  - M defined up to a scale factor – 11 independent entries

Three Camera Models

- Perspective Camera Model
  - Making some assumptions
    - Known center: Ox = Oy = 0
    - Square pixel: Sx = Sy = 1
  - 11 independent entries <-> 7 parameters

$$\mathbf{M} = \begin{bmatrix} -fr_{11} & -fr_{12} & -fr_{13} & -fT_x \\ -fr_{21} & -fr_{22} & -fr_{23} & -fT_y \\ r_{31} & r_{32} & r_{33} & T_z \end{bmatrix}$$

- Weak-Perspective Camera Model
  - Average Distance $\overline{Z}$ >> Range δZ
  - Define centroid vector $\overline{P}_w$

$$Z = \overline{\mathbf{Z}} = \mathbf{R}_3^{\mathbf{T}} \overline{\mathbf{P}}_w + T_z$$

  - 8 independent entries

$$\mathbf{M}_{wp} = \begin{bmatrix} -fr_{11} & -fr_{12} & -fr_{13} & -fT_x \\ -fr_{21} & -fr_{22} & -fr_{23} & -fT_y \\ 0 & 0 & 0 & \mathbf{R}_3^{\mathbf{T}} \overline{\mathbf{P}}_w + T_z \end{bmatrix}$$

- Affine Camera Model
  - Mathematical Generalization of Weak-Pers
  - Doesn't correspond to physical camera
  - But simple equation and appealing geometry
    - Doesn't preserve angle BUT parallelism
  - 8 independent entries

$$\mathbf{M}_{af} = \begin{bmatrix} a_{11} & a_{12} & a_{13} & b_1 \\ a_{21} & a_{22} & a_{23} & b_2 \\ 0 & 0 & 0 & b_3 \end{bmatrix}$$

Camera Models for a Plane

- Planes are very common in the Man-Made World

$$n_x X_w + n_y Y_w + n_z Z_w = d \iff \mathbf{n}^{\mathbf{T}} \mathbf{P_w} = d$$

  - One more constraint for all points: Zw is a function of Xw and Yw
- Special case: Ground Plane
  - Zw=0
  - Pw =(Xw, Yw,0, 1)^T
  - 3D point -> 2D point

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{bmatrix} -fr_{11} & -fr_{12} & -fr_{13} & -fT_x \\ -fr_{21} & -fr_{22} & -fr_{23} & -fT_y \\ r_{31} & r_{32} & r_{33} & T_z \end{bmatrix} \begin{pmatrix} X_w \\ Y_w \\ Z_w = 0 \\ 1 \end{pmatrix}$$

- Projective Model of a Plane
  - 8 independent entries
- General Form ?
  - 8 independent entries

Camera Models for a Plane

- A Plane in the World

$$n_x X_w + n_y Y_w + n_z Z_w = d \iff \mathbf{n}^{\mathbf{T}}\mathbf{P_w} = d$$

  - One more constraint for all points: Zw is a function of Xw and Yw
- Special case: Ground Plane
  - Zw=0
  - Pw =(Xw, Yw,0, 1)^T
  - 3D point -> 2D point

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{bmatrix} -fr_{11} & -fr_{12} & \boxed{-fr_{13}} & -fT_x \\ -fr_{21} & -fr_{22} & -fr_{23} & -fT_y \\ r_{31} & r_{32} & r_{33} & T_z \end{bmatrix} \begin{pmatrix} X_w \\ Y_w \\ Z_w = 0 \\ 1 \end{pmatrix}$$

- Projective Model of Zw=0
  - 8 independent entries

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{bmatrix} -fr_{11} & -fr_{12} & -fT_x \\ -fr_{21} & -fr_{22} & -fT_{23} \\ r_{31} & r_{32} & T_z \end{bmatrix} \begin{pmatrix} X_w \\ Y_w \\ 1 \end{pmatrix}$$

- General Form ?
  - 8 independent entries

Camera Models for a Plane

- A Plane in the World

$$n_x X_w + n_y Y_w + n_z Z_w = d \iff \mathbf{n}^{\mathbf{T}}\mathbf{P_w} = d$$

  - One more constraint for all points: Zw is a function of Xw and Yw
- Special case: Ground Plane
  - Zw=0
  - Pw =(Xw, Yw,0, 1)^T
  - 3D point -> 2D point

