3D Vision

Viktor Larsson

Spring 2019
## Schedule

<table>
<thead>
<tr>
<th>Date</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feb 18</td>
<td>Introduction</td>
</tr>
<tr>
<td>Feb 25</td>
<td><strong>Geometry, Camera Model, Calibration</strong></td>
</tr>
<tr>
<td>Mar 4</td>
<td>Features, Tracking / Matching</td>
</tr>
<tr>
<td>Mar 11</td>
<td><strong>Project Proposals by Students</strong></td>
</tr>
<tr>
<td>Mar 18</td>
<td>Structure from Motion (SfM) + papers</td>
</tr>
<tr>
<td>Mar 25</td>
<td>Dense Correspondence (stereo / optical flow) + papers</td>
</tr>
<tr>
<td>Apr 1</td>
<td>Bundle Adjustment &amp; SLAM + papers</td>
</tr>
<tr>
<td>Apr 8</td>
<td><strong>Student Midterm Presentations</strong></td>
</tr>
<tr>
<td>Apr 15</td>
<td>Multi-View Stereo &amp; Volumetric Modeling + papers</td>
</tr>
<tr>
<td>Apr 22</td>
<td>Easter break</td>
</tr>
<tr>
<td>Apr 29</td>
<td>3D Modeling with Depth Sensors + papers</td>
</tr>
<tr>
<td>May 6</td>
<td>3D Scene Understanding + papers</td>
</tr>
<tr>
<td>May 13</td>
<td>4D Video &amp; Dynamic Scenes + papers</td>
</tr>
<tr>
<td>May 20</td>
<td>papers</td>
</tr>
<tr>
<td>May 27</td>
<td><strong>Student Project Demo Day = Final Presentations</strong></td>
</tr>
</tbody>
</table>
3D Vision– Class 2

Projective Geometry and Camera Model

points, lines, planes, conics and quadrics
Transformations, camera model

Read tutorial chapter 2 and 3.1
http://www.cs.unc.edu/~marc/tutorial/

Chapters 1, 2 and 5 in Hartley and Zisserman 1st edition
Or Chapters 2, 3 and 6 in 2nd edition
See also Chapter 2 in Szeliski book
Topics Today

- Lecture intended as a review of material covered in Computer Vision lecture
- Probably the hardest lecture (since very theoretic) in the class ...
- ... but fundamental for any type of 3D Vision application
- Key takeaways:
  - 2D primitives (points, lines, conics) and their transformations
  - 3D primitives and their transformations
  - Camera model and camera calibration
Overview

• 2D Projective Geometry

• 3D Projective Geometry

• Camera Models & Calibration
2D Projective Geometry?

Projections of planar surfaces

Measure distances

2D Projective Geometry?

Discovering details

Piero della Francesca, La Flagellazione di Cristo (1460)

2D Projective Geometry?

Image Stitching
2D Projective Geometry?

Image Stitching
2D Euclidean Transformations

- Rotation (around origin)
  \[
  \begin{pmatrix}
  x' \\
  y'
  \end{pmatrix} = \begin{bmatrix}
  \cos \alpha & -\sin \alpha \\
  \sin \alpha & \cos \alpha
  \end{bmatrix} \begin{pmatrix}
  x \\
  y
  \end{pmatrix}
  \]

- Translation
  \[
  \begin{pmatrix}
  x'' \\
  y''
  \end{pmatrix} = \begin{pmatrix}
  x' \\
  y'
  \end{pmatrix} + \begin{pmatrix}
  t_x \\
  t_y
  \end{pmatrix}
  \]

- “Extended coordinates”
  \[
  \begin{pmatrix}
  x'' \\
  y'' \\
  1
  \end{pmatrix} = \begin{bmatrix}
  \cos \alpha & -\sin \alpha & t_x \\
  \sin \alpha & \cos \alpha & t_y \\
  0 & 0 & 1
  \end{bmatrix} \begin{pmatrix}
  x \\
  y \\
  1
  \end{pmatrix}
  \]
Homogeneous Coordinates

Homogenous coordinates

\[ \begin{pmatrix} x \\ y \end{pmatrix} \rightarrow \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}, \quad w \neq 0 \]

Equivalence class of vectors

\[ \begin{pmatrix} 3 \\ -2 \\ 1 \end{pmatrix} = \begin{pmatrix} 6 \\ -4 \\ 2 \end{pmatrix} = \begin{pmatrix} -9 \\ 6 \\ -3 \end{pmatrix} \]

2D projective space: \( \mathbb{P}^2 = \mathbb{R}^3 \setminus \{(0,0,0)\} \)
Homogeneous Coordinates

(Homogeneous) representation of 2D line:

\[ ax + by + c = 0 \]
\[ \left( a, b, c \right)^T \left( x, y, 1 \right) = 0 \]

The point \( x \) lies on the line \( l \) if and only if \( l^T x = 0 \)

Note that scale is unimportant for incidence relation

\[ \left( a, b, c \right)^T \sim k \left( a, b, c \right)^T, \quad k \neq 0 \]
\[ \left( x, y, 1 \right)^T \sim k \left( x, y, 1 \right)^T, \quad k \neq 0 \]

**Homogeneous coordinates** \( \left( x_1, x_2, x_3 \right)^T \) but only 2DOF

**Inhomogeneous coordinates** \( \left( x, y \right)^T = \left( x_1 / x_3, x_2 / x_3 \right)^T \)
2D Projective Transformations

Definition:

A *projectivity* is an invertible mapping $h$ from $\mathbb{P}^2$ to itself such that three points $x_1, x_2, x_3$ lie on the same line if and only if $h(x_1), h(x_2), h(x_3)$ do.

