Camera Calibration

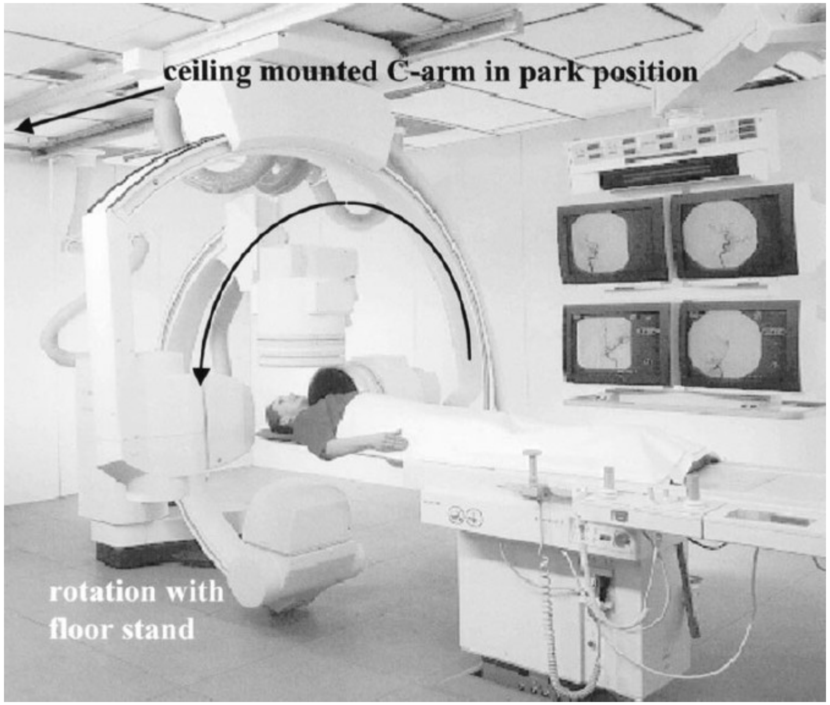
quirin.n.meyer@stud.informatik.uni-erlangen.de

Outline

- Motivation
- Camera
  - Projective Mapping
  - Homogeneous coordinates
- Calibration
- Application to C-Arm CT

Motivation

- C-Arm CT
- Detector and X-ray source rotate around patient
  - Up to 220 Degrees
- Application: Intervention (e.g. During operations)

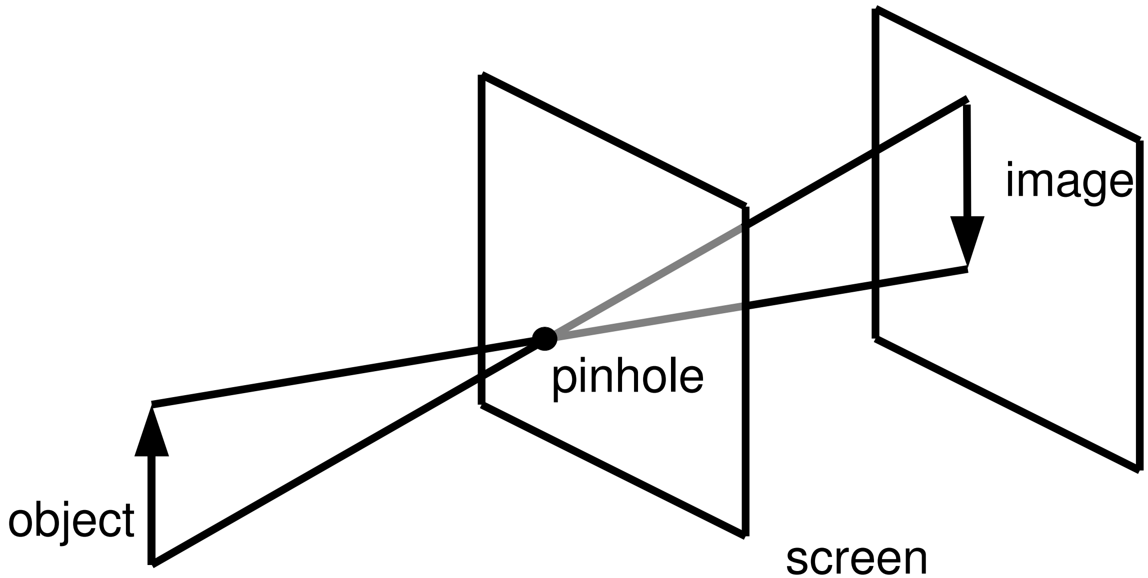


Motivation

- Difficulties when applying C-ARM CT
  - Detector and source trajectory not an ideal circle arc (ideal Feldkamp geometry vs. Irregular Feldkamp geometry)
  - Perturbed by mechanical quantities
    - Inertia
    - Gravity
- Deviations are not negligible and must be corrected
- Applying regular Feldkamp algorithm for reconstruction exhibits artifacts