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{bmatrix} -fr_{11} & -fr_{12} & -fr_{13} & -fT_x \\ -fr_{21} & -fr_{22} & -fr_{23} & -fT_y \\ r_{31} & r_{32} & r_{33} & T_z \end{bmatrix} \begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix}$$

- Projective Model of Zw=0
  - 8 independent entries
- General Form ?
  - nz = 1

$$Z_w = d - n_x X_w - n_y Y_w$$

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \begin{bmatrix} -f(r_{11} - n_x r_{13}) & -f(r_{12} - n_y r_{13}) & -f(dr_{13} + T_x) \\ -f(r_{21} - n_x r_{23}) & -f(r_{22} - n_y r_{23}) & -f(dr_{23} + T_y) \\ (r_{31} - n_x r_{33}) & (r_{32} - n_y r_{33}) & (dr_{33} + T_z) \end{bmatrix} \begin{pmatrix} X_w \\ Y_w \\ 1 \end{pmatrix}$$

  - 8 independent entries

- **2D (x_{im},y_{im}) -> 3D (X_w, Y_w, Z_w) ?**

Applications and Issues

- Graphics /Rendering
  - From 3D world to 2D image
    - Changing viewpoints and directions
    - Changing focal length
  - Fast rendering algorithms
- Vision / Reconstruction
  - From 2D image to 3D model
    - Inverse problem
    - Much harder / unsolved
  - Robust algorithms for matching and parameter estimation
  - Need to estimate camera parameters first
- Calibration
  - Find intrinsic & extrinsic parameters
  - Given image-world point pairs
  - Probably a partially solved problem ?
  - 11 independent entries
    - <-> 10 parameters: f_x, f_y, o_x, o_y, α,β,γ, Tx,Ty,Tz

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \mathbf{M}_{\mathbf{int}}\mathbf{M}_{\mathbf{ext}} \begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix}$$

$$\begin{pmatrix} x_{im} \\ y_{im} \end{pmatrix} = \begin{pmatrix} x_1/x_3 \\ x_2/x_3 \end{pmatrix}$$

$$\mathbf{M}_{\text{int}} = \begin{bmatrix} -f_x & 0 & o_x \\ 0 & -f_y & o_y \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{M}_{ext} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & T_x \\ r_{21} & r_{22} & r_{23} & T_y \\ r_{31} & r_{32} & r_{33} & T_z \end{bmatrix}$$

Camera Model Summary

- Geometric Projection of a Camera
  - Pinhole camera model
  - Perspective projection
  - Weak-Perspective Projection
- Camera Parameters (10 or 11)
  - Intrinsic Parameters: f, o_x,o_y, s_x,s_y,k_1: 4 or 5 independent parameters
  - Extrinsic parameters: R, T – 6 DOF (degrees of freedom)
- Linear Equations of Camera Models (without distortion)
  - General Projection Transformation Equation : 11 parameters
  - Perspective Camera Model: 11 parameters
  - Weak-Perspective Camera Model: 8 parameters
  - Affine Camera Model: generalization of weak-perspective: 8
  - Projective transformation of planes: 8 parameters

Next

- Determining the value of the extrinsic and intrinsic parameters of a camera

Calibration
(Ch. 6)

$$\begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = \mathbf{M}_{\mathbf{int}} \mathbf{M}_{\mathbf{ext}} \begin{pmatrix} X_w \\ Y_w \\ Z_w \\ 1 \end{pmatrix} \Rightarrow$$

$$\mathbf{M}_{\text{int}} = \begin{bmatrix} -f_x & 0 & o_x \\ 0 & -f_y & o_y \\ 0 & 0 & 1 \end{bmatrix}$$

$$\mathbf{M}_{ext} = \begin{bmatrix} r_{11} & r_{12} & r_{13} & T_x \\ r_{21} & r_{22} & r_{23} & T_y \\ r_{31} & r_{32} & r_{33} & T_z \end{bmatrix}$$