Agenda

● Exam logistics
● Preparation tips
● Core topics
Midterm logistics

- Format
  - 10 True/False questions, 5 multiple choice questions, and 4 short answer questions
- 80 Minutes during class time (1:30 PM - 2:50 PM, May 6)
- 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 and 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, null space of transposed matrix
- Invertibility; Rank; Determinant
- Special matrices: identity matrix, triangular matrix, orthogonal matrix
- 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:

\[
\begin{bmatrix}
  x \\
  y
\end{bmatrix} \leftrightarrow \begin{bmatrix}
  x \\
  y \\
  1
\end{bmatrix} \leftrightarrow \begin{bmatrix}
  wx \\
  wy \\
  w
\end{bmatrix}
\]

3D:

\[
\begin{bmatrix}
  x \\
  y \\
  z
\end{bmatrix} \leftrightarrow \begin{bmatrix}
  x \\
  y \\
  z \\
  1
\end{bmatrix} \leftrightarrow \begin{bmatrix}
  wx \\
  wy \\
  wz \\
  w
\end{bmatrix}
\]
Homogeneous coordinates give us a neat way of representing 2D lines as vectors/orthogonality constraints:

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

=> symmetry between lines and points

=> cross products suddenly becomes very useful!
2D Lines

How can we get the line connecting two points?

Given:

\[
\begin{bmatrix}
  x_1 & y_1 & 1 \\
  x_2 & y_2 & 1
\end{bmatrix}
\]

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:
\[
\begin{bmatrix}
a_1 & b_1 & c_1 \\
a_2 & b_2 & c_2 \\
\end{bmatrix}
\]

Unknown:
\[
\begin{bmatrix}
x \\
y \\
1 \\
\end{bmatrix}
\]

Subject to:
\[
\begin{align*}
[a_1 & b_1 & c_1] [x & y & 1]^T &= 0 \\
[a_2 & b_2 & c_2] [x & y & 1]^T &= 0
\end{align*}
\]

Solution:
\[
\begin{bmatrix}
w \times \\
w \cdot \\
w
\end{bmatrix} = \begin{bmatrix}
a_1 \\
b_1 \\
c_1
\end{bmatrix} \times \begin{bmatrix}
a_2 \\
b_2 \\
c_2
\end{bmatrix}
\]
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

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

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

\[ P' = \begin{bmatrix} \alpha & -\alpha \cot \theta & c_x & 0 \\ 0 & \frac{\beta}{\sin \theta} & c_y & 0 \\ 0 & 0 & 1 & 0 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \\ 1 \end{bmatrix} \]
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?

\[
\begin{align*}
  x' &= \frac{f'}{z} x \\
  y' &= \frac{f'}{z} y \\
  x' &= \frac{f'}{z_0} x \\
  y' &= \frac{f'}{z_0} y
\end{align*}
\]
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

\[
P' = M P_w = K [R \ T] P_w
\]
Single View Metrology

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

We used this for camera calibration in PSET 1!

\[ \mathbf{v} = \mathbf{K} \mathbf{d} \quad [\text{Eq. 24}] \]

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

\[ \cos \theta = \frac{\mathbf{v}_1^T \omega \mathbf{v}_2}{\sqrt{\mathbf{v}_1^T \omega \mathbf{v}_1 \mathbf{v}_2^T \omega \mathbf{v}_2}} \quad [\text{Eq. 28}] \]

\[ \theta = 90 \Rightarrow \mathbf{v}_1^T \omega \mathbf{v}_2 = 0 \quad [\text{Eq. 29}] \]

\[ \mathbf{\omega} = (\mathbf{K} \mathbf{K}^T)^{-1} \quad [\text{Eq. 30}] \]

Useful to:
- To calibrate the camera
- To estimate the geometry of the 3D world
Epipolar Geometry

Essential matrix:
A point $\rightarrow$ epipolar line mapping for canonical cameras ($K = I$)

\[
\begin{align*}
    l' &= E^T p \\
    l &= E p' \\
    p^T E p' &= 0
\end{align*}
\]
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 $\rightarrow$ 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:

\[ p^T F p' = 0 \]

\[ W \]

\[ \begin{pmatrix} u_1 u'_1 & u_1 v'_1 & u_1 v'_1 & v_1 u'_1 & v_1 u'_1 & v_1 & u'_1 & v'_1 \end{pmatrix} = 0 \]

[Eq. 15]

\[ F = \begin{pmatrix} F_{11} & F_{12} & F_{13} \\ F_{21} & F_{22} & F_{23} \\ F_{31} & F_{32} & F_{33} \end{pmatrix} \]

**Estimating F**

=> Solve with SVD, then project to rank 2

• Homogeneous system \[ W f = 0 \]
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

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

\[
D = \begin{bmatrix}
\hat{x}_{11} & \hat{x}_{12} & \cdots & \hat{x}_{1n} \\
\hat{x}_{21} & \hat{x}_{22} & \cdots & \hat{x}_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
\hat{x}_{m1} & \hat{x}_{m2} & \cdots & \hat{x}_{mn}
\end{bmatrix}
= \begin{bmatrix}
A_1 \\
A_2 \\
\vdots \\
A_m
\end{bmatrix}
\begin{bmatrix}
X_1 \\
X_2 \\
\vdots \\
X_n
\end{bmatrix}
\]

[Eq. 10]

points \((3 \times n)\)
cameras \((2m \times 3)\)
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

- 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

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

Voxel carving
- 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):

\[
E = \sum_{i=1}^{n} (y_i - mx_i - b)^2
\]

- Normal Equation:

\[
h = (X^T X)^{-1} X^T Y
\]

- 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
RANSAC

*Random sample consensus*

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
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
Good Luck!

Questions