# CS231A Midterm Review

May 2nd 2025

# Agenda

- Exam logistics
- Preparation tips
- Core topics

# Midterm logistics

- Format
  - 8 True/False questions, 6 multiple choice questions, and 3 short answer questions
- 80 Minutes during class time (1:30 PM - 2:50 PM, May 5)
- Gates B1- Basement floor of the Gates Building
- Practice exam
- SCPD students - 24 hours window
- *Open notes but closed Internet*
- No electronic devices are allowed (calculators are allowed)

# Preparing for the midterm

Resources:

- Lectures 1 - 10
- Problem Sets 0 - 2
- Course notes
- Recommended textbooks

Again: open notes!

- Focus on foundations & high-level understanding; you will have time to look up details.

# Core topics (1/2)

- General background
  - Necessary linear algebra
  - Homogeneous coordinates
  - Transformations
  - Formulating & solving least squares problems (when do we use an SVD?)
- Camera models
  - Perspective & non-perspective
  - Degrees of freedom
  - Distortion
  - Calibration
- Single view metrology
  - Vanishing points, vanishing lines

(disclaimer: this guide is not meant to be comprehensive)

# Core topics (2/2)

- Multiview geometry
  - Epipolar geometry; essential and fundamental matrices; 8-point algorithm
  - Structure from motion
  - Stereo
  - Perspective, affine, similarity ambiguities
- Active and volumetric stereo
  - Structured lighting
  - Space carving & Shadow carving & Voxel coloring
- Fitting & Matching
  - Least squares
  - RANSAC
  - Hough transforms
- Representations & Representation Learning (High Level Questions)

(disclaimer: this guide is not meant to be comprehensive)

# Necessary Linear Algebra

- 4 Basic spaces of a matrix: Null space, column space, row space
- Invertibility; Rank; Determinant
- Special matrices: identity matrix, orthogonal matrix, etc.
- QR decomposition: Decomposition of a matrix into orthogonal and upper triangular matrices.
- SVD:
    - Data Compression: Vectors corresponding to k largest singular values
    - Solve a (non-zero) vector in the null space of a matrix approximately: The vector corresponding to the smallest singular value

# Homogeneous Coordinates

- Augmented space for writing coordinates:

2D: $$\begin{bmatrix} x \\ y \end{bmatrix} \iff \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \iff \begin{bmatrix} wx \\ wy \\ w \end{bmatrix}$$

3D: $$\begin{bmatrix} x \\ y \\ z \end{bmatrix} \iff \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \iff \begin{bmatrix} wx \\ wy \\ wz \\ w \end{bmatrix}$$

# 2D Lines

Homogeneous coordinates give us a neat way of representing 2D lines as vectors/orthogonality constraints:

$$ax + by + c = 0$$

$$\begin{bmatrix} a & b & c \end{bmatrix} \begin{bmatrix} x & y & 1 \end{bmatrix}^T = 0$$

=> symmetry between lines and points

=> cross products suddenly becomes very useful!

# 2D Lines

How can we get the line connecting two points?

**Given:**

$$\begin{bmatrix} x_1 & y_1 & 1 \end{bmatrix}$$

$$\begin{bmatrix} x_2 & y_2 & 1 \end{bmatrix}$$

**Unknown:**

$$\begin{bmatrix} a & b & c \end{bmatrix}$$

**Subject to:**

$$\begin{bmatrix} a & b & c \end{bmatrix} \begin{bmatrix} x_1 & y_1 & 1 \end{bmatrix}^T = 0$$

$$\begin{bmatrix} a & b & c \end{bmatrix} \begin{bmatrix} x_2 & y_2 & 1 \end{bmatrix}^T = 0$$

**Solution:**

$$\begin{bmatrix} a \\ b \\ c \end{bmatrix} = \begin{bmatrix} x_1 \\ y_1 \\ 1 \end{bmatrix} \times \begin{bmatrix} x_2 \\ y_2 \\ 1 \end{bmatrix}$$

# 2D Lines

How can we get the intersection of two lines?

**Given:**

$$\begin{bmatrix} a_1 & b_1 & c_1 \end{bmatrix}$$

$$\begin{bmatrix} a_2 & b_2 & c_2 \end{bmatrix}$$

**Unknown:**

$$\begin{bmatrix} x & y & 1 \end{bmatrix}$$

**Subject to:**

$$\begin{bmatrix} a_1 & b_1 & c_1 \end{bmatrix} \begin{bmatrix} x & y & 1 \end{bmatrix}^T = 0$$

$$\begin{bmatrix} a_2 & b_2 & c_2 \end{bmatrix} \begin{bmatrix} x & y & 1 \end{bmatrix}^T = 0$$

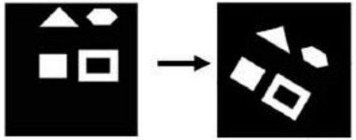
**Solution:**

$$\begin{bmatrix} wx \\ wy \\ w \end{bmatrix} = \begin{bmatrix} a_1 \\ b_1 \\ c_1 \end{bmatrix} \times \begin{bmatrix} a_2 \\ b_2 \\ c_2 \end{bmatrix}$$

# Transformations

**Isometric transformations:**
Distances preserved

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} R & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

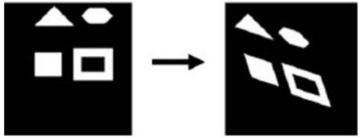


**Similarity transformations:**
Shapes preserved

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} SR & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}, \quad S = \begin{bmatrix} s & 0 \\ 0 & s \end{bmatrix}$$