Theorem:

A mapping $h : \mathbb{P}^2 \to \mathbb{P}^2$ is a *projectivity* if and only if there exist a non-singular 3x3 matrix $H$ such that for any point in $\mathbb{P}^2$ represented by a vector $x$ it is true that $h(x) = Hx$.

Definition: Projective transformation

$$
\begin{pmatrix}
  x'_1 \\
  x'_2 \\
  x'_3
\end{pmatrix} =
\begin{bmatrix}
  h_{11} & h_{12} & h_{13} \\
  h_{21} & h_{22} & h_{23} \\
  h_{31} & h_{32} & h_{33}
\end{bmatrix}
\begin{pmatrix}
  x_1 \\
  x_2 \\
  x_3
\end{pmatrix}
$$

or

$$
x' = Hx
$$

8DOF

projectivity = collineation = proj. transformation = *homography*
Hierarchy of 2D Transformations

**Projective 8dof**

\[
\begin{bmatrix}
h_{11} & h_{12} & h_{13} \\
h_{21} & h_{22} & h_{23} \\
h_{31} & h_{32} & h_{33}
\end{bmatrix}
\]

**Affine 6dof**

\[
\begin{bmatrix}
a_{11} & a_{12} & t_x \\
a_{21} & a_{22} & t_y \\
0 & 0 & 1
\end{bmatrix}
\]

**Similarity 4dof**

\[
\begin{bmatrix}
sr_{11} & sr_{12} & t_x \\
sr_{21} & sr_{22} & t_y \\
0 & 0 & 1
\end{bmatrix}
\]

**Euclidean 3dof**

\[
\begin{bmatrix}
r_{11} & r_{12} & t_x \\
r_{21} & r_{22} & t_y \\
0 & 0 & 1
\end{bmatrix}
\]

**Transformed squares**

**Invariants**

Concurrency, collinearity, order of contact (intersection, tangency, inflection, etc.), cross ratio

Parallelism, ratio of areas, ratio of lengths on parallel lines (e.g. midpoints), linear combinations of vectors (centroids), **The line at infinity** \( l_\infty \)

Ratios of lengths, angles, The circular points \( I, J \)

Absolute lengths, angles, areas
Working with Homogeneous Coordinates

- “Homogenize”: $\begin{pmatrix} x \\ y \\ 1 \end{pmatrix} \mapsto \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$

- Apply $H$: $\begin{pmatrix} x'' \\ y'' \\ z'' \end{pmatrix} = H \begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$

- De-homogenize: $\begin{pmatrix} x'' \\ y'' \\ z'' \end{pmatrix} \mapsto \begin{pmatrix} x''/z'' \\ y''/z'' \end{pmatrix} = \begin{pmatrix} x' \\ y' \end{pmatrix}$
Lines to Points, Points to Lines

- Intersections of lines

Find \( x \) such that
\[
\begin{align*}
  l_1^T x &= 0 \\
  l_2^T x &= 0
\end{align*}
\]
\[ x = l_1 \times l_2 \]

- Line through two points

Find \( l \) such that
\[
\begin{align*}
  l^T x_1 &= 0 \\
  l^T x_2 &= 0
\end{align*}
\]
\[ l = x_1 \times x_2 \]
Transformation of Points and Lines

- For a point transformation
  \[ x' = Hx \]
- Transformation for lines
  \[ l' = H^{-T}l \]

\[ l^T x = 0 \quad \rightarrow \quad l^T (H^{-1}H)x = 0 \quad \rightarrow \quad (H^{-T}l)^T Hx = 0 \]
Ideal Points

- Intersections of parallel lines?

\[ l_1 \times l_2 = \begin{pmatrix} a \\ b \\ c \end{pmatrix} \times \begin{pmatrix} a \\ b \\ c' \end{pmatrix} = (c' - c) \begin{pmatrix} b \\ -a \\ 0 \end{pmatrix} \]

- Parallel lines intersect in \textit{Ideal Points} \((x_1, x_2, 0)^\top\)
Ideal Points

• Ideal points correspond to directions

\[ l_1 = (a, b, c) \]

\[ (a, b) \quad (b, -a) \]

• Unaffected by translation

\[
\begin{bmatrix}
  r_{11} & r_{12} & t_x \\
  r_{21} & r_{22} & t_y \\
  0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
  x \\
  y \\
  0
\end{bmatrix}
= \begin{bmatrix}
  r_{11}x + r_{12}y \\
  r_{21}x + r_{22}y \\
  0
\end{bmatrix}
\]
The Line at Infinity