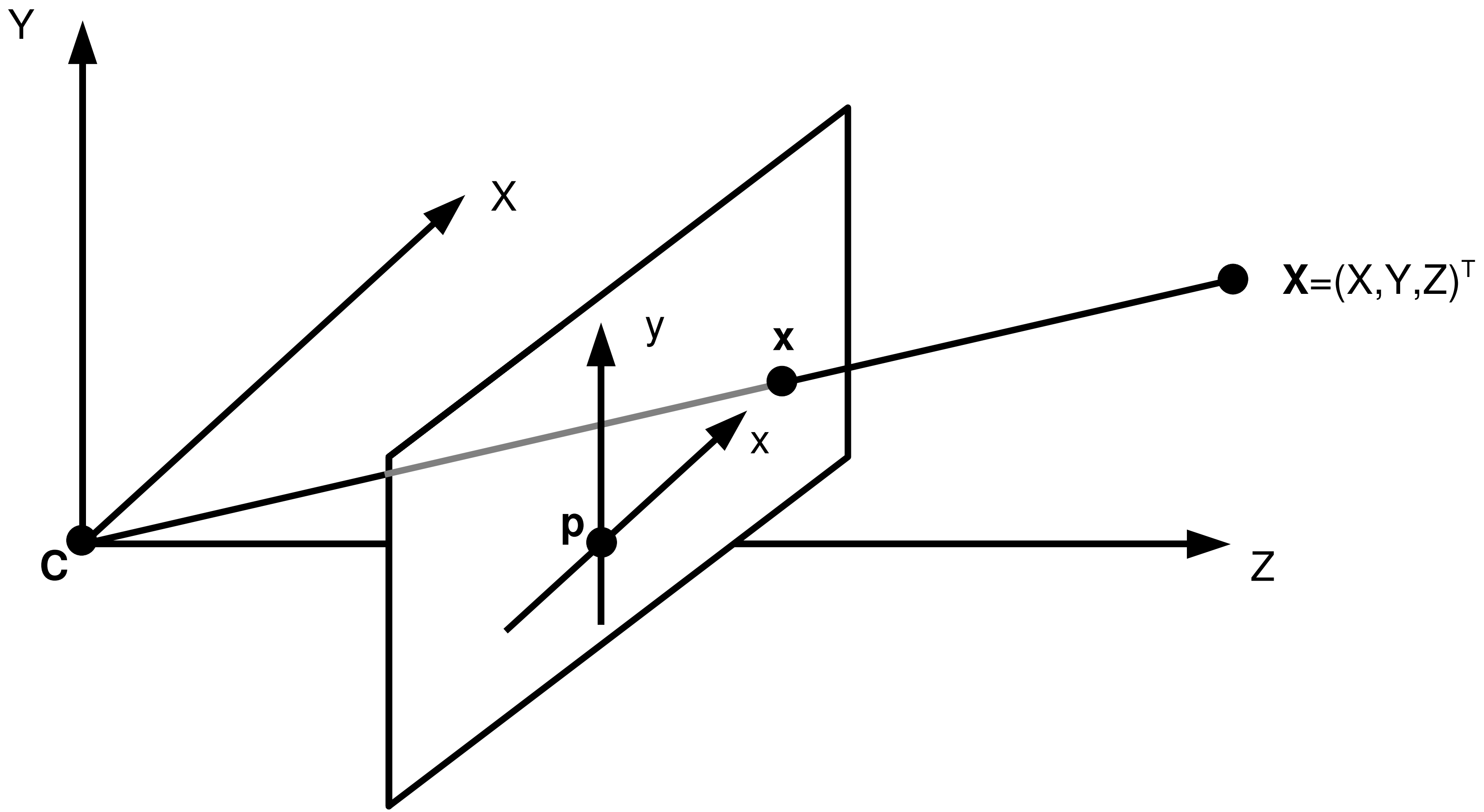
Camera

- Pinhole camera model
  - Reflected light from the object shines through the pinhole
  - Gets projected onto the image screen
  - Note that directions get flipped



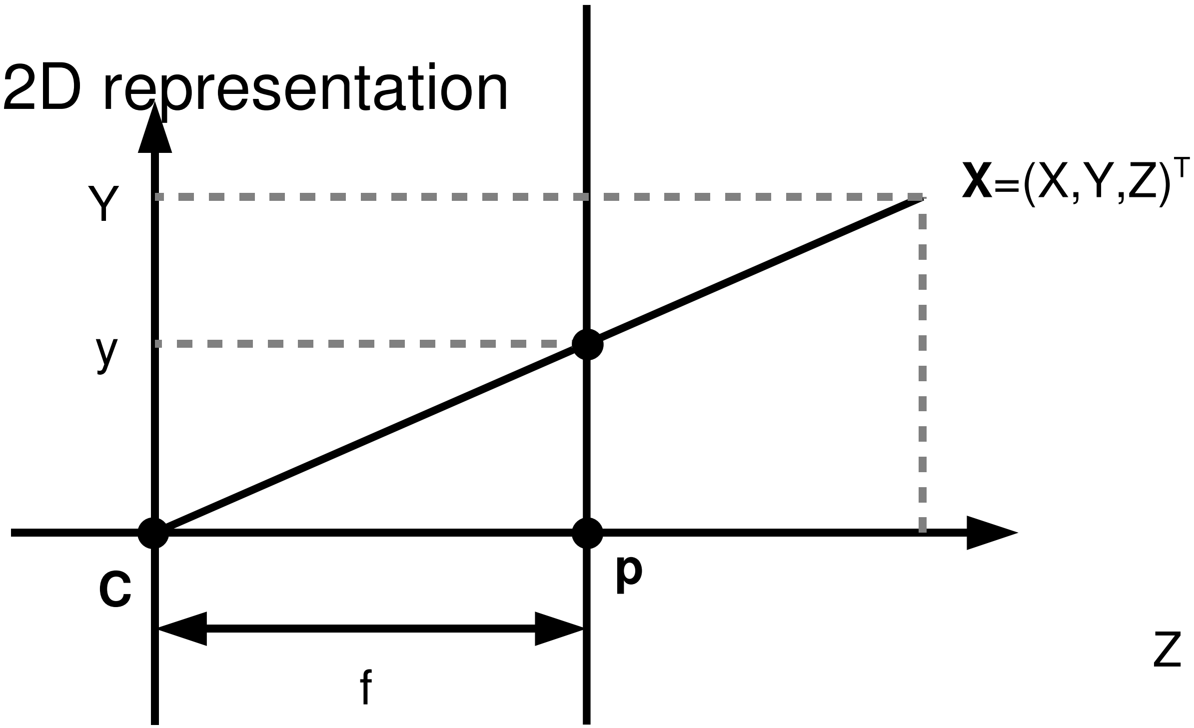
Camera

- Geometric pinhole model



- How to calculate the coordinates of $\mathbf{x} = (x,y)^T$

- **Projective Mapping**
  - Consider 2D representation



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

- Analogously:

