Projective Structure from Motion
(Uncalibrated Perspective Cameras)

EECS 598-08 Fall 2014
Instructor: Jason Corso (jjcorso)
Foundations of Computer Vision
web.eecs.umich.edu/~jjcorso/t/598F14

Readings:  FP 8.3
Date:      11/5/14

Materials on these slides have come from many sources in addition to myself (primarily Silvio Savarese to whom I am ultimately grateful); individual slides reference specific sources.
Structure from motion problem

Given $m$ images of $n$ fixed 3D points

\[ x_{ij} = M_i X_j, \quad i = 1, \ldots, m, \quad j = 1, \ldots, n \]

Courtesy of Silvio Savarese.
Structure from motion problem

From the m x n correspondences $x_{ij}$, estimate:

- $m$ projection matrices $M_i$
- $n$ 3D points $X_j$
- **Position ambiguity:** it is impossible based on the images alone to estimate the absolute location and pose of the scene w.r.t. a 3D world coordinate frame.
The calibration matrix has not changed!

Courtesy of Silvio Savarese.
In the general case (nothing is known) the ambiguity is expressed by an arbitrary affine or projective transformation.

\[
x_j = M_i X_j
\]

\[
H X_j
\]

\[
M_i H^{-1}
\]

\[
x_j = M_i X_j = (M_i H^{-1})(H X_j)
\]

Courtesy of Silvio Savarese.
Projective ambiguity

Courtesy of Silvio Savarese.
Affine ambiguity

Courtesy of Silvio Savarese.
Structure from motion ambiguity

• The ambiguity exists even for calibrated cameras

• For calibrated cameras, the similarity ambiguity is the only ambiguity

[Longuet-Higgins '81]

- The scene is determined by the images only up a similarity transformation (rotation, translation and scaling)
Structure from motion ambiguity

-Scale ambiguity: it is impossible based on the images alone to estimate the absolute scale of the scene (i.e. house height)

Courtesy of Silvio Savarese.
Structure from motion problem

Given $m$ images of $n$ fixed 3D points

\[ \mathbf{x}_{ij} = \mathbf{M}_i \mathbf{X}_j, \quad i = 1, \ldots, m, \quad j = 1, \ldots, n \]

Courtesy of Silvio Savarese.
Structure from motion problem

$m$ cameras $M_1 \ldots M_m$

\[
M_i = \begin{bmatrix}
a_{11} & a_{12} & a_{13} & b_1 \\
a_{21} & a_{22} & a_{23} & b_2 \\
a_{31} & a_{32} & a_{33} & 1 
\end{bmatrix}
\]
The Projective Structure-from-Motion Problem

Given $m$ images of $n$ fixed points $X_j$ we can write

$$x_{ij} = M_i X_j \quad \text{for } i = 1, \ldots, m \quad \text{and} \quad j = 1, \ldots, n.$$ 

**Problem:** estimate the $m$ $3\times4$ matrices $M_i$ and the $n$ positions $X_j$ from the $m \times n$ correspondences $x_{ij}$.

- With no calibration info, cameras and points can only be recovered up to a $4 \times 4$ projective (15 parameters)
- Given two cameras, how many points are needed?
- How many equations and how many unknown?

$2m \times n$ equations in $11m+3n - 15$ unknowns

So 7 points! $[2 \times 2 \times 7 = 28; \quad 11 \times 2 + 3 \times 7 - 15 = 28]$
Structure-from-Motion Algorithms

- Algebraic approach (by fundamental matrix)
- Factorization method (by SVD)
- Bundle adjustment
Algebraic approach (2-view case)

- Compute the fundamental matrix $F$ from two views (e.g. 8 point algorithm)
- Use $F$ to estimate projective cameras
- Use these cameras to triangulate and estimate points in 3D
Algebraic approach (2-view case)

Apply a projective transformation $H$ such that:

$$M_1 H^{-1} = \begin{bmatrix} I & 0 \end{bmatrix} \quad M_2 H^{-1} = \begin{bmatrix} A & b \end{bmatrix}$$