• Line through two ideal points?

\[
\begin{pmatrix} x \\ y \\ 0 \end{pmatrix} \times \begin{pmatrix} x' \\ y' \\ 0 \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ xy' - x'y \end{pmatrix} = \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = l_\infty
\]

• Line at infinity \( l_\infty = (0,0,1)^T \) intersects all ideal points

\[
l_\infty^T x = l_\infty^T \begin{pmatrix} x_1 \\ x_2 \\ x_3 \end{pmatrix} = x_3 = 0
\]

\[\mathbb{P}^2 = \mathbb{R}^2 \cup l_\infty\]  

Note that in \( \mathbb{P}^2 \) there is no distinction between ideal points and others
The Line at Infinity

The line at infinity $l_\infty = (0,0,1)^T$ is a fixed line under a projective transformation $H$ if and only if $H$ is an affinity (affine transformation).

$$l'_\infty = H_A^{-T} l_\infty = \begin{bmatrix} A^{-T} & 0 \\ -t^T A^{-T} & 1 \end{bmatrix} \begin{pmatrix} 0 \\ 0 \\ 1 \end{pmatrix} = l_\infty$$

Affine trans. $H_A = \begin{bmatrix} A & t \\ 0^T & 1 \end{bmatrix}$

Note: not fixed pointwise
Conics

- Curve described by 2$^{nd}$-degree equation in the plane

1. Parabola
2. Ellipse
3. Hyperbola

Conics

• Curve described by 2\textsuperscript{nd}-degree equation in the plane

\[ ax^2 + bxy + cy^2 + dx + ey + f = 0 \]

or homogenized \( x \mapsto x_1/x_3, y \mapsto x_2/x_3 \)

\[ ax_1^2 + bx_1x_2 + cx_2^2 + dx_1x_3 + ex_2x_3 + fx_3^2 = 0 \]

or in matrix form \( x^T C x = 0 \)

\[
\begin{pmatrix}
  x_1 & x_2 & x_3
\end{pmatrix}
\begin{bmatrix}
  a & b/2 & d/2 \\
  b/2 & c & e/2 \\
  d/2 & e/2 & f
\end{bmatrix}
\begin{pmatrix}
  x_1 \\
  x_2 \\
  x_3
\end{pmatrix} = 0
\]

• 5DOF (degrees of freedom): \( \{a:b:c:d:e:f\} \) (defined up to scale)
Five Points Define a Conic

For each point the conic passes through

\[ ax_i^2 + bx_iy_i + cy_i^2 + dx_i + ey_i + f = 0 \]

or

\[ (x_i^2, x_iy_i, y_i^2, x_i, y_i, 1)c = 0 \quad c = (a,b,c,d,e,f)^T \]

stacking constraints yields

\[
\begin{bmatrix}
  x_1^2 & x_1y_1 & y_1^2 & x_1 & y_1 & 1 \\
  x_2^2 & x_2y_2 & y_2^2 & x_2 & y_2 & 1 \\
  x_3^2 & x_3y_3 & y_3^2 & x_3 & y_3 & 1 \\
  x_4^2 & x_4y_4 & y_4^2 & x_4 & y_4 & 1 \\
  x_5^2 & x_5y_5 & y_5^2 & x_5 & y_5 & 1 \\
\end{bmatrix}c = 0
\]
Tangent Lines to Conics

The line $l$ tangent to $C$ at point $x$ on $C$ is given by $l = Cx$
Dual Conics

- A line tangent to the conic $C$ satisfies $1^T C^* 1 = 0$
- In general ($C$ full rank): $C^* = C^{-1}$
- Dual conics = line conics = conic envelopes
Degenerate Conics

- A conic is degenerate if matrix $C$ is not of full rank

  $$C = lm^T + ml^T$$

  e.g. two lines (rank 2)

- Degenerate line conics: 2 points (rank 2), double point (rank 1)

  $$C = ll^T$$

- Note that for degenerate conics \( (C^*)^* \neq C \)
Transformation of Points, Lines and Conics

- For a point transformation
  \[ x' = Hx \]
- Transformation for lines
  \[ l' = H^{-T}l \]
- Transformation for conics
  \[ C' = H^{-T}CH^{-1} \]
- Transformation for dual conics
  \[ C^{*'} = HC^*H^T \]
Application: Removing Perspective