**Affine transformations:**
Parallelism preserved

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} A & t \\ 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

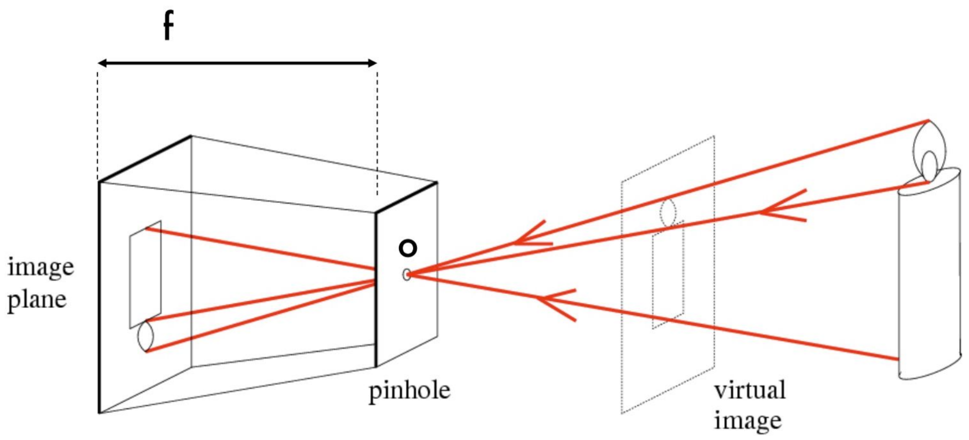


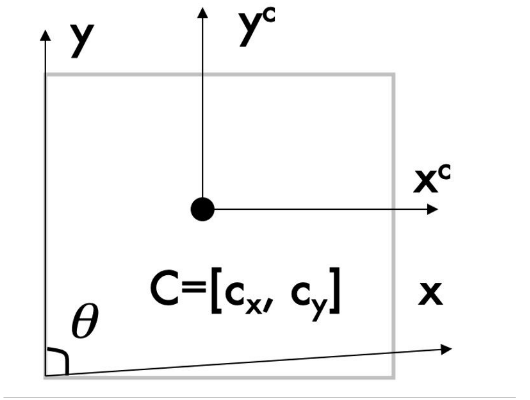
**Projective transformations:**
Lines preserved

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} A & t \\ v & b \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

# Pinhole Cameras

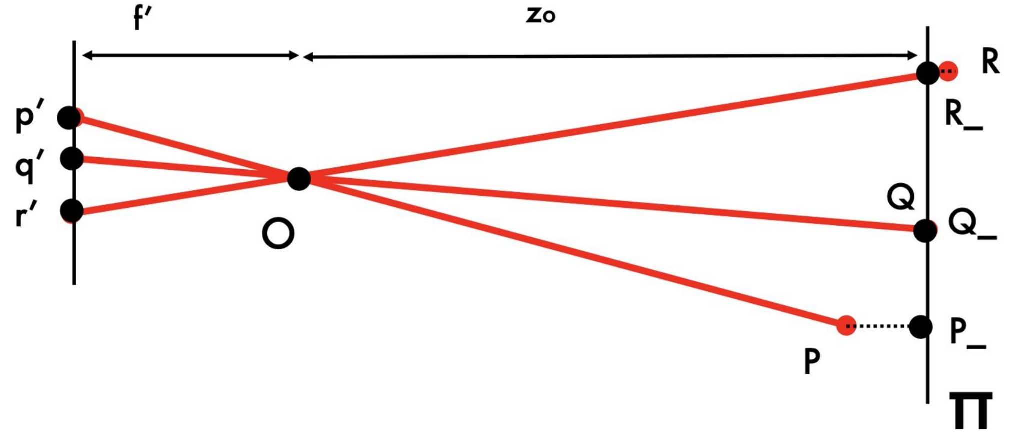
$$P' = \begin{bmatrix} \alpha & -\alpha\cot\theta & c_x & 0 \\ 0 & \dfrac{\beta}{\sin\theta} & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix}$$



# Camera Models

- Weak perspective projection
  - Useful when relative depth of the scene is **small** and **distant**
  - Magnification m is the ratio of the depth of the scene to camera focal length *f’*
  - Under what cases is the weak perspective accurate and why?




$$\begin{cases} x' = \dfrac{f'}{z} x \\[2em] y' = \dfrac{f'}{z} y \end{cases} \rightarrow \begin{cases} x' = \boxed{\dfrac{f'}{z_0}} x \\[1em] y' = \dfrac{f'}{z_0} y \end{cases}$$

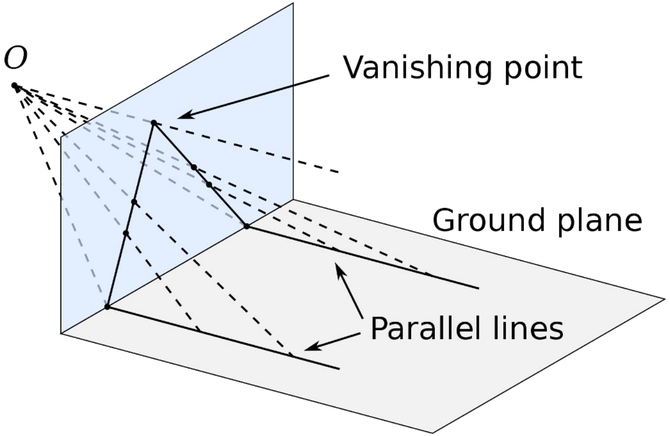
# Camera Calibration

- Intrinsic Parameters: K
- Extrinsic Parameters: R, T
- 11 DOF
  - 5 from K
  - 3 from R
  - 3 from T
- Degenerate cases
- Know how to construct the homogeneous linear system

$$P' = M\,P_w = \underbrace{K}_{\text{Internal parameters}}\underbrace{\begin{bmatrix} R & T \end{bmatrix}}_{\text{External parameters}} P_w$$

