Intro to 3D + Camera Calibration

EECS 442 – David Fouhey
Fall 2019, University of Michigan

http://web.eecs.umich.edu/~fouhey/teaching/EECS442_F19/
Our goal: Recovery of 3D structure

J. Vermeer, *Music Lesson*, 1662

Next few classes

• First: some intuitions and examples from biological vision about 3D perception
• But first, a brief review
Let’s Take a Picture!

Slide inspired by S. Seitz; image from Michigan Engineering
Projection Matrix

Projection \((fx/z, fy/z)\) is matrix multiplication

\[
\begin{bmatrix}
fx \\
fy \\
z
\end{bmatrix} \equiv \begin{bmatrix}
f & 0 & 0 & 0 \\
0 & f & 0 & 0 \\
0 & 0 & 1 & 0
\end{bmatrix} \begin{bmatrix}
x \\
y \\
z
\end{bmatrix} \rightarrow \begin{bmatrix}
fx/z \\
fy/z
\end{bmatrix}
\]

Slide inspired from L. Lazebnik
Given a \textit{calibrated camera} and an image, we only know the ray corresponding to each pixel.

Nowhere near enough constraints for $X$. 

Diagram credit: S. Lazebnik
Single-view Ambiguity

http://en.wikipedia.org/wiki/Ames_room

Slide Credit: J. Hays
Single-view Ambiguity

Diagram credit: J. Hays
Single-view Ambiguity

Rashad Alakbarov shadow sculptures
Resolving Single-view Ambiguity

- Shoot light (lasers etc.) out of your eyes!
- Con: not so biologically plausible, dangerous?
Resolving Single-view Ambiguity

- Shoot light (lasers etc.) out of your eyes!
- Con: not so biologically plausible, dangerous?
Resolving Single-view Ambiguity

- Stereo: given 2 calibrated cameras in different views and correspondences, can solve for X

Original diagram credit: S. Lazebnik
Human stereopsis: disparity

From Bruce and Green, Visual Perception, Physiology, Psychology and Ecology

Human eyes **fixate** on point in space – rotate so that corresponding images form in centers of fovea.
Human stereopsis: disparity

Disparity occurs when eyes fixate on one object; others appear at different visual angles.

From Bruce and Green, Visual Perception, Physiology, Psychology and Ecology

Adapted from David Forsyth, UC Berkeley
Stereo photography and stereo viewers

Take two pictures of the same subject from two slightly different viewpoints and display so that each eye sees only one of the images.

Invented by Sir Charles Wheatstone, 1838

Image from fisher-price.com

Slide credit: J. Hays
http://www.well.com/~jimg/stereo/stereo_list.html

Slide credit: J. Hays
Autostereograms

Exploit disparity as depth cue using single image.

(Single image random dot stereogram, Single image stereogram)

Slide credit: J. Hays, Images from magiceye.com
Autostereograms

Slide credit: J. Hays, Images from magiceye.com
Yeah, yeah, but...

Not all animals see stereo:
Prey animals (large field of view to spot predators)
Stereoblind people
Resolving Single-view Ambiguity

• One option: move, find correspondence.
• If you know how you moved and have a calibrated camera, can solve for $X$
Knowing $R, t$

- How do you know how far you moved?
- Can solve via vision
- Can solve via ears
- **Why does your inner ear have 3 ducts?**
- Can solve via signals sent to muscles
Yeah, yeah, but…

You haven’t been here before, yet you probably have a fairly good understanding of this scene.
Pictorial Cues – Shading

[Figure from Prados & Faugeras 2006]
Pictorial Cues – Texture

Pictorial Cues – Perspective effects

Image credit: S. Seitz
Pictorial Cues – Familiar Objects

Monitor: probably not 12 feet wide.

Desk surface: probably flat
Reality of 3D Perception

• 3D perception is absurdly complex and involves integration of many cues:
  • Learned cues for 3D
  • Stereo between eyes
  • Stereo via motion
  • Integration of known motion signals to muscles (efferent copy), acceleration sensed via ears
  • Past experience of touching objects
• All connect: learned cues from 3D probably come from stereo/motion cues in large part

Really fantastic article on cues for 3D from Cutting and Vishton, 1995: [https://pmvish.people.wm.edu/cutting%26vishton1995.pdf](https://pmvish.people.wm.edu/cutting%26vishton1995.pdf)
How are Cues Combined?