Two stages:
- From perspective to affine transformation via the line at infinity
- From affine to similarity transformation via the circular points
Affine Rectification

projection

\[
\mathbf{I} = \mathbf{H}_p(l_\infty)
\]

affine rectification

\[
\begin{pmatrix}
1 & 1 & 0 \\
0 & 1 & 0 \\
0 & 0 & 1
\end{pmatrix}^{-\mathbf{T}}
\begin{pmatrix}
l_1 \\
l_2 \\
l_3
\end{pmatrix}
= 
\begin{pmatrix}
a_{11} & a_{12} & l_1 \\
a_{21} & a_{22} & l_2 \\
0 & 0 & 1/l_3
\end{pmatrix}
\begin{pmatrix}
l_1 \\
l_2 \\
l_3
\end{pmatrix}
= 
\begin{pmatrix}
0 \\
0 \\
1
\end{pmatrix}
\]

metric rectification
Affine Rectification

\[ \mathbf{v}_1 = \mathbf{l}_1 \times \mathbf{l}_2 \]

\[ \mathbf{v}_2 = \mathbf{l}_3 \times \mathbf{l}_4 \]

\[ \mathbf{l}_\infty = \mathbf{v}_1 \times \mathbf{v}_2 \]
Metric Rectification

• Need to measure a quantity that is not invariant under affine transformations
The Circular Points

\[ I = \begin{pmatrix} 1 \\ i \\ 0 \end{pmatrix} \quad J = \begin{pmatrix} 1 \\ -i \\ 0 \end{pmatrix} \]

The circular points \( I, J \) are fixed points under the projective transformation \( H \) iff \( H \) is a similarity
The Circular Points

• every circle intersects $l_\infty$ at the “circular points”

\[ x_1^2 + x_2^2 + dx_1x_3 + ex_2x_3 + fx_3^2 = 0 \]
\[ x_1^2 + x_2^2 = 0 \]
\[ x_3 = 0 \]

\[ I = (1,i,0)^T \]
\[ J = (1,-i,0)^T \]

• Algebraically, encodes orthogonal directions

\[ I = (1,0,0)^T + i(0,1,0)^T \]
Conic Dual to the Circular Points

\[ C^*_\infty = \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 0 \end{bmatrix} \]

\[ C^*_\infty = IJ^T + JJ^T \]

\[ C^*_\infty = H_S C^*_\infty H_S^T \]

The dual conic \( C^*_\infty \) is fixed conic under the projective transformation \( H \) iff \( H \) is a similarity
Measuring Angles via the Dual Conic

- Euclidean: \( \mathbf{l} = (l_1, l_2, l_3)^T \quad \mathbf{m} = (m_1, m_2, m_3)^T \)
  \[
  \cos \theta = \frac{l_1 m_1 + l_2 m_2}{\sqrt{(l_1^2 + l_2^2)(m_1^2 + m_2^2)}}
  \]

- Projective: \( \cos \theta = \frac{\mathbf{l}^T \mathbf{C}^* \mathbf{m}}{\sqrt{(\mathbf{l}^T \mathbf{C}^* \mathbf{l})(\mathbf{m}^T \mathbf{C}^* \mathbf{m})}} \)

- Knowing the dual conic on the projective plane, we can measure Euclidean angles!
Metric Rectification

- Dual conic under affinity

\[
\mathbf{C}_\infty^* = \begin{pmatrix} \mathbf{A} & \mathbf{t} \\ \mathbf{0}^T & 1 \end{pmatrix} \begin{pmatrix} \mathbf{I} & \mathbf{0} \\ \mathbf{0}^T & 0 \end{pmatrix} \begin{pmatrix} \mathbf{A}^T & 0 \\ \mathbf{t}^T & 1 \end{pmatrix} = \begin{pmatrix} \mathbf{A}\mathbf{A}^T & \mathbf{0} \\ \mathbf{0}^T & 0 \end{pmatrix}
\]

- \( \mathbf{S} = \mathbf{A}\mathbf{A}^T \) symmetric, estimate from two pairs of orthogonal lines (due to \( \mathbf{l}^T \mathbf{C}^* \mathbf{m} = 0 \))

\[
\left( \mathbf{l}_1^{'\prime} \mathbf{m}_1^{'\prime}, \mathbf{l}_1^{'\prime} \mathbf{m}_2^{'\prime} + \mathbf{l}_2^{'\prime} \mathbf{m}_1^{'\prime}, \mathbf{l}_2^{'\prime} \mathbf{m}_2^{'\prime} \right) \mathbf{s} = 0
\]

Note: Result defined up to similarity
Update to Euclidean Space

- Metric space: Measure ratios of distances
- Euclidean space: Measure absolute distances
- Can we update metric to Euclidean space?
- Not without additional information
Important Points so far ...

• Definition of 2D points and lines
• Definition of homogeneous coordinates
• Definition of projective space
• Effect of transformations on points, lines, conics
• Next: Analogous concepts in 3D
Overview

- 2D Projective Geometry
- 3D Projective Geometry
- Camera Models & Calibration
3D Points and Planes

- 2D: duality point - line, 3D: duality point - plane

- Homogeneous representation of 3D points and planes
  \[ \pi_1 X_1 + \pi_2 X_2 + \pi_3 X_3 + \pi_4 X_4 = 0 \]

- The point \( \mathbb{X} \) lies on the plane \( \mathbb{\pi} \) if and only if
  \[ \mathbb{\pi}^T \mathbb{X} = 0 \]

- The plane \( \mathbb{\pi} \) goes through the point \( \mathbb{X} \) if and only if
  \[ \mathbb{\pi}^T \mathbb{X} = 0 \]
Planes from Points