# Single View Metrology

Under projective transformation, parallel lines converge to a vanishing point:




$$\mathbf{v} = K\,\mathbf{d}$$
[Eq. 24]

$$\mathbf{n} = K^{T}\mathbf{l}_{\text{horiz}}$$
[Eq. 27]

$$\cos\theta = \frac{v_1^T \omega\, v_2}{\sqrt{v_1^T \omega\, v_1}\sqrt{v_2^T \omega\, v_2}}$$
[Eq. 28]

$$\xrightarrow{\theta = 90} \quad v_1^T \omega\, v_2 = 0$$
[Eq. 29]

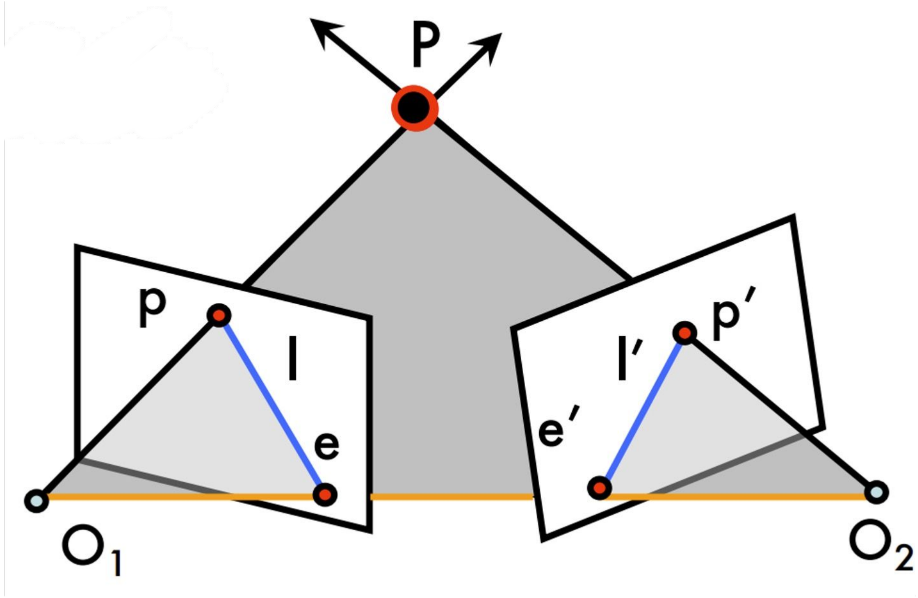
$$\omega = (K\,K^T)^{-1}$$
[Eq. 30]

Useful to:

- To calibrate the camera
- To estimate the geometry of the 3D world

We used this for camera calibration in PSET 1!

# Epipolar Geometry




**Essential matrix:**

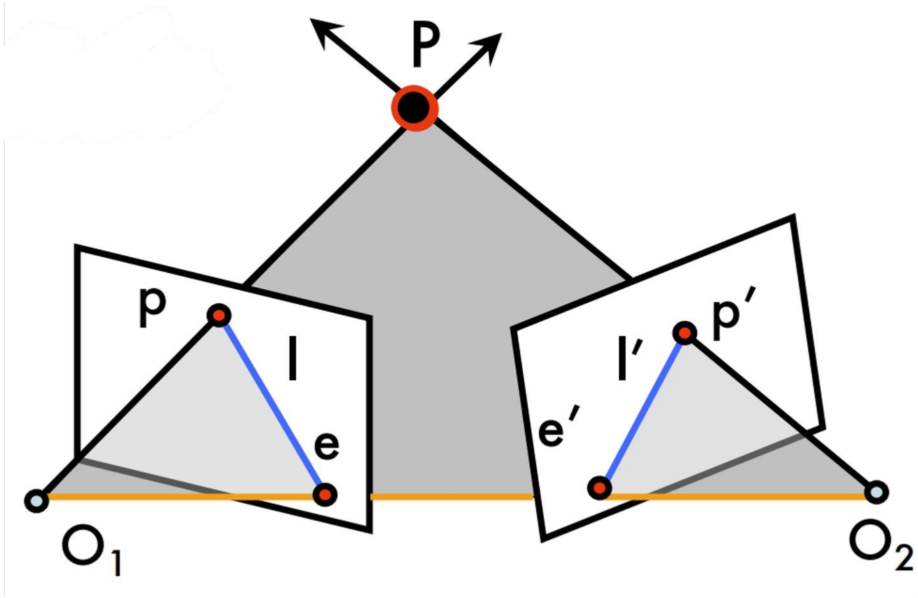
A point → epipolar line mapping for canonical cameras (K = I)

$$l' = E^T p$$

$$l = Ep'$$

$$p^T E p' = 0$$

# Epipolar Geometry




Given a 2D point in one camera, a correspondence in the other must lie on an epipolar line