Canonical perspective cameras

Courtesy of Silvio Savarese.
Algebraic approach (Fundamental matrix)

\[ \tilde{X} = H \ X \]

\[ x = M_1 \ H^{-1} \ \tilde{X} = [I \ | \ 0] \tilde{X} \quad \quad x' = M_2 \ H^{-1} \ \tilde{X} = [A \ | \ b] \tilde{X} \]

\[ x' = [A \ | \ b] \tilde{X} = [A \ | \ b] \begin{bmatrix} \tilde{X}_1 \\ \tilde{X}_2 \\ \tilde{X}_3 \\ 1 \end{bmatrix} = A[1 \ | \ 0] \begin{bmatrix} \tilde{X}_1 \\ \tilde{X}_2 \\ \tilde{X}_3 \\ 1 \end{bmatrix} + b = A[1 \ | \ 0] \tilde{X} + b = Ax + b \]

\[ x' \times b = (Ax + b) \times b = Ax \times b \]

\[ (Ax \times b) \cdot x' = (x' \times b) \cdot x' = 0 \]

\[ x'^T (Ax \times b)^T = 0 \]

\[ x'^T [b_x] A x = 0 \quad \text{is this familiar?} \]

\[ F = [b_x] A \]

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

Courtesy of Silvio Savarese.
Cross product as matrix multiplication

\[
a \times b = \begin{bmatrix}
0 & -a_z & a_y \\
-\alpha_z & 0 & -\alpha_x \\
-\alpha_y & \alpha_x & 0 \\
\end{bmatrix} \begin{bmatrix}
b_x \\
b_y \\
b_z \\
\end{bmatrix} = [a_x]b
\]
Algebraic approach (Fundamental matrix)

\[ x'^T F x = 0 \quad F = [b_x]A \]

- Compute the fundamental matrix F from two views (eg. 8 point algorithm)

Can verify that:

\[ F \cdot b = [b_x]A \cdot b = 0 \quad \text{Compute } b \text{ as least sq. solution of } F b = 0 \]

\[ A = [b_x]^{-1} F = -[b_x] F \]

Notice that b is an epipole
Epipolar Constraint [from earlier lecture]

- \( F x_2 \) is the epipolar line associated with \( x_2 \) (\( l_1 = F x_2 \))
- \( F^T x_1 \) is the epipolar line associated with \( x_1 \) (\( l_2 = F^T x_1 \))
- \( F \) is singular (rank two)
- \( F e_2 = 0 \) and \( F^T e_1 = 0 \)
- \( F \) is 3x3 matrix; 7 DOF

Courtesy of Silvio Savarese.
Algebraic approach (Fundamental matrix)

\[ x'^T F x = 0 \quad \quad F = [b_x]A \]

- Compute the fundamental matrix \( F \) from two views (eg. 8 point algorithm)

Can verify that:

\[ F \cdot b = [b_x]A \cdot b = 0 \quad \rightarrow \quad \text{Compute } b \text{ as least sq. solution of } F \cdot b = 0 \]

\[ \text{det}(F) = 0; \quad |b| = 1 \]

\[ A = [b_x]^{-1} F = -[b_x] F \]

Notice that \( b \) is an epipole

\[ M^p_1 = \begin{bmatrix} 1 & 0 \end{bmatrix} \quad \quad M^p_2 = \begin{bmatrix} -[e_x]F & e \end{bmatrix} \]

Perspective cameras are known

HZ, page 254
PF, page 288

Courtesy of Silvio Savarese.
Structure-from-Motion Algorithms

- Algebraic approach (by fundamental matrix)
- Factorization method (by SVD)
- Bundle adjustment
Projective factorization

\[
D = \begin{bmatrix}
Z_{11}x_{11} & Z_{12}x_{12} & \cdots & Z_{1n}x_{1n} \\
Z_{21}x_{21} & Z_{22}x_{22} & \cdots & Z_{2n}x_{2n} \\
& & \ddots & \\
Z_{m1}x_{m1} & Z_{m2}x_{m2} & \cdots & Z_{mn}x_{mn}
\end{bmatrix}
= \begin{bmatrix}
M_1 \\
M_2 \\
\vdots \\
M_m
\end{bmatrix}
\begin{bmatrix}
x_1 & x_2 & \cdots & x_n
\end{bmatrix}
\]

points (4 \times n)
cameras (3m \times 4)

\[D = MS \text{ has rank } 4\]
A factorization method - (affine case; last lecture)

- Let’s create a $2m \times n$ data (measurement) matrix:

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

(2m × n)

The measurement matrix $D = MS$ has rank 3

(it’s a product of a 2mx3 matrix and 3xn matrix)