Solve $\pi$ from $X_1^T \pi = 0$, $X_2^T \pi = 0$ and $X_3^T \pi = 0$

$$
\begin{bmatrix}
X_1^T \\
X_2^T \\
X_3^T
\end{bmatrix} \pi = 0 \quad \text{(solve $\pi$ as right nullspace of } \begin{bmatrix}
X_1^T \\
X_2^T \\
X_3^T
\end{bmatrix})
$$
Points from Planes

Solve $X$ from $\pi_1^T X = 0$, $\pi_2^T X = 0$ and $\pi_3^T X = 0$

$$\begin{bmatrix} \pi_1^T \\ \pi_2^T \\ \pi_3^T \end{bmatrix} X = 0 \quad \text{(solve } X \text{ as right nullspace of)} \quad \begin{bmatrix} \pi_1^T \\ \pi_2^T \\ \pi_3^T \end{bmatrix}$$

Representing a plane by its span

$$X = M x \quad M = \begin{bmatrix} X_1 & X_2 & X_3 \end{bmatrix} \in R^{4 \times 3}$$

$$\pi^T M = 0$$
Quadrics and Dual Quadrics

\[ X^T Q X = 0 \quad (Q : 4 \times 4 \text{ symmetric matrix}) \]

- 9 DOF (up to scale)
- In general, 9 points define quadric
- \( \det(Q) = 0 \leftrightarrow \text{degenerate quadric} \)
- tangent plane \( = QX \)
- Dual quadric: \( X^T Q^* = 0 \) (\( Q^* \) adjoint)
- relation to quadric \( Q^* = Q^{-1} \) (non-degenerate)

Transformation of 3D points, planes and quadrics

- Transformation for points (2D equivalent)
  \[ X' = HX \]
  \[ x' = Hx \]

- Transformation for planes
  \[ l' = H^{-T}l \]
  \[ l' = H^{-T}l \]

- Transformation for quadrics
  \[ Q' = H^{-T}QH^{-1} \]
  \[ C' = H^{-T}CH^{-1} \]

- Transformation for dual quadrics
  \[ Q'^* = HQ^*H^T \]
  \[ C'^* = HC^*H^T \]
The Plane at Infinity

The plane at infinity $\pi_\infty = (0, 0, 0, 1)^T$ is a fixed plane under a projective transformation $H$ iff $H$ is an affinity

$$
\pi'_\infty = H_A^{-T} \pi_\infty = \begin{bmatrix}
A^{-T} & 0 \\
-t^T A^{-T} & 1
\end{bmatrix} \begin{pmatrix} 0 \\ 0 \\ 0 \\ 1 \end{pmatrix} = \pi_\infty
$$

1. canonical position $= (0,0,0,1)^T$
2. contains all directions $D = (X_1, X_2, X_3, 0)^T$
3. two planes are parallel $\iff$ line of intersection in $\pi_\infty$
4. line $\parallel$ line (or plane) $\iff$ point of intersection in $\pi_\infty$
5. 2D equivalent: line at infinity
Hierarchy of 3D Transformations

Euclidean
6dof

\[
\begin{bmatrix}
R & t \\
0^T & 1
\end{bmatrix}
\]
Hierarchy of 3D Transformations

- projective
  - Plane at infinity
- affine
- similarity

Absolute conic
The Absolute Conic

- The absolute conic $\Omega_\infty$ is a (point) conic on $\pi_\infty$
- In a metric frame: $\begin{cases} X_1^2 + X_2^2 + X_3^2 \\ X_4 \end{cases} = 0$

or conic for directions: $\left( X_1, X_2, X_3 \right) I \left( X_1, X_2, X_3 \right)^T$

(with no real points)

The absolute conic $\Omega_\infty$ is a fixed conic under the projective transformation $H$ iff $H$ is a similarity

1. $\Omega_\infty$ is only fixed as a set
2. Circles intersect $\Omega_\infty$ in two circular points
3. Spheres intersect $\pi_\infty$ in $\Omega_\infty$
The absolute dual quadric $\Omega^*_\infty$ is a fixed quadric under the projective transformation $H$ iff $H$ is a similarity.

1. 8 dof
2. plane at infinity $\pi_\infty$ is the nullvector of $\Omega_\infty$
3. angles: $\cos \theta = \frac{\pi_1^T \Omega^*_\infty \pi_2}{\sqrt{(\pi_1^T \Omega^*_\infty \pi_1)(\pi_2^T \Omega^*_\infty \pi_2)}}$
Important Points so far …

• Def. of 2D points and lines, 3D points and planes
• Def. of homogeneous coordinates
• Def. of projective space (2D and 3D)
• Effect of transformations on points, lines, planes
• Next: Projections from 3D to 2D
Overview

• 2D Projective Geometry
• 3D Projective Geometry
• Camera Models & Calibration
Camera Model