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

Camera

- Question: What to do with the nonlinearity?
- Answer: Use **homogeneous coordinates**
- 3D projective space
  - $\mathbb{P}^3 = \mathbb{R}^4 - (0,0,0,0)^T$
  - $\mathbf{X} \in \mathbb{P}^4 = (X_1, X_2, X_3, X_4) \quad \text{with} \quad X_4 \neq 0$
  - Mapping

$$\mathbb{P}^3 \rightarrow \mathbb{R}^3 : (X_1, X_2, X_3, X_4)^T \rightarrow (\frac{X_1}{X_4}, \frac{X_2}{X_4}, \frac{X_3}{X_4})^T$$
$$\mathbb{R}^3 \rightarrow \mathbb{P}^3 : (X_1, X_2, X_3)^T \rightarrow (X_1, X_2, X_3, 1)^T$$

- Equivalence of two homogeneous points:

$$\mathbf{a} \cong \mathbf{b} \iff \mathbf{a} = s\mathbf{b} \quad \text{with} \quad s \in \mathbb{R} - \{0\}$$

Camera

- Remember:

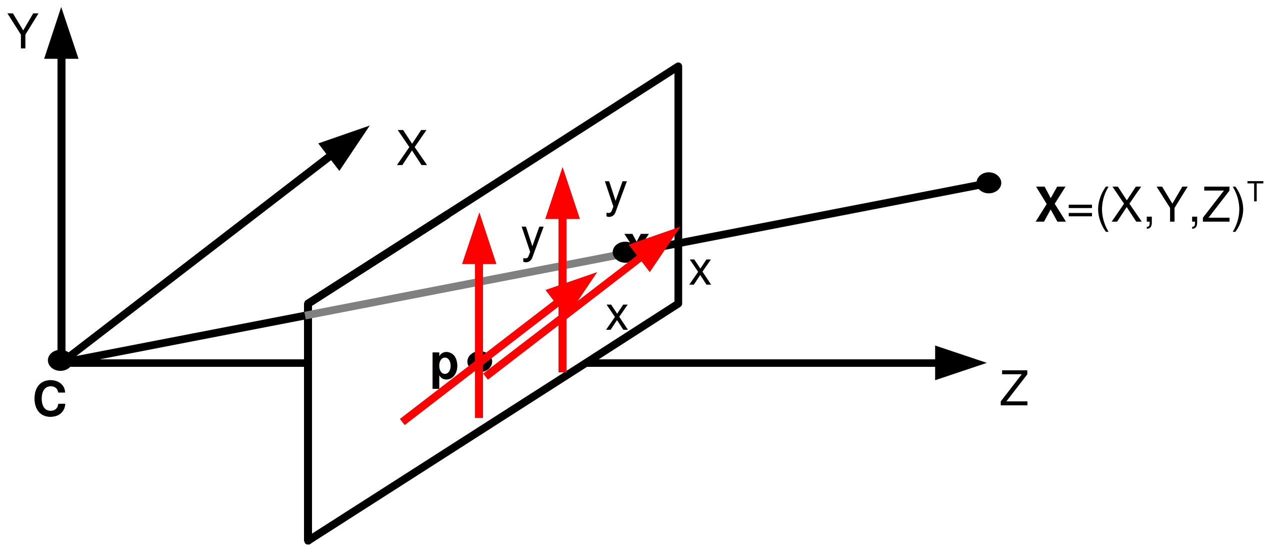
$$\mathbf{x} = \begin{pmatrix} x \\ y \end{pmatrix} = P(\mathbf{X}) = \begin{pmatrix} fX/Z \\ fY/Z \end{pmatrix}$$

- **x** in homogeneous coordinates: $(fX, fY, Z)^T$
- Find suitable linear mapping

$$\mathbf{P}' = \begin{pmatrix} f & 0 & 0 & 0 \\ 0 & f & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \in \mathbb{R}^{3\times 4}$$

Camera

- Refinement:
  - Move the origin of the image coordinate system away from the principle point **p**



$$(X, Y, Z)^T \rightarrow (X/Z + p_x, Y/Z + p_y)^T$$

$$\mathbf{P}' = \begin{pmatrix} f & 0 & p_x & 0 \\ 0 & f & p_y & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \in \mathbb{R}^{3\times 4}$$

Camera

- Refinement:
  - Number of pixels in unit distance: m_x, m_y

$$\mathbf{P}' = \begin{pmatrix} fm_x & 0 & p_x & 0 \\ 0 & fm_y & p_y & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \in \mathbb{R}^{3\times 4}$$

  - Pixel axises are not perpendicular: skew factor *s* required:

$$\mathbf{P}' = \begin{pmatrix} fm_x & s & p_x & 0 \\ 0 & fm_y & p_y & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \in \mathbb{R}^{3\times 4}$$

Camera

- Calibration matrix **K**

$$\mathbf{P}' = \begin{pmatrix} fm_x & s & p_x & 0 \\ 0 & fm_y & p_y & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \in \mathbb{R}^{3\times 4}$$