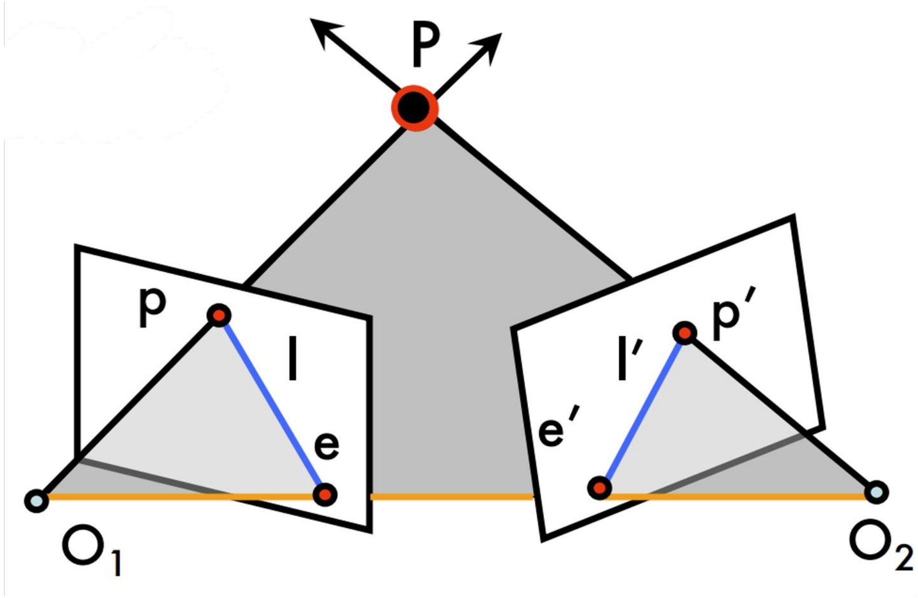
If ***p′*** is known, we can compute ***l*** and search for ***p*** using:

$$l^T p = 0$$

If ***p*** is known, we can compute ***l′*** and search for ***p′*** using:

$$l'^T p' = 0$$

# Epipolar Geometry




**Fundamental matrix:**

A point → epipolar line mapping for general projective cameras

$$l' = F^T p$$

$$l = F p'$$

$$p^T F p' = 0$$

# Epipolar Geometry

Computing the fundamental matrix with the 8-point algorithm:

$$\mathrm{p}^{\mathrm{T}}\ \mathrm{F}\ \mathrm{p}' = 0$$

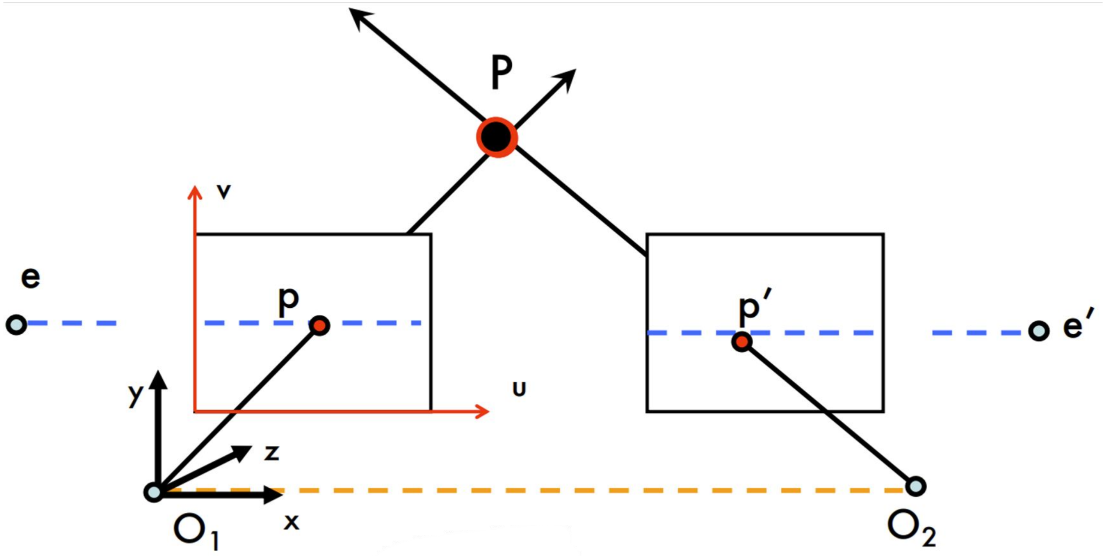
## Estimating F

$$\mathbf{W}\begin{pmatrix} u_1u'_1 & u_1v'_1 & u_1 & v_1u'_1 & v_1v'_1 & v_1 & u'_1 & v'_1 & 1 \\ u_2u'_2 & u_2v'_2 & u_2 & v_2u'_2 & v_2v'_2 & v_2 & u'_2 & v'_2 & 1 \\ u_3u'_3 & u_3v'_3 & u_3 & v_3u'_3 & v_3v'_3 & v_3 & u'_3 & v'_3 & 1 \\ u_4u'_4 & u_4v'_4 & u_4 & v_4u'_4 & v_4v'_4 & v_4 & u'_4 & v'_4 & 1 \\ u_5u'_5 & u_5v'_5 & u_5 & v_5u'_5 & v_5v'_5 & v_5 & u'_5 & v'_5 & 1 \\ u_6u'_6 & u_6v'_6 & u_6 & v_6u'_6 & v_6v'_6 & v_6 & u'_6 & v'_6 & 1 \\ u_7u'_7 & u_7v'_7 & u_7 & v_7u'_7 & v_7v'_7 & v_7 & u'_7 & v'_7 & 1 \\ u_8u'_8 & u_8v'_8 & u_8 & v_8u'_8 & v_8v'_8 & v_8 & u'_8 & v'_8 & 1 \end{pmatrix} \underbrace{\begin{pmatrix} F_{11} \\ F_{12} \\ F_{13} \\ F_{21} \\ F_{22} \\ F_{23} \\ F_{31} \\ F_{32} \\ F_{33} \end{pmatrix}}_{\mathbf{f}} = 0 \quad \text{[Eqs. 15]}$$