Relation between pixels and rays in space
Pinhole Camera
Pinhole Camera
Pinhole Camera
Pinhole Camera

camera center 
$(0, 0, 0)^T$

figure adapted from Hartley and Zisserman, 2004
Pinhole Camera

Projection as matrix multiplication:

\[
\begin{pmatrix}
    x' \\
    y' \\
    z'
\end{pmatrix} =
\begin{pmatrix}
    f & 0 & 0 \\
    0 & f & 0 \\
    0 & 0 & 1
\end{pmatrix}
\begin{pmatrix}
    X \\
    Y \\
    Z
\end{pmatrix} =
\begin{pmatrix}
    fX \\
    fY \\
    fZ
\end{pmatrix} =
\begin{pmatrix}
    fX/Z \\
    fY/Z \\
    1
\end{pmatrix}
\]

De-homogenization:

\[
\begin{pmatrix}
    x \\
    y
\end{pmatrix} =
\begin{pmatrix}
    x'/z' \\
    y'/z'
\end{pmatrix}
\]
Pinhole Camera

Principal point \( \mathbf{p} = (p_x, p_y) \)

Mapping to pixel coordinates:
\[
\begin{pmatrix} x \\ y \end{pmatrix} \mapsto \begin{pmatrix} x + p_y \\ y + p_y \end{pmatrix}
\]

Projection as matrix multiplication:
\[
\begin{pmatrix} x' \\ y' \\ z' \end{pmatrix} = \begin{pmatrix} f & 0 & p_x \\ 0 & f & p_y \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} X \\ Y \\ Z \end{pmatrix}
\]
Intrinsic Camera Parameters

General intrinsic camera calibration matrix:

$$K = \begin{pmatrix} f & s & px \\ 0 & \alpha f & py \\ 0 & 0 & 1 \end{pmatrix}$$

In practice:

$$K = \begin{pmatrix} f & 0 & w/2 \\ 0 & f & h/2 \\ 0 & 0 & 1 \end{pmatrix}$$
Extrinsic Camera Parameters

Transformation from global to camera coordinates:

\[ \mathbf{X}_{\text{cam}} = R \left( \mathbf{X}_{\text{global}} - \tilde{C} \right) \]
Projection Matrix

Projection from 3D global coordinates to pixels:

\[
\lambda \begin{pmatrix} x \\ y \\ 1 \end{pmatrix} = K (R X_{\text{global}} + t)
\]

3x4 matrix
(maps from \( \mathbb{P}^3 \) to \( \mathbb{P}^2 \))

projection matrix

figure adapted from Hartley and Zisserman, 2004
Practical Camera Calibration

Method and Pictures from Zhang (ICCV’99): “Flexible Camera Calibration By Viewing a Plane From Unknown Orientations”

Unknown: constant camera intrinsics \( K \) 
(varying) camera poses \( R, t \)

Known: 3D coordinates of chessboard corners 

=> Define to be the \( z=0 \) plane \( (X=[X_1 X_2 0 1]^T) \)

Point is mapped as 
\[
\lambda x = K (r_1, r_2, r_3, t) X \\
\lambda x = K (r_1, r_2, t) [X_1 X_2 1]^T
\]

Homography \( H \) between image and chess coordinates, estimate from known \( X_i \) and measured \( x_i \)
Direct Linear Transformation (DLT)

\[
x'_i \neq Hx_i = 0
\]

\[
x'_i = (x'_i, y'_i, w'_i)^T
\]

\[
Hx_i = \begin{pmatrix}
h^1_x x_i \\
h^2_x x_i \\
h^3_x x_i
\end{pmatrix}
\]

\[
x'_i \times Hx_i = \begin{pmatrix}
y'_i h^3 x_i - w'_i h^2 x_i \\
w'_i h^1 x_i - x'_i h^3 x_i \\
x'_i h^2 x_i - y'_i h^1 x_i
\end{pmatrix}
\]

\[
\begin{bmatrix}
0^T & -w'_i x_i^T & y'_i x_i^T \\
w'_i x_i^T & 0^T & -x'_i x_i^T \\
-y'_i x_i^T & x'_i x_i^T & 0^T
\end{bmatrix}\begin{pmatrix}
h^1 \\
h^2 \\
h^3
\end{pmatrix} = 0
\]
Direct Linear Transformation (DLT)

- Equations are linear in $\mathbf{h}$: $A_i \mathbf{h} = 0$

- Only 2 out of 3 are linearly independent (2 equations per point)

$$\begin{bmatrix}
0^T & 0^T & 0^T \\
-w_i' \mathbf{x}_i^T & -w_i' \mathbf{x}_i^T & y_i' \mathbf{x}_i^T \\
w_i' \mathbf{x}_i^T & w_i' \mathbf{x}_i^T & y_i' \mathbf{x}_i^T \\
y_i' \mathbf{x}_i^T & x_i' \mathbf{x}_i^T & -x_i' \mathbf{x}_i^T
\end{bmatrix}\begin{bmatrix}
h_1^1 \\
h_2^2 \\
h_3^3
\end{bmatrix} = 0$$

(only drop third row if $w_i' \neq 0$)

- Holds for any homogeneous representation, e.g. $(x_i', y_i', 1)$
Direct Linear Transformation (DLT)