- Encodes **intrinsic parameters** (5 DOF)
  - Optical
  - Geometric
  - Invariant of camera movement and position

$$\mathbf{K} = \begin{pmatrix} \alpha_x & s & p_x \\ 0 & \alpha_y & p_y \\ 0 & 0 & 1 \end{pmatrix} \in \mathbb{R}^{3\times 3}$$

Camera

- $\mathbf{P_s}$: Projection-model matrix:

$$\mathbf{P_s} = \begin{pmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \in \mathbb{R}^{3\times 4}$$

- Decompostion of Matrix **P'** into $\mathbf{P_s}$ and **K**:

$$\mathbf{P}' = \mathbf{K}\mathbf{P_s}$$

- Until now: Camera is at fixed location
- Goal: Camera can be arbitrary placed in the World Coordinate System

Camera

- Ridged body movement of Camera
  - Rotation/Orientation: Matrix $\mathbf{R} \in \mathbb{R}^{3\times 3}$
  - Translation: Vector $\mathbf{t} \in \mathbb{R}^{3}$
- Moving of a vertex by a rigid body mapping:

$$\mathbf{v}' = \mathbf{R}\mathbf{v} + \mathbf{t}$$

- Note that **R** must be orthogonal
- Examples of **R**
  - Rotation Matrix around principal axis

$$\mathbf{R_z} = \begin{pmatrix} \cos(\phi) & -\sin(\phi) & 0 \\ \sin(\phi) & \cos(\phi) & 0 \\ 0 & 0 & 1 \end{pmatrix}$$

Camera

- Creating appropriate camera matrix
- Assume camera is located at **c'** in world coordinates with an orientation defined by **R**
- **x** 3D point in world coordinates
- $\mathbf{x}_{cam}$ same point in camera coordinate system:

$$\mathbf{x}_{cam} = \mathbf{R}(\mathbf{x} - \mathbf{c}') = \mathbf{R} \cdot \mathbf{x} - \mathbf{R} \cdot \mathbf{c}'$$

- Therefore set:

$$\mathbf{t} := -\mathbf{R} \cdot \mathbf{c}'$$

Camera

- Make use of homogeneous coordinates to get rid of the addition:

$$\mathbf{D} = \begin{pmatrix} & \mathbf{R} & & \mathbf{t} \\ 0 & 0 & 0 & 1 \end{pmatrix} \in \mathbb{R}^{4\times 4}$$

- Now a single vertex can be transformed by one matrix multiplication
  - $\mathbf{v}' = \mathbf{D}\mathbf{v} \quad \text{with} \quad \mathbf{v}', \mathbf{v} \in \mathbb{P}^4$
  - Note that vertex must be extended to homogeneous coordinates

Camera

- Parameter in the matrix **D**: **extrinsic parameters**
- Degrees of freedom
  - Rotation
    - Axis, i.e. Direction, 2 DOF
    - Angle 1 DOF
  - Translation
    - Vector: 3 DOF
  - --> totally 6 Degrees of freedom for rigid body motion

Camera

- Transforming a point from world coordinate system into image coordinate system:

$$\begin{pmatrix} x \\ y \\ u \end{pmatrix} = \mathbf{P} \cdot \begin{pmatrix} X \\ Y \\ Z \\ W \end{pmatrix} = \mathbf{K} \cdot \mathbf{P}_s \cdot \mathbf{D} \cdot \begin{pmatrix} X \\ Y \\ Z \\ W \end{pmatrix}$$

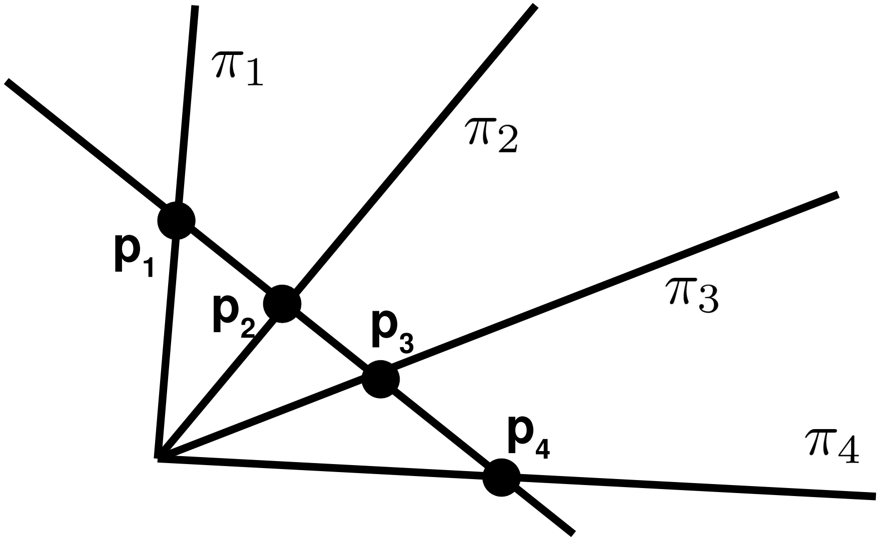
$$= \begin{pmatrix} fm_x & s & p_x & 0 \\ 0 & fm_y & p_y & 0 \\ 0 & 0 & 1 & 0 \end{pmatrix} \cdot \begin{pmatrix} & \mathbf{R} & & \mathbf{t} \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} X \\ Y \\ Z \\ W \end{pmatrix}$$