Courtesty of Silvio Savarese.
Projective factorization

\[
D = \begin{bmatrix}
Z_{11}x_{11} & Z_{12}x_{12} & \cdots & Z_{1n}x_{1n} \\
Z_{21}x_{21} & Z_{22}x_{22} & \cdots & Z_{2n}x_{2n} \\
\vdots & \vdots & \ddots & \vdots \\
Z_{m1}x_{m1} & Z_{m2}x_{m2} & \cdots & Z_{mn}x_{mn}
\end{bmatrix}
= \begin{bmatrix}
M_1 \\
M_2 \\
\vdots \\
M_m
\end{bmatrix}
\begin{bmatrix}
x_1 \\
x_2 \\
\vdots \\
x_n
\end{bmatrix}
\]

\[D = MS\] has rank 4

- If we knew the depths \(z\), we could factorize \(D\) to estimate \(M\) and \(S\)
- If we knew \(M\) and \(S\), we could solve for \(z\)
- Solution: iterative approach (alternate between above two steps)

Courtesy of Silvio Savarese.
Structure-from-Motion Algorithms

- Algebraic approach (by fundamental matrix)
- Factorization method (by SVD)
- Bundle adjustment
Bundle adjustment

- Non-linear method for refining structure and motion
- Minimizing re-projection error

\[
E(M, X) = \sum_{i=1}^{m} \sum_{j=1}^{n} D(x_{ij}, M_i X_j)^2
\]
Bundle adjustment

- Non-linear method for refining structure and motion
- Minimizing re-projection error

\[ E(M, X) = \sum_{i=1}^{m} \sum_{j=1}^{n} D(x_{ij}, M_i X_j)^2 \]

• Advantages
  - Handle large number of views
  - Handle missing data

• Limitations
  - Large minimization problem (parameters grow with number of views)
  - Requires good initial condition

Used as the final step of SFM
Removing the ambiguities:
the Stratified reconstruction

• up grade reconstruction from perspective to affine
  [by measuring the plane at infinity]

• up grade reconstruction from affine to metric
  [by measuring the absolute conic]

Recovering the metric reconstruction
from the perspective one is called self-calibration

Courtesy of Silvio Savarese.
Self-calibration

Process of determining intrinsic camera parameters directly from un-calibrated images

Suppose we have a projective reconstruction \{M_i, X_j\}

**GOAL:** find a rectifying (non-singular) homography H such that \{M_i H, H^{-1}X_j\} is a metric reconstruction

\[
\begin{align*}
\overline{M}_i & = M_i H \\ i & = 1 \cdots m \\ \overline{M}_i & = K_i [R_i \quad T_i]
\end{align*}
\]

If world ref. system = camera 1 ref. system:

\[
\overline{M}_1 = K_1 [I \quad 0]
\]

If the perspective camera is canonical:

\[
M_1 = [I \quad 0]
\]
Self-calibration

\[
\bar{M}_i = M_i H
\]

\[
[K_1 \ 0] = [I \ 0] H
\]

\[
A = K_1
\]
\[
t = 0
\]

We can set \(k=1\)
(this fixes the scale of the reconstruction)

\[
H = \begin{bmatrix} K_1 & 0 \\ V & 1 \end{bmatrix}
\]

Planes at infinity

\[
p_\infty = H^{-1} \Pi_\infty = \begin{bmatrix} p \\ 1 \end{bmatrix}
\]

See appendix

\[
H = \begin{bmatrix} K_1 & 0 \\ -p^T K_1 & 1 \end{bmatrix}
\]

Courtesy of Silvio Savarese.
Planes at infinity (lecture 5)

\[ \Pi_\infty = \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} \]

In the metric (Euclidean) world coordinates

2 planes are parallel iff their intersections is a line that belongs to \( \Pi_\infty \)

The projective transformation of a plane at infinity can be expressed as

\[ \Pi_\infty = H^{-1} \Pi_\infty = \begin{bmatrix} p \\ 1 \end{bmatrix} \]
Self-calibration

\[
\mathbf{\pi}_\infty = \begin{bmatrix} p \\ 1 \end{bmatrix} = H^{-1} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix}
\]

\[
H = \begin{bmatrix} K_1 & 0 \\ v & 1 \end{bmatrix}
\]

\[
= \begin{bmatrix} K_1^{-T} & -K_1^{-T}v \\ 0 & 1 \end{bmatrix} \begin{bmatrix} 0 \\ 0 \\ 0 \\ 1 \end{bmatrix} = \begin{bmatrix} -K_1^{-T}v \\ 1 \end{bmatrix}
\]

\[
v = -p^T K_1
\]