- Homogeneous system $\mathbf{W}\,\mathbf{f} = 0$

=> Solve with SVD, then project to rank 2

# Epipolar Geometry




Parallel images planes or rectification:
simplifies correspondence problem, moves epipoles to infinity

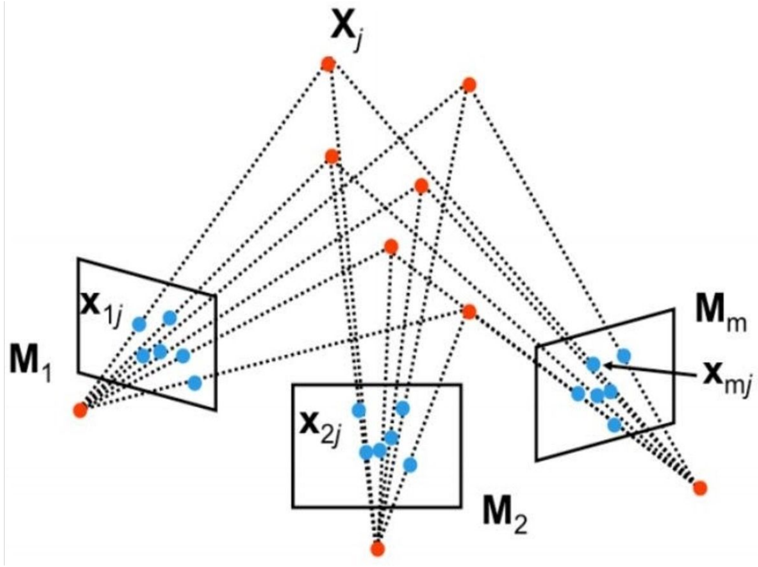
# Structure from Motion

Determining *structure* and *motion*

- Structure: **n** 3D points
- Motion: **m** projection matrices

You've implemented a few algorithms for this!

- Factorization
- Triangulation



# Factorization Method

$$\mathbf{D} = \underbrace{\begin{bmatrix} \hat{\mathbf{x}}_{11} & \hat{\mathbf{x}}_{12} & \cdots & \hat{\mathbf{x}}_{1n} \\ \hat{\mathbf{x}}_{21} & \hat{\mathbf{x}}_{22} & \cdots & \hat{\mathbf{x}}_{2n} \\ & & \ddots & \\ \hat{\mathbf{x}}_{m1} & \hat{\mathbf{x}}_{m2} & \cdots & \hat{\mathbf{x}}_{mn} \end{bmatrix}}_{(2m \times n)} = \underbrace{\begin{bmatrix} \mathbf{A}_1 \\ \mathbf{A}_2 \\ \vdots \\ \mathbf{A}_m \end{bmatrix}}_{\text{cameras } (2m \times 3)} \underbrace{\begin{bmatrix} \mathbf{X}_1 & \mathbf{X}_2 & \cdots & \mathbf{X}_n \end{bmatrix}}_{\text{points } (3 \times n)} \qquad \text{[Eq. 10]}$$

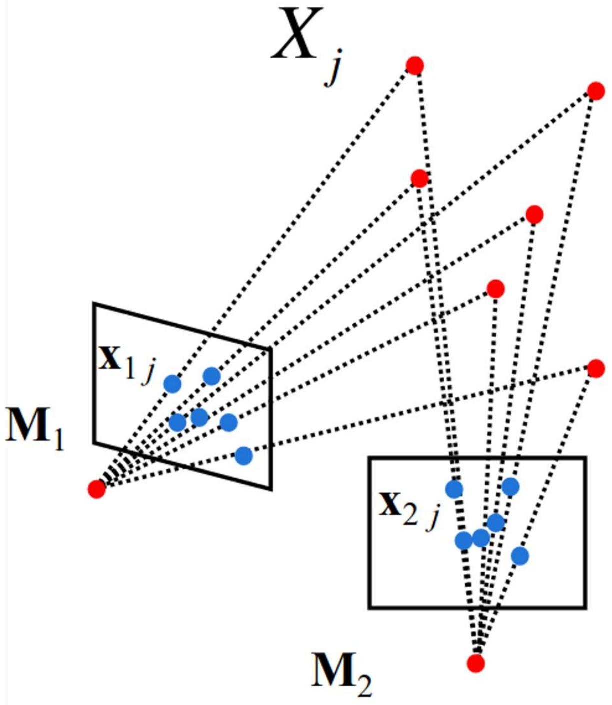
M

S

- Affine Structure from Motion
- Assume all points are visible
- SVD - solution not unique
- Ambiguities
  - Affine Ambiguity
  - Similarity Ambiguity

# Algebraic approach

- Compute fundamental matrix F
- Use F to estimate projective cameras
- Use these cameras to triangulate and estimate points in 3D
- Works with 2 views