- Getting image coordinates: perform **perspective divide** (i.e. convert homogeneous coordinates into euclidean coordinates)

$$x_{img} = \frac{x}{u} \qquad y_{img} = \frac{y}{u}$$

Camera

- Properties of projection matrices
  - Matrix is unique up to constant value
  - Lines map to pines, plans to planes
  - Line segments do **not** map to line segments
  - Does not preserve parallelism
  - Preserves cross ratio
    - 4 planes: $\{\pi_1, \pi_2; \pi_3, \pi_4\} = \frac{(\pi_1 - \pi_3)(\pi_2 - \pi_4)}{(\pi_1 - \pi_4)(\pi_2 - \pi_3)}$



Camera

- Summary
  - Pinhole camera: Projection Matrix
    - Intrinsic parameters (Geometry and optical properties)
    - Extrinsic Parameters (Location and Orientation)
  - Use of homogeneous coordinates
  - Projection matrix plus perspective Divide transforms world coordinates into image coordinates

Calibration

- Given: intrinsic and extrinsic parameters: Create projection matrix
- Given: Projection matrix – How to retrieve parameters
  - RQ-Decomposition
    - $\mathbf{P} = (\mathbf{M} \quad \mathbf{d}) \quad \text{with} \quad \mathbf{P} \in \mathbb{R}^{3\times 4}, \mathbf{M} \in \mathbb{R}^{3\times 3}, \mathbf{d} \in \mathbb{R}^3$
    - $\mathbf{M} = \mathbf{RQ}$
    - **Q** orthogonal matrix, which is **R** (orientation)
    - **R** upper right diagonal matrix, which is **K**
    - Algorithmically: **Givens** rotation
  - Location **c** of camera
    - Solve: $\mathbf{Pc} = \mathbf{0}$

Calibration

- Other quantities retrieved through **P**
  - Vanishing points
    - Column vectors of $\mathbf{P}$: $\mathbf{p}_1, \quad \mathbf{p}_2, \quad \mathbf{p}_3$
  - Principle Point
    - $\mathbf{x}_0 = \mathbf{M}\mathbf{m}^3$
  - Principle Ray
    - $\mathbf{v} = det(M)\mathbf{m}^3$
  - Principle Plane
    - $\mathbf{P}^3$

Calibration

- “Process of estimating the intrinsic and extrinsic parameters of a camera” [0]
- Here: Estimating projection matrix **P**
- Linear approach
- Remember:

$$\begin{pmatrix} x \\ y \\ u \end{pmatrix} = \begin{pmatrix} p_{11} & p_{12} & p_{13} & p_{14} \\ p_{21} & p_{22} & p_{23} & p_{24} \\ p_{31} & p_{32} & p_{33} & p_{34} \end{pmatrix} \cdot \begin{pmatrix} X \\ Y \\ Z \\ W \end{pmatrix}$$

- Perspective divide:

$$x_{img} = \frac{x}{u} \qquad y_{img} = \frac{y}{u}$$

Calibration

- Multiply out

$$x = \mathbf{p}^1\mathbf{X}$$
$$y = \mathbf{p}^2\mathbf{X}$$
$$w = \mathbf{p}^3\mathbf{X}$$

- Perspective divide:

$$x_{img} = \frac{\mathbf{p}^1\mathbf{X}}{\mathbf{p}^3\mathbf{X}}$$

$$y_{img} = \frac{\mathbf{p}^2\mathbf{X}}{\mathbf{p}^3\mathbf{X}}$$

- Multiply denominator:

$$\mathbf{p}^3\mathbf{X}x_{img} = \mathbf{p}^1\mathbf{X}$$

$$\mathbf{p}^3\mathbf{X}y_{img} = \mathbf{p}^2\mathbf{X}$$

Calibration

- One correspondence

$$\mathbf{x}_{img} \leftrightarrow \mathbf{X}$$

$$\mathbf{p}^3 \cdot \mathbf{X} \cdot x_{img} = \mathbf{p}^1 \cdot \mathbf{X}$$
$$\mathbf{p}^3 \cdot \mathbf{X} \cdot y_{img} = \mathbf{p}^2 \cdot \mathbf{X}$$

$$\mathbf{p}^1 \cdot \mathbf{X} - \mathbf{p}^3 \cdot \mathbf{X} \cdot x_{img} = 0$$
$$\mathbf{p}^2 \cdot \mathbf{X} - \mathbf{p}^3 \cdot \mathbf{X} \cdot y_{img} = 0$$

$$\begin{pmatrix} \mathbf{X} & \mathbf{0} & -\mathbf{X} \cdot x_{img} \\ \mathbf{0} & \mathbf{X} & -\mathbf{X} \cdot y_{img} \end{pmatrix} \begin{pmatrix} \mathbf{p}^1 \\ \mathbf{p}^2 \\ \mathbf{p}^3 \end{pmatrix} = \mathbf{0}$$

$$\mathbf{A}_i \begin{pmatrix} \mathbf{p}^1 \\ \mathbf{p}^2 \\ \mathbf{p}^3 \end{pmatrix} = \mathbf{0} \quad \text{with} \quad \mathbf{A}_i \in \mathbb{R}^{2\times 12}$$

Calibration

- Take *N* corresponding points in image spage and world space: $\mathbf{x}_i^{img} \leftrightarrow \mathbf{X}_i \quad i = 1..N$
- For every correspondence point make a matrix $\mathbf{A}_i$:
- Matrix **A** assembled out has
  - 12 columns
  - 2N rows
- Remember: Camera has 11 DOF, while $(\mathbf{p}_1,\mathbf{p}_2,\mathbf{p}_3)^T$ has 12
  - Set: $\|(p_{11}, p_{12}, ..., p_{34})^T\| = 1$