Courtesy of Silvio Savarese.
Self-calibration

**GOAL:** find a rectifying homography \( H \) such that

\[
\{M_i, X_j\} \rightarrow \{M_i \ H, H^{-1}X_j\} \text{ is a metric reconstruction}
\]

\[
H = \begin{bmatrix} K_1 & 0 \\ -p^T & 1 \end{bmatrix}
\]

\[
K = \begin{bmatrix} \alpha & -\alpha \cot \theta & u_o \\ 0 & \beta & v_o \\ 0 & \frac{\sin \theta}{\alpha} & 1 \end{bmatrix}
\]

\( K_1 = \) calibration matrix of first camera

\( \pi_\infty = [p \ 1]^T \) = plane at infinity in the projective reconstruction

5 unknowns

3 unknowns

Courtesy of Silvio Savarese.
Self-calibration basic equation

\[
\begin{align*}
\overline{M}_i &= [A_i \ a_i] &= \text{perspective reconstruction of the camera (known)} \\
\overline{M}_i &= K_i[R_i \ T_i] &= \text{metric reconstruction of the camera (unknown)} \\
H &= \begin{bmatrix} K_1 & 0 \\ -p^T K_1 & 1 \end{bmatrix} &= \text{rectifying homography (unknown)} \\
\overline{M}_i &= M_i H & i = 2 \cdots m
\end{align*}
\]

\[
\begin{bmatrix} K_i & R_i \\ T_i \end{bmatrix} = \begin{bmatrix} A_i & a_i \end{bmatrix} \begin{bmatrix} K_1 & 0 \\ -p^T K_1 & 1 \end{bmatrix} = \begin{bmatrix} A_i \ K_1 - a_i p^T K_1 & a_i \end{bmatrix}
\]

\[
K_i \ R_i = (A_i - a_i p^T)K_1 \quad \rightarrow \quad R_i = K_i^{-1}(A_i - a_i p^T)K_1
\]

Courtesy of Silvio Savarese.
Self-calibration basic equation

\[
\begin{align*}
R_i &= K_i^{-1} (A_i - a_i p^T) K_1 \\
R_i^T &= K_1^T (A_i - a_i p^T)^T K_i^{-T} \\
R_i R_i^T &= I \\
K_i^{-1} (A_i - a_i p^T) K_1 K_1^T (A_i - a_i p^T)^T K_i^{-T} &= I \\
(A_i - a_i p^T) K_1 K_1^T (A_i - a_i p^T)^T &= K_i K_i^T
\end{align*}
\]

Courtesy of Silvio Savarese.
Absolute conic $\Omega_\infty$ is a $\mathbf{C} \in \mathbf{\Pi}_\infty$

Any $\mathbf{x} \in \Omega_\infty$ satisfies:

$\mathbf{x}^T \Omega_\infty \mathbf{x} = 0$

$\Omega_\infty = \begin{bmatrix} 1 & 1 \\ 1 & 0 \end{bmatrix} \begin{cases} x_1^2 + x_2^2 + x_3^2 = 0 \\ x_4 = 0 \end{cases}$

Projective transformation of $\Omega_\infty$

$\omega = (K^T K)^{-1}$

$\omega^* = K K^T$

Dual image of the absolute conic

Courtesy of Silvio Savarese.
Properties of $\omega$

$$\omega = (K^T K)^{-1}$$

- It is not function of $R$, $T$

- Symmetric (5 unknowns)

$$\omega = \begin{bmatrix}
\omega_1 & \omega_2 & \omega_4 \\
\omega_2 & \omega_3 & \omega_5 \\
\omega_4 & \omega_5 & \omega_6
\end{bmatrix}$$

Courtesy of Silvio Savarese.
Self-calibration basic equation

\[
\left(A_i - a_i p^T\right) K_1 K_1^T \left(A_i - a_i p^T\right)^T = K_i \ K_i^T
\]

\[
\left(A_i - a_i p^T\right) \omega_i^* \left(A_i - a_i p^T\right)^T = \omega_i^* \quad \text{i=2...m}
\]

[A_i and a_i are known]

How many unknowns?

- 3 from \( p \)
- 5 from \( \omega_i \) [per view]

How many equations?

