CSci 4270 and 6270
Computational Vision, Fall 2019
Lecture 11: Two Camera Geometry
March 4, 2021

Cameras and Calibration

- What we have covered previously (all the way back to Lecture 5):
  - Perspective projections
  - Transformations
  - Extrinsic and intrinsic parameters
- What we cover here:
  - Real lenses and lens distortions (briefly)
  - Special case of relative positions of two cameras: rotation about optical center, and images of planar surfaces
  - General case of two images — intro to stereo geometry
  - Introduction to camera geometry estimation problems and forming montages.

Aside on Debugging Computer Vision Code

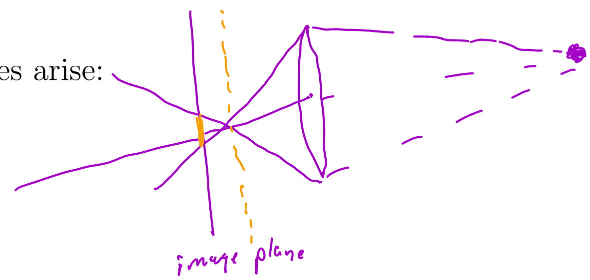
- Debugging computer vision software is challenging!
  - Large data
  - Complicated intensity patterns
  - Algorithms that will produce imperfect results even if implemented "correctly';'.
- Approaches to use, sometims in combination:
  - Small, synthetic images where results are easily predicted. Example: the disk image.
  - Tiny image – simple 2d arrays — where the results of a computation can be hand-traced and verified. Example: make a synthetic gradient image that may be even as small as 4x6 and trace the equations to compute the energy and extract the seam.
  - Choose a particular pixel location and track the value of it and perhaps its neighboring pixels as a computation progresses. Write utility functions to support this.

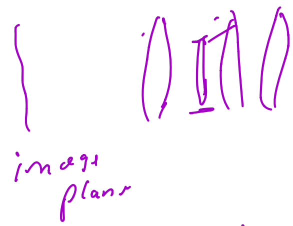
Lens Effects

When we consider real lenses, a number of issues arise:

- Defocus blurring
- Vignetting and other intensity distortions
- Lens distortions

We will discuss each of these briefly.





brighter in center

Radial Lens Distortion

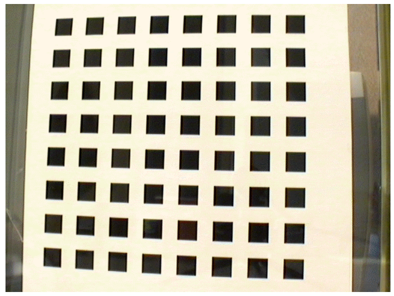


Image from Microsoft Research

- Shift radially away from the center (barrel distortion) or toward the center (pin-cushion).
- Quadratic or quartic model, applied after division but before decentering

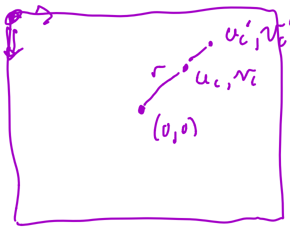
Radial Lens Distortion

- Let (u_c, v_c) be a point in the centered, camera coordinate system — i.e. the coordinates with $(0,0)$ at the projection of the optical axis rather than at the upper left corner — and compute $r = \sqrt{u_c^2 + v_c^2}$.

- Then the distorted (still centered) coordinates are

$$
\begin{aligned}
u_c' &= u_c(1 + \kappa_1 r^2 + \kappa_2 r^4) \\
v_c' &= v_c(1 + \kappa_1 r^2 + \kappa_2 r^4)
\end{aligned}
$$

κ_1, κ_2



  Sometimes $\kappa_2 = 0$, so that only a quadratic model is used.

- (u_c', v_c') is decentered to obtain the actual pixel coordinates.

- Unfortunately, this model is not algebraically invertible, so either approximations or numerical-inverse methods must be used when going from image pixels back to world coordinates.

For the remainder of the discussion in this set of notes we ignore radial-lens distortion.

Recall Our Camera Model

- We model the mapping of a point in the world onto an image as a perspective projection and describe using a 3x4 matrix:

$$\mathbf{M} = \underbrace{\mathbf{K}}\left(\overset{3\times3}{\mathbf{R}^\top} \quad \overset{3\times1}{-\mathbf{R}^\top\mathbf{t}}\right)$$

$$\left(\underbrace{KR^T}_{3\times3} \quad \underbrace{-KR^Tt}_{3\times1}\right)$$

$R, \vec{t}$

where

  - **K** is the 3x3, upper triangular, intrinsic parameter matrix,
  - **R** and **t** describe position "camera" relative to the world coordinate system with a rotation followed by a translation.

- For convenience we often replace $\mathbf{R}' = \mathbf{R}^\top$ by **R** and $\mathbf{t}' = -\mathbf{R}\mathbf{t}$, reversing the transformation. Then we drop the $'$ and just write

$$\mathbf{M} = \mathbf{K}\left(\mathbf{R} \quad \mathbf{t}\right)$$

This can be somewhat confusing so we have to be very careful to be clear what is meant throughout.

- Sometimes we just talk about **M** using its components, its rows, or its columns. For example,

$$\mathbf{M} = \left(\mathbf{m}_1 \quad \mathbf{m}_2 \quad \mathbf{m}_3 \quad \mathbf{m}_4\right),$$

$$m_1 = \begin{pmatrix} m_{11} \\ m_{21} \\ m_{31} \end{pmatrix} \qquad \begin{pmatrix} m_{11} & m_{12} & \cdots \\ m_{21} & m_{22} & \\ m_{31} & m_{32} & \end{pmatrix}$$

where each $\mathbf{m}_i$ is a 3x1 column vector.

- To work with perspective projection of points we use homogeneous coordinates in 2d and 3d.

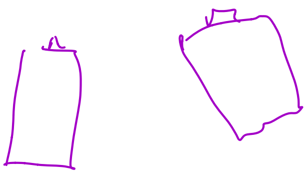
$$\tilde{u} = \begin{pmatrix} u \\ v \\ w \end{pmatrix} \qquad x = u/w \qquad y = v/w \qquad \tilde{x} = \begin{pmatrix} X \\ Y \\ Z \\ 1 \end{pmatrix}$$

2d

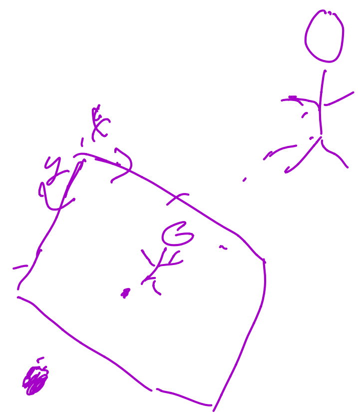
$$\begin{pmatrix} x \\ y \\ 1 \end{pmatrix}$$

Multiple Images

- We now extend our understanding of a single camera to multiple images and "cameras".
  - Each new image, even when taken with the same physical camera, is modeled using a distinct camera matrix because its extrinsic parameters, its intrinsic parameters, or both may have changed.
- We need to understand the relationship between different images and cameras.



  - As one example of the utility of this, it tells us when mosaics can and can not be build.
  - By this we mean that we can transform one image onto another accurately without knowing depth and without (too much) ghosting.
- We will look at
  - Two special cases
  - The two-view general case



Multiple Images: Rotation About the Image Center

The first simplified model considers what happens when the camera is rotated in place...

- For convenience, we place the world coordinate system origin at the optical center.
- Doing this, each "camera" can be written

$$\mathbf{M}_i = \mathbf{K}_i \left(\mathbf{R}_i \quad \mathbf{0}\right) \qquad \vec{t} \qquad \left(K_i R_i \mid \vec{0}\right) \leftarrow \text{last is } 0$$

- Also, any pixel in image coordinates corresponds to a line in the world of the (parametric) form

$$\mathbf{p}(t) = \begin{pmatrix} \hat{\mathbf{d}}t \\ 1 \end{pmatrix}$$

  where $\mathbf{d}$ is a three-component, unit direction vector.

- We will then derive the projection of $\mathbf{p}(t)$ onto each image to form the homogeneous coordinate locations $\tilde{\mathbf{u}}_i$ for $i = 1, 2$.
- Next we will eliminate the unknowns t and $\mathbf{d}$ since the expressions must work for all line directions.

$$\vec{p}(t) = \begin{pmatrix} \hat{d}t \\ 1 \end{pmatrix} \qquad \hat{d} \text{ is direction}$$

$$\underset{3\times4}{M_i} \underset{4\times1}{\begin{pmatrix} \hat{d}t \\ 1 \end{pmatrix}} = \underset{3\times1}{\tilde{u}_i} = \qquad \tilde{u}_i = K_i R_i \hat{d} t$$

drop t

$$\underset{3\times1}{\tilde{u}_1} = \underset{3\times3}{K_1} \underset{3\times3}{R_1} \underset{3\times1}{\hat{d}} \qquad \tilde{u}_2 = K_2 R_2 \hat{d}$$

$$\hat{d} = R_1^T K_1^{-1} \tilde{u}_1$$
$$\hat{d} = R_2^T K_2^{-1} \tilde{u}_2$$
$$\Rightarrow R_1^T K_1^{-1} \tilde{u}_1 = R_2^T K_2^{-1} \tilde{u}_2$$

$$\Rightarrow \tilde{u}_2 = K_2 R_2 R_1^T K_1^{-1} \tilde{u}_1$$

Rotation About the Image Center (continued)

- We will use the foregoing to show in class that pixel location $\tilde{\mathbf{u}}_1$ in image I_1 maps onto pixel location

$$\tilde{\mathbf{u}}_2 = \mathbf{K}_2\mathbf{R}_2\mathbf{R}_1^\top\mathbf{K}_1^{-1}\tilde{\mathbf{u}}