Calibration

- rank(**A**)=11
  - If rank(**A**)=12: **Ap**=**0** --> single solution **p**=**0**
  - If at least 6 points are given rank(**A**) = 11
    - Restrictions:
      - Points may not be coplanar (if all points are coplanar rank(**A**) = 8)
      - Points may not lie on a *twisted cubic* (--> rank(**A**) < 11)
  - Due to noise: more than 6 points must be provided
    - System is usually overdetermined
    - Minimize: $\|\mathbf{Ap}\| = \mathbf{0}$
      - + normalization constrain

Calibration Algorithm

- **Direct Linear Transformation Algorithm (DLT)**

Given N corresponding points: $\mathbf{x}_i^{img} \leftrightarrow \mathbf{X}_i$

Find: Matrix **P** such that: $\mathbf{x}_i^{img} = \mathbf{P}\mathbf{X}_i$

For each correspondence create $\mathbf{A}_i$

Assemble matrix **A** out of $\mathbf{A}_i$

Use singular value decomposition: $\mathbf{A}=\mathbf{U}\mathbf{D}\mathbf{V}^T$

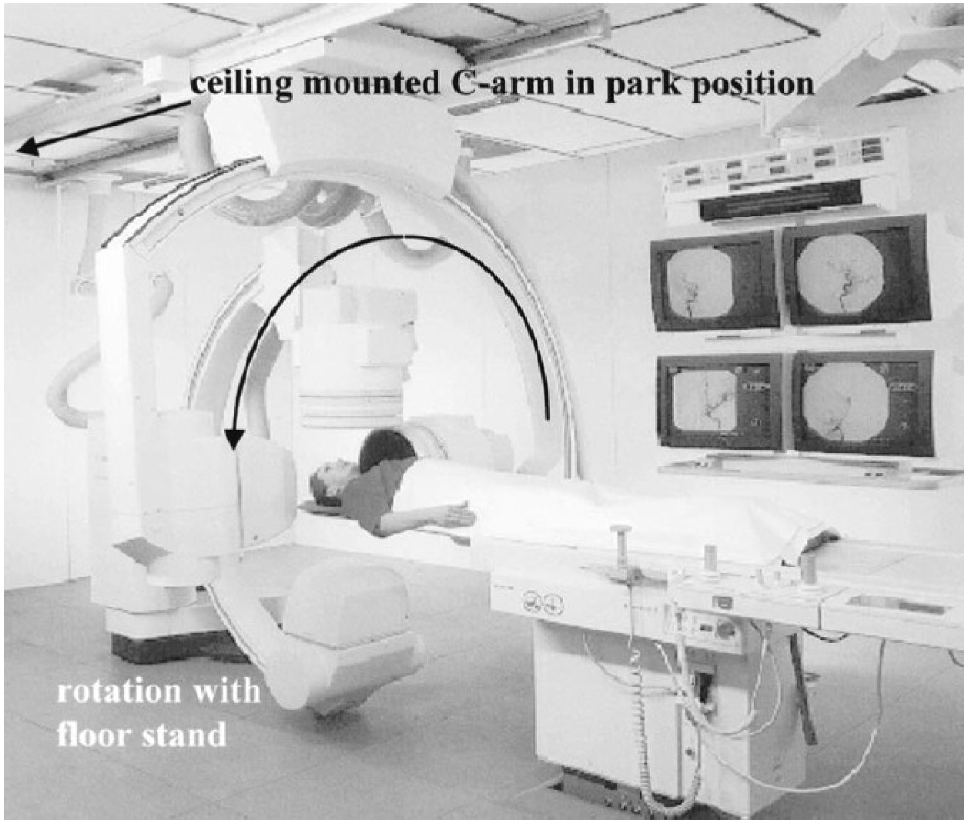
Pick the singular vector **p** corresponding to the smallest singular value

Summary

- Now we know
  - What a camera is and how it is described in terms of projective mappings
  - Out of a camera matrix we can calculate the extrinsic and intrinsic parameters
  - Given a set of world coordinates and a corresponding set of image coordinates we can calculate the matrix **P**

C-Arm CT

- C-Arm CT
- Detector and X-ray source rotate around patient
  - Up to 220 Degrees
  - Step: 0.4 degrees
  - i.e. 550 projections
- Table can moved (not considered here)
- Application: Intervention (e.g. During operations)