Ames illusion persists (in a weaker form) even if you have stereo vision—gussing the texture is rectilinear is usually incredibly reliable.

More Formally
Multi-view geometry problems

**Calibration:**
We need camera intrinsics / K in order to figure out where the rays are
Multi-view geometry problems

Recovering structure:
Given cameras and correspondences, find 3D.

Camera 1 \(R_1, t_1\)
Camera 2 \(R_2, t_2\)
Camera 3 \(R_3, t_3\)
Multi-view geometry problems

Stereo/Epipolar Geometry: Given 2 cameras and find where a point could be
Multi-view geometry problems

Motion:
Figure out $R$, $t$ for a set of cameras given correspondences.
Outline

• (Today) Calibration:
  • Getting intrinsic matrix/K

• Single view geometry:
  • measurements with 1 image

• Stereo/Epipolar geometry:
  • 2 pictures $\rightarrow$ depthmap

• Structure from motion (SfM):
  • 2+ pictures $\rightarrow$ cameras, pointcloud
Typical Perspective Model

\[ p = \begin{bmatrix} f & 0 & u_0 \\ 0 & f & v_0 \\ 0 & 0 & 1 \end{bmatrix} \]

- **focal length**
- **principal point (image coords of camera origin on retina)**
- **rotation**
- **translation**
- **2D Projection of X**
- **3D point**

\[ \begin{bmatrix} R_{3x3} & t_{3x1} \end{bmatrix} X_{4x1} \]
Camera Calibration

\[ p \equiv \begin{bmatrix} f & 0 & u_0 \\ 0 & f & v_0 \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} R_{3 \times 3} & t_{3 \times 1} \end{bmatrix} \begin{bmatrix} X_{4 \times 1} \end{bmatrix} \]

\[ \begin{bmatrix} u \\ v \\ 1 \end{bmatrix} \equiv M_{3 \times 4} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix} \]

If I can get pairs of \([X,Y,Z]\) and \([u,v]\) → equations to constrain \(M\)
How do I get \([X,Y,Z]\), \([u,v]\)
Camera Calibration

A funny object with multiple planes.
Camera Calibration Targets

Using a tape measure

Known 2d image coords

880 214
43 203
270 197
886 347
745 302
943 128
476 590
419 214
317 335
783 521
235 427
665 429
655 362
427 333
412 415
746 351
434 415
525 234
716 308
602 187

Known 3d locations

312.747 309.140 30.086
305.796 311.649 30.356
307.694 312.358 30.418
310.149 307.186 29.298
311.937 310.105 29.216
310.106 310.572 30.682
307.106 306.876 28.660
309.317 312.490 30.230
307.435 310.151 29.318
308.253 306.300 28.881
306.650 309.301 28.905
308.069 306.831 29.189
309.671 308.834 29.029
308.255 309.955 29.267
307.546 308.613 28.963
311.036 309.206 28.913
307.518 308.175 29.069
309.950 311.262 29.990
312.160 310.772 29.080
311.988 312.709 30.514

Image credit: J. Hays
Camera Calibration Targets

A set of views of a plane (not covered today)
Camera Calibration Targets

A single, huge plane. What’s this for?
Camera calibration

- Given $n$ points with known 3D coordinates $X_i$ and known image projections $p_i$, estimate the camera parameters.
Camera Calibration: Linear Method

\[ p_i \equiv MX_i \]

Remember (from geometry): this implies \( MX_i p_i \) are scaled copies of each other

\[ p_i = \lambda MX_i, \lambda \neq 0 \]

Remember (from homography fitting): this implies their cross product is 0

\[ p_i \times MX_i = 0 \]
Camera Calibration: Linear Method

\[ p_i \times MX_i = 0 \]

\[
\begin{bmatrix}
  u_i \\
  v_i \\
  1
\end{bmatrix} \times \begin{bmatrix}
  M_1 X_i \\
  M_2 X_i \\
  M_3 X_i
\end{bmatrix} = \begin{bmatrix}
  0 \\
  0 \\
  0
\end{bmatrix}
\]

...Some tedious math occurs...