# Bundle Adjustment

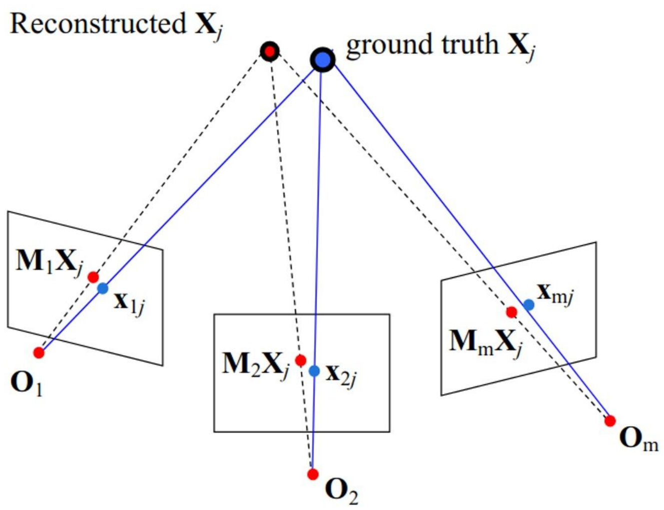
Non-linear method for refining structure and motion
Goal: minimize reprojection

error Advantages

- Handle large number of views
- Handle missing data

Limitations

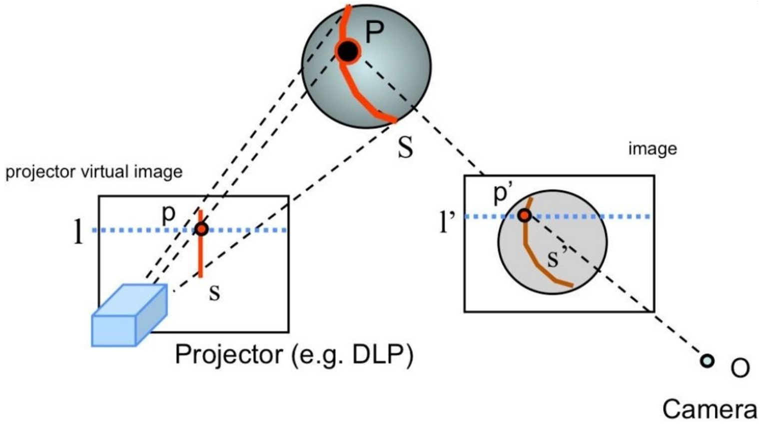
- Large minimization problem
- Require good initialization



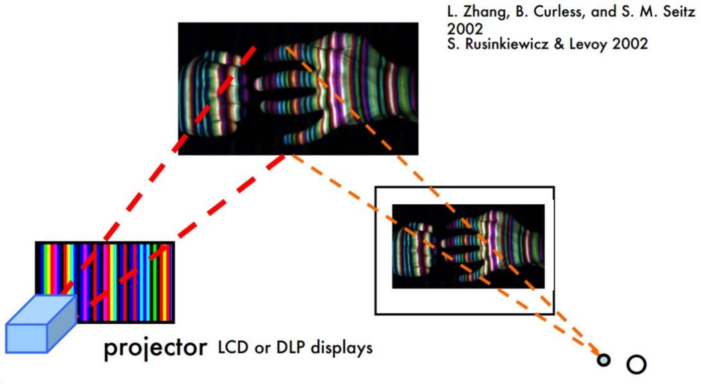
# Active Stereo

Active Stereo

- Replaces one camera with a projector
- Solves matching problem



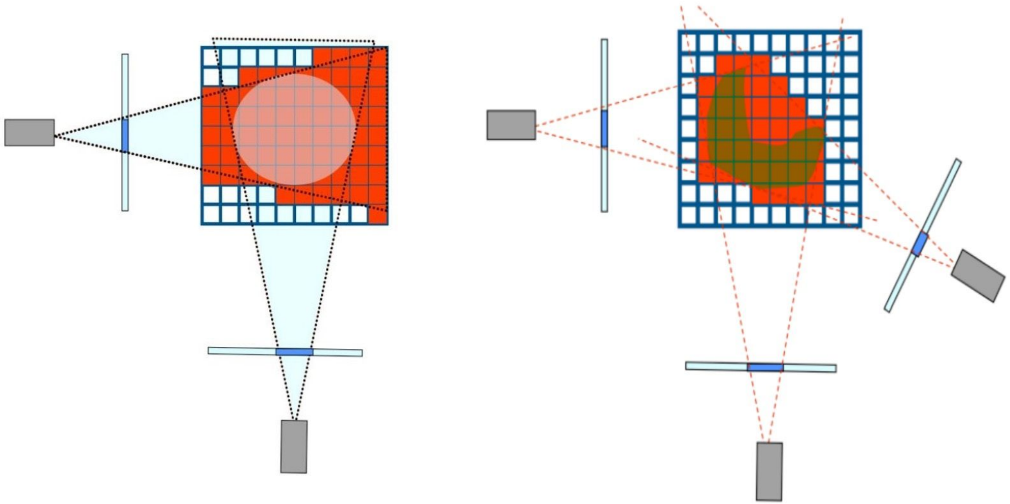




# Volumetric Stereo

Space carving

- Use contours and silhouettes
- Complexity: O(N^3)
- Octrees
- Conservative estimations
- Cannot carve concavity