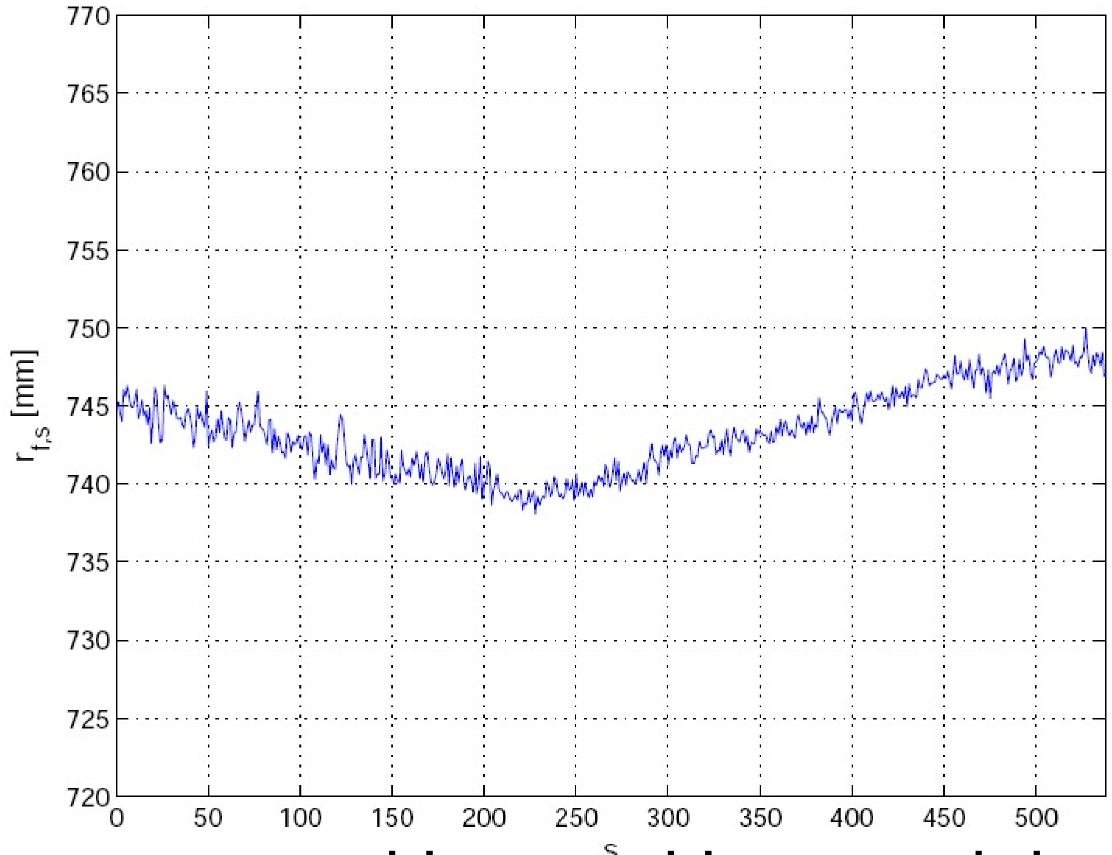


C-Arm CT

- Difficulties when applying C-ARM CT
  - Detector and source trajectory not an ideal circle arc (ideal Feldkamp geometry vs. Irregular Feldkamp geometry)
  - Perturbed by mechanical quantities
    - Inertia
    - Gravity
- Deviations are not negligible and must be corrected
- Applying regular Feldkamp algorithm results in artifacts

C-Arm CT

- Quantification of errors [7]:



- Distance of camera position to origin of volume coordinate system

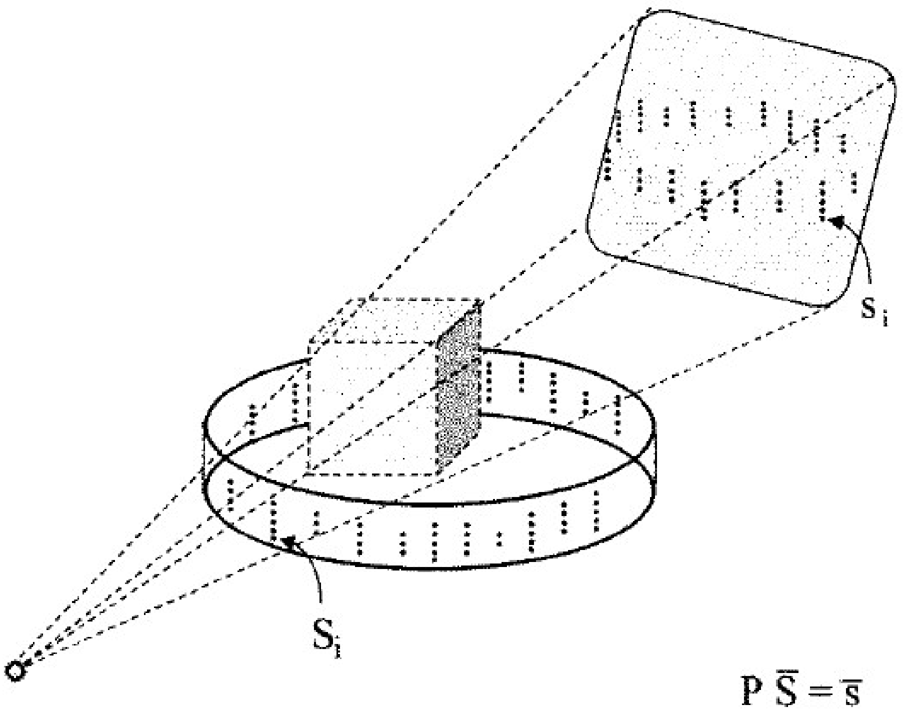
- Ideal: r = 745mm

C-Arm CT

- The good news: Errors are **reproducible**/**deterministic**
- Allows offline calibration:
  - Determine deviations from ideal
  - Use phantom
  - Typically done once a year for real C-Arm devices
- Estimate projection matrices $\mathbf{P}_i$ for all locations of the C-Arm
- Estimation of **P** see above