- Solving for homography $H$

\[
\begin{bmatrix}
A_1 \\
A_2 \\
A_3 \\
A_4
\end{bmatrix} \begin{bmatrix} h 
\end{bmatrix} = 0
\]

size $A$ is 8x9 (2eq.) or 12x9 (3eq.), but rank 8

- Trivial solution is $h=0_9^T$ is not interesting
- 1D null-space yields solution of interest
  pick for example the one with $\|h\| = 1$
Over-determined solution

\[
\begin{bmatrix}
A_1 \\
A_2 \\
\vdots \\
A_n
\end{bmatrix} Ah = 0
\]

No exact solution because of inexact measurement, i.e., “noise”

Find approximate solution
- Additional constraint needed to avoid 0, e.g., \( \| h \| = 1 \)
- \( Ah = 0 \) not possible, so minimize \( \| Ah \| \)
Objective

Given \( n \geq 4 \) 2D to 2D point correspondences \( \{x_i \leftrightarrow x_i'\} \), determine the 2D homography matrix \( H \) such that \( x_i' = Hx_i \).

Algorithm

(i) For each correspondence \( x_i \leftrightarrow x_i' \) compute \( A_i \). Usually only two first rows needed.
(ii) Assemble \( n \) 2x9 matrices \( A_i \) into a single 2nx9 matrix \( A \)
(iii) Obtain SVD of \( A \). Solution for \( h \) is last column of \( V \)
(iv) Determine \( H \) from \( h \)
Importance of Normalization

\[
\begin{bmatrix}
0 & 0 & 0 & -x_i' & -y_i' & -1 & y_i'x_i & y_i'y_i & y_i'' \\
x_i & y_i & 1 & 0 & 0 & 0 & -x_i'x_i & -x_i'y_i & -x_i''
\end{bmatrix}
\begin{pmatrix}
h^1 \\
h^2 \\
h^3
\end{pmatrix} = 0
\]

\[\sim 10^2 \sim 10^2 \ 1 \sim 10^2 \sim 10^2 \ 1 \sim 10^4 \sim 10^4 \sim 10^2\]

orders of magnitude difference!

Monte Carlo simulation for identity computation based on 5 points
(not normalized ↔ normalized)
Normalized DLT Algorithm

Objective
Given $n \geq 4$ 2D to 2D point correspondences $\{x_i \leftrightarrow x'_i\}$, determine the 2D homography matrix $H$ such that $x'_i = Hx_i$

Algorithm
(i) Normalize points $\tilde{x}_i = T_{\text{norm}}x_i$, $\tilde{x}'_i = T'_{\text{norm}}x'_i$
(ii) Apply DLT algorithm to $\tilde{x}_i \leftrightarrow \tilde{x}'_i$
(iii) Denormalize solution $H = T'^{-1}_{\text{norm}}\tilde{H}T_{\text{norm}}$

Normalization (independently per image):
• Translate points such that centroid is at origin
• Isotropic scaling such that mean distance to origin is $\sqrt{2}$
Geometric Distance

\( \mathbf{X} \) measured coordinates
\( \hat{\mathbf{X}} \) estimated coordinates
\( \overline{\mathbf{X}} \) true coordinates

\( d(.,.) \) Euclidean distance (in image)

Error in one image

\[
\hat{\mathbf{H}} = \arg\min_{\mathbf{H}} \sum_i d(x'_i, \mathbf{H}\overline{x}_i)^2 \quad \text{e.g. calibration pattern}
\]

Symmetric transfer error

\[
\hat{\mathbf{H}} = \arg\min_{\mathbf{H}} \sum_i d(x_i, \mathbf{H}^{-1}x'_i)^2 + d(x'_i, \mathbf{H}x_i)^2
\]

Reprojection error

\[
(\hat{\mathbf{H}}, \hat{x}_i, \hat{x}'_i) = \arg\min_{\hat{\mathbf{H}}, \hat{x}_i, \hat{x}'_i} \sum_i d(x_i, \hat{x}_i)^2 + d(x'_i, \hat{x}'_i)^2
\]

subject to \( \hat{x}_i = \hat{\mathbf{H}}\hat{x}'_i \)
Reprojection Error

\[ d(x, H^{-1}x')^2 + d(x', Hx)^2 \]

\[ d(x, \hat{x})^2 + d(x', \hat{x}')^2 \]
Statistical Cost Function and Maximum Likelihood Estimation

- Optimal cost function related to noise model
- Assume zero-mean isotropic Gaussian noise (assume outliers removed)

\[
Pr(x) = \frac{1}{2\pi\sigma^2} e^{-d(x,\bar{x})^2/(2\sigma^2)}
\]

Error in one image

\[
Pr\left(\{x'_i\} | H\right) = \prod_i \frac{1}{2\pi\sigma^2} e^{-d(x'_i,H\bar{x}_i)^2/(2\sigma^2)}
\]

\[
\log Pr\left(\{x'_i\} | H\right) = -\frac{1}{2\sigma^2} \sum_i d(x'_i,H\bar{x}_i)^2 + \text{const}
\]