5 independent equations [per view]

Art of self-calibration:

use constraints on \( \omega \) (K) to generate enough equations on the unknowns

Courtesy of Silvio Savarese.
Self-calibration – identical Ks

\[
(A_i - a_i p^T) \omega^* (A_i - a_i p^T)^T = \omega^*
\]

• For m views, 5(m-1) constraints
• Number of unknowns: 8

m ≥ 3 provides enough constraints

To solve the self-calibration problem with identical cameras we need at least 3 views
Properties of $\omega$

$$\omega = (K^T K)^{-1}$$

1. $\omega = \begin{bmatrix} \omega_1 & \omega_2 & \omega_4 \\ \omega_2 & \omega_3 & \omega_5 \\ \omega_4 & \omega_5 & \omega_6 \end{bmatrix}$

2. $\omega_2 = 0$ zero-skew

3. $\omega_2 = 0$
   $\omega_1 = \omega_3$

   square pixel

4. $\omega_4 = \omega_5 = 0$

   zero-offset

Courtesy of Silvio Savarese.
Self-calibration – other constraints

\[
(A_i - a_ip^T) \omega_i^* (A_i - a_ip^T)^T = \omega_i^*
\]

- zero-offset \( \omega_4 = \omega_5 = 0 \) \( \rightarrow \) 2 m linear constraints
- zero-skew \( \omega_2 = 0 \) \( \rightarrow \) m linear constraints

etc…

Courtesy of Silvio Savarese.
## Self-calibration - summary

<table>
<thead>
<tr>
<th>Condition</th>
<th>N. Views</th>
</tr>
</thead>
<tbody>
<tr>
<td>• Constant internal parameters</td>
<td>3</td>
</tr>
<tr>
<td>• Aspect ratio and skew known</td>
<td>4</td>
</tr>
<tr>
<td>• Focal length and offset vary</td>
<td></td>
</tr>
<tr>
<td>• Aspect ratio and skew constant</td>
<td>5</td>
</tr>
<tr>
<td>• Focal length and offset vary</td>
<td></td>
</tr>
<tr>
<td>• skew = 0, all other parameters vary</td>
<td>8</td>
</tr>
</tbody>
</table>

Issue: the larger is the number of view, the harder is the correspondence problem.

Bundle adjustment helps!

Courtesy of Silvio Savarese.
Self-calibration - summary

Constraints on camera motion can be incorporated

- Linearly translating camera

- Single axis of rotation: turntable motion

Courtesy of Silvio Savarese.
SFM problem - summary

1. Estimate structure and motion up perspective transformation
   1. Algebraic
   2. factorization method
   3. bundle adjustment

2. Convert from perspective to metric (self-calibration)

3. Bundle adjustment

** or **

1. Bundle adjustment with self-calibration constraints
Correspondences

• Can refine feature matching after a structure and motion estimate has been produced

  – decide which ones obey the *epipolar geometry*

  – decide which ones are *geometrically consistent*

  – (optional) iterate between correspondences and SfM estimates using MCMC

[Dellaert et al., Machine Learning 2003]
SFM Summed Up...

From: http://www.youtube.com/watch?v=i7ierVkJYa8
Applications

Courtesy of Oxford Visual Geometry Group
Applications

D. Nistér, PhD thesis ‘01

Courtesy of Silvio Savarese.
Applications

M. Pollefeys et al 98---

Courtesy of Silvio Savarese.
Applications


Courtesy of Silvio Savarese.
Photo synth

Incremental reconstruction of construction sites
Initial pair – 2168 & Complete Set 62,323 points, 160 images
The registration of images (08.27.08) within the reconstructed scene!

Student Dining and Residence Hall project in Champaign, IL. Images courtesy of Turner Construction.
The results of automated progress detection

Student Dining Construction Project photolog (286 image)

Courtesy of Silvio Savarese.
Non-rigid SFM...an example

Nonrigid Structure from Motion in Trajectory Space

NIPS 2008

Ijaz Akhter\textsuperscript{1}  Yaser Sheikh\textsuperscript{2}  Sohaib Khan\textsuperscript{1}  Takeo Kanade\textsuperscript{2}

\textsuperscript{1}LUMS School of Science and Engineering, Lahore, Pakistan
\textsuperscript{2}Carnegie Mellon University, Pittsburgh, PA, USA

http://cvlab.lums.edu.pk/nrsfm
Next Lecture: Introduction to Visual Recognition

• Readings: FP 15.1, 18.1