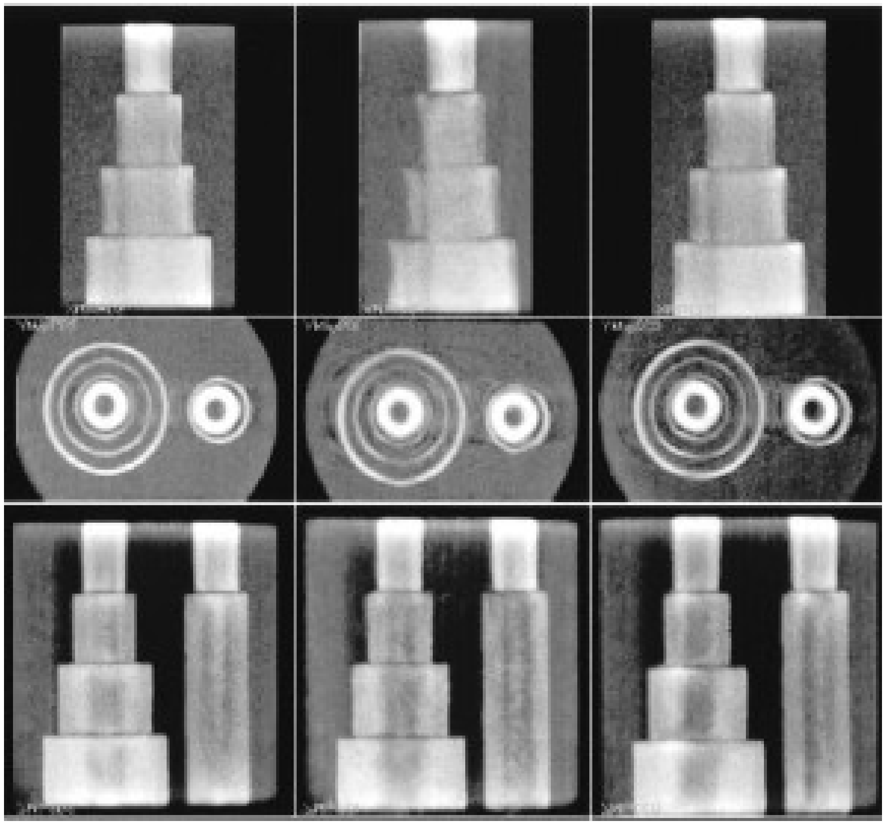
C-Arm CT

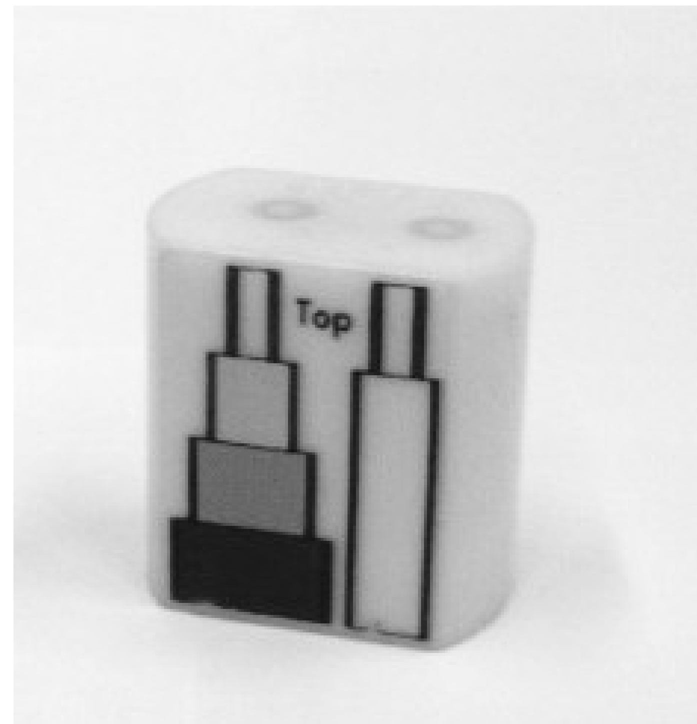
- Estimation of **P** in practice:
  - Place marker phantom in C-Arm CT
  - Use 100 – 150 corresponding points



C-Arm CT

- Marker phantom





C-Arm CT

- For reconstruction: Backprojection in homogeneous coordinates

```
For every projection i
  For every voxel (vx,vy,vz)
    (x,v,w)= P[i] * (vx,vy,vz,1)
    u = x/w; v = y/w;
    Backproject(u, v);
```

- Note that no decomposition of **P** is required

C-Arm CT

- Optimization: Incremental implementation
- Voxel position: $\mathbf{X} = \mathbf{X}_0 + (i \cdot \delta x, j \cdot \delta y, k \cdot \delta z, 0)^T$
- Transformation:

$$\begin{aligned} \mathbf{PX} &= \mathbf{P}\left[\mathbf{X}_0 + (i \cdot \delta x, j \cdot \delta y, k \cdot \delta z, 0)^T\right] \\ &= \mathbf{PX}_0 + i\mathbf{P}\delta x + j\mathbf{P}\delta y + k\mathbf{P}\delta z \end{aligned}$$

- Precalculation of: $\mathbf{PX}_0, \mathbf{P}\delta x, \mathbf{P}\delta y, \mathbf{P}\delta z$
- Algorithm almost incremental (besides perspective divide)

Summary

- Calibration of C-Arm CT Scanners
  - For every location on the arc, calculate projection matrix
  - Use those projection matrices while reconstructing
  - Can be implemented efficiently

Discussion

What questions do you have?

Literature

[0] Faugeras O., “Three-Dimensional Computer Vision”, MIT Press, 1993

[1] Hartley R., Zisserman A., “Multiple View Geometry”, Cambridge University Press, 2004

[2] Foley J. et al, “Computer Graphics – Principals and Practice, 2nd Edition”, Addison Wesley, 1996

[3] Shirley P. “Fundamentals of Computer Graphics”, A K Peters Ltd., 2002

[4] Hornegger J. Pauls D., “Medical Imaging I”, Lecture slides of lecture held at FAU Erlangen, winter term 2005

[5] Greiner G. “Computer Graphics – Lecture Transcript”, Lecture transcripts of lecture held at FAU, winter term 2003

[6] Wiesent K. et al, “Enhanced 3-D-Reconstruction Algorithm for C-Arm Systems Suitable for Interventional Procedures”, IEEE Transactions on Medical Imaging, Vol. 19, No. 5, May 2000

[7] Dennerlein F., “3D Image Reconstruction from Cone-Beam Projections using a Trajectory consisting of a Partial Circle and Line Segments”, Master Thesis in Computer Science, Patter Recognition Chair, FAU, 2004