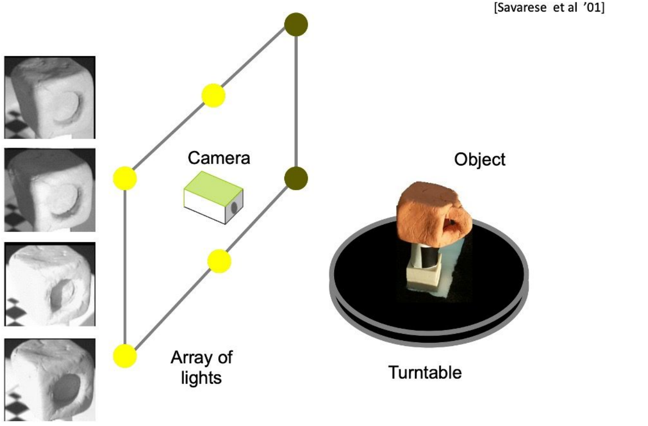


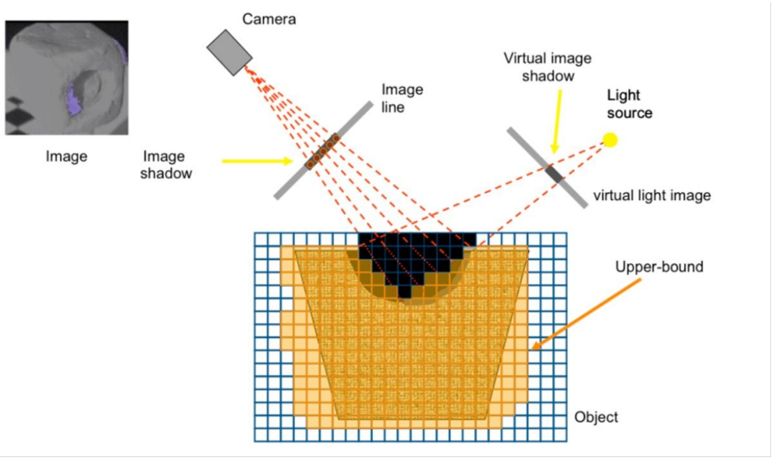
# Volumetric Stereo

[Savarese et al '01]

Shadow carving

- Use shadows
- Complexity: O(2N^3)
- Conservative estimations
- Can carve concavity
- Limitations with reflective & low albedo regions

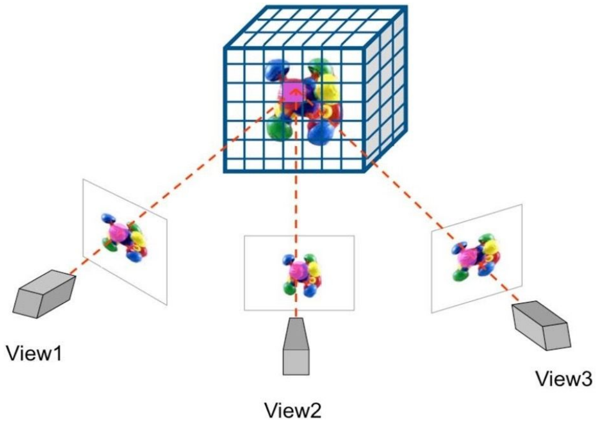


# Volumetric Stereo

Voxel coloring

- Use colors
- Complexity: O(LN^3)
- Model intrinsic scene colors and textures



# Fitting and Matching

- 3 Techniques:
  - Least Square Methods
    - Normal Equations
    - SVD
  - RANSAC
  - Hough Transform
- Advantages and disadvantages of each technique?

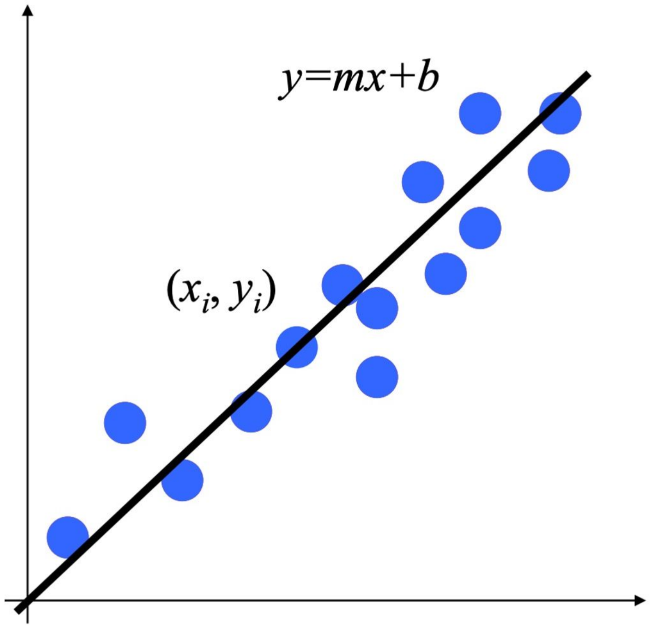
# Least square

- Find (m, b) to minimize the fitting error (residual):