(see Homography derivation)

\[
\begin{bmatrix}
  0^T \\
  X_i^T \\
  -v_iX_i^T
\end{bmatrix} \begin{bmatrix}
  -X_i^T \\
  0^T \\
  u_iX_i^T
\end{bmatrix} \begin{bmatrix}
  M_1^T \\
  M_2^T \\
  M_3^T
\end{bmatrix} = \begin{bmatrix}
  0 \\
  0 \\
  0
\end{bmatrix}
\]
Camera Calibration: Linear Method

\[
\begin{bmatrix}
0^T & -X_i^T & \nu_iX_i^T \\
X_i^T & 0^T & -u_iX_i^T \\
-\nu_iX_i^T & u_iX_i^T & 0^T
\end{bmatrix}
\begin{bmatrix}
M_1^T \\
M_2^T \\
M_3^T
\end{bmatrix}
= \begin{bmatrix}
0 \\
0 \\
0
\end{bmatrix}
\]

How many linearly independent equations? 2

How many equations per \([u,v] + [X,Y,Z]\) pair? 2

If \(M\) is 3x4, how many degrees of freedom? 11
Camera Calibration: Linear Method

\[
\begin{bmatrix}
{0^T} & {X_i^T} & {-v_1X_i^T} \\
{X_1^T} & {0^T} & {-u_1X_i^T} \\
{\ldots} & {\ldots} & {\ldots} \\
{0^T} & {X_n^T} & {-v_1X_n^T} \\
{X_n^T} & {0^T} & {-u_nX_n^T}
\end{bmatrix}
\begin{bmatrix}
M_1^T \\
M_2^T \\
M_3^T
\end{bmatrix}
= \begin{bmatrix}
0 \\
0 \\
0
\end{bmatrix}
\]

How do we solve problems of the form

\[
\arg \min \|An\|^2_2, \|n\|^2_2 = 1 ?
\]

Eigenvector of \(A^TA\) with smallest eigenvalue

Derivation from L. Lazebnik; note we negate one of the equations from the cross product
In Practice

Degenerate configurations (e.g., all points on one plane) an issue. Usually need multiplane targets.
In Practice

I pulled a fast one.

We want: \[ p \equiv K_{3\times3}[R_{3\times3}, t_{3\times1}] \quad X_{4\times1} \]

We get: \[ p \equiv M_{3\times4}X_{4\times1} \]

What’s the difference between \( K[R,t] \) and \( M \)?

Solution: QR-decomposition on left-most 3x3 matrix → finite options of a upper triangular matrix * rotation
In Practice

If \( \mathbf{p}_i = \mathbf{Mx}_i \) is overconstrained, the objective function isn’t actually the one you care about.

Instead:
1) initialize parameters with linear model
2) Apply off-the-shelf non-linear optimizer to:

\[
\sum \| \text{proj}(\mathbf{MX}_i) - [u_i, v_i]^T \|_2^2
\]

Advantage: can also add radial distortion, not optimize over known variables, add constraints
What Does This Get You?

Given projection $p_i$ of unknown 3D point $X$ in two or more images (with known cameras $M_i$), find $X$
Triangulation

Given projection $p_i$ of unknown 3D point $X$ in two or more images (with known cameras $M_i$), find $X$.

Why is the calibration here important?
Triangulation

Rays in principle should intersect, but in practice usually don’t exactly due to noise, numerical errors.
Find shortest segment between viewing rays, set $X$ to be the midpoint of the segment.
Triangulation – Non-linear Optim.

Find $X$ minimizing $d(p_1, M_1 X)^2 + d(p_2, M_2 X)^2$
Triangulation – Linear Optimization

\[ p_1 \equiv M_1X \quad p_1 \times M_1X = 0 \quad [p_{1x}]M_1X = 0 \]

\[ p_2 \equiv M_2X \quad p_2 \times M_2X = 0 \quad [p_{2x}]M_2X = 0 \]

Cross Prod. as matrix

\[
\begin{bmatrix}
0 & -a_3 & a_2 \\
a_3 & 0 & -a_1 \\
-a_2 & a_1 & 0
\end{bmatrix}
\begin{bmatrix}
b_1 \\
b_2 \\
b_3
\end{bmatrix}
= [a_x]b
\]

Two eqns per camera for 3 unkn. in X