Maximum Likelihood Estimate:

\[
\min \sum d(x'_i,H\bar{x}_i)^2
\]
Gold Standard Algorithm

Objective
Given $n \geq 4$ 2D to 2D point correspondences \( \{x_i \leftrightarrow x'_i\} \), determine the Maximum Likelihood Estimation of $H$ (this also implies computing optimal $x'_i = Hx_i$)

Algorithm
(i) **Initialization**: compute an initial estimate using normalized DLT or RANSAC

(ii) **Geometric minimization of symmetric transfer error**:
    - Minimize using Levenberg-Marquardt over 9 entries of $h$
    - Compute initial estimate for optimal $\{x_i\}$
    - Minimize cost $\sum d(x_i, \hat{x}_i)^2 + d(x'_i, \hat{x}'_i)^2$ over $\{H, x_1, x_2, \ldots, x_n\}$
    - If many points, use sparse method
Radial Distortion

Due to spherical lenses (cheap)

(One possible) model:

\[
\begin{bmatrix}
  x \\
  y \\
  1
\end{bmatrix}
\sim
\begin{bmatrix}
  f_x & s & c_x \\
  0 & f_y & c_y \\
  0 & 0 & 1
\end{bmatrix}
R
\begin{bmatrix}
  1 & 0 & 0 & 0 \\
  0 & 1 & 0 & 0 \\
  0 & 0 & 1 & 0 \\
\end{bmatrix}
\begin{bmatrix}
  R^T \\
  -R^Tt \\
  0 \\
  1
\end{bmatrix}
\begin{bmatrix}
  X \\
  Y \\
  Z \\
  1
\end{bmatrix}
\]

\[R: \quad (x, y) = (1 + K_1(x^2 + y^2) + K_2(x^2 + y^2)^2 + ...) \begin{bmatrix}
  x \\
  y
\end{bmatrix}\]
Calibration with Radial Distortion

- Low radial distortion:
  - Ignore radial distortion during initial calibration
  - Estimate distortion parameters, refine full calibration

- High radial distortion: Simultaneous estimation
  - Kukelova et al., “Real-Time Solution to the Absolute Pose Problem with Unknown Radial Distortion and Focal Length”, ICCV 2013
Bouguet Toolbox

Camera Calibration Toolbox for Matlab

http://www.vision.caltech.edu/bouguetj/calib_doc/
Rolling Shutter Cameras

- Image build row by row
- Distortions based on depth and speed
- Many mobile phone cameras have rolling shutter

Video credit: Olivier Saurer
Rolling Shutter Effect

Global shutter  Rolling shutter

Slide credit: Cenek Albl
Event-based, 6-DOF Pose Tracking for High-Speed Maneuvers

Elias Mueggler, Basil Huber and Davide Scaramuzza

ROBOTICS & PERCEPTION GROUP
rpg.ifi.uzh.ch

University of Zurich UZH
Department of Informatics

ETH Zürich
<table>
<thead>
<tr>
<th>Date</th>
<th>Topic</th>
</tr>
</thead>
<tbody>
<tr>
<td>Feb 18</td>
<td>Introduction</td>
</tr>
<tr>
<td><strong>Feb 25</strong></td>
<td><strong>Geometry, Camera Model, Calibration</strong></td>
</tr>
<tr>
<td>Mar 4</td>
<td>Features, Tracking / Matching</td>
</tr>
<tr>
<td>Mar 11</td>
<td><strong>Project Proposals by Students</strong></td>
</tr>
<tr>
<td>Mar 18</td>
<td>Structure from Motion (SfM) + papers</td>
</tr>
<tr>
<td>Mar 25</td>
<td>Dense Correspondence (stereo / optical flow) + papers</td>
</tr>
<tr>
<td>Apr 1</td>
<td>Bundle Adjustment &amp; SLAM + papers</td>
</tr>
<tr>
<td><strong>Apr 8</strong></td>
<td><strong>Student Midterm Presentations</strong></td>
</tr>
<tr>
<td>Apr 15</td>
<td>Multi-View Stereo &amp; Volumetric Modeling + papers</td>
</tr>
<tr>
<td>Apr 22</td>
<td>Easter break</td>
</tr>
<tr>
<td>Apr 29</td>
<td>3D Modeling with Depth Sensors + papers</td>
</tr>
<tr>
<td>May 6</td>
<td>3D Scene Understanding + papers</td>
</tr>
<tr>
<td>May 13</td>
<td>4D Video &amp; Dynamic Scenes + papers</td>
</tr>
<tr>
<td>May 20</td>
<td>papers</td>
</tr>
<tr>
<td><strong>May 27</strong></td>
<td><strong>Student Project Demo Day = Final Presentations</strong></td>
</tr>
</tbody>
</table>

- **ETH**
Reminder

- Project presentation in 2 weeks
- Form team & decide project topic
  - By March 1\textsuperscript{st}
- Talk with supervisor, submit proposal
  - By March 8th
Next class:
Features, Tracking / Matching