$$\boxed{E = \sum_{i=1}^{n} (y_i - m x_i - b)^2}$$

- Normal Equation: $\boxed{h = \left(X^T X\right)^{-1} X^T Y}$

- Fail for vertical lines

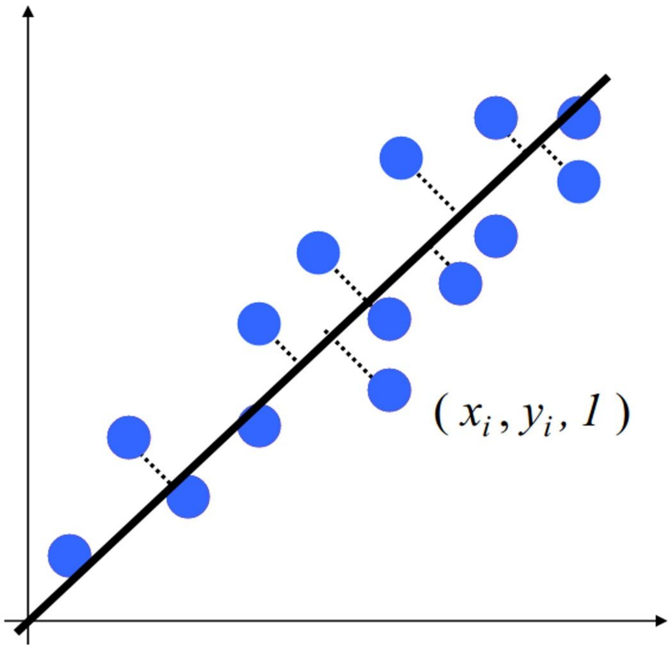


# Least square

- Find a line to minimize the sum of squared distance to the points

$$E = \sum_{i=1}^{n} (ax_i + by_i + d)^2$$

- Can be solved by SVD
- However, **susceptible to outliers!**

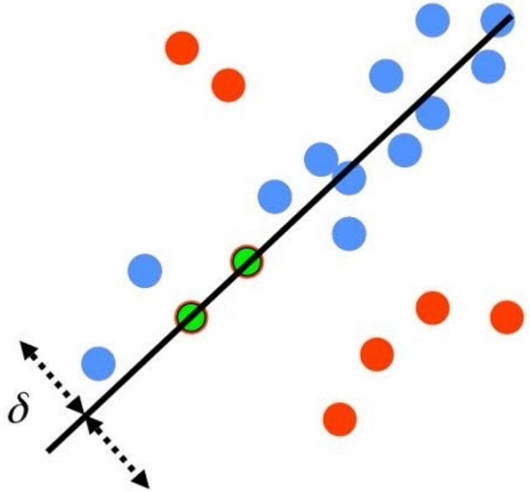


# RANSAC

***Ran**dom **sa**mple **c**onsensus*

For fitting a model to noisy data! Iterative approach:

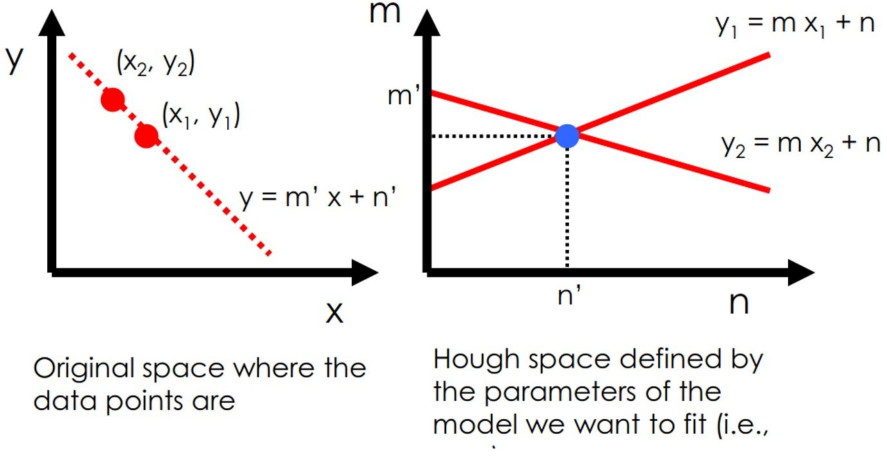
- Sample a subset of points
- Fit our model
- Count the total # of inliers that match this model
- Repeat for a finite number of iterations *M* and keep the set with the maximal number of inliers

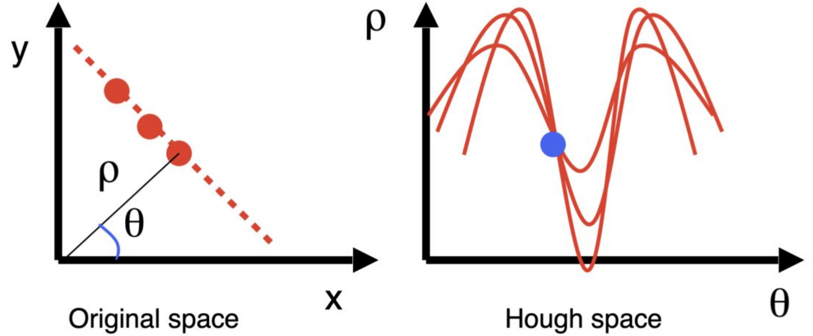


# Hough Transforms

Key idea for line fitting:

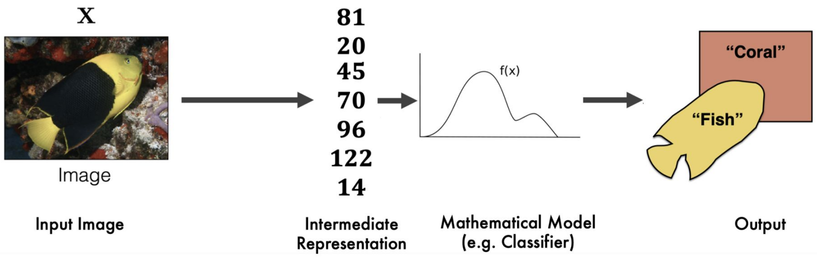
- Map points in (x,y) to a line in our Hough space
- Each point in our Hough space represents a line in our (x,y) space
- Intersection of lines in hough space = line
- Polar line representation
- Discretization and voting



# Representation Learning

- What is a representation?
- Input vs. intermediate vs. output representation
- What makes a good representation?
- Supervised vs. Unsupervised vs. Self-supervised learning



# Good Luck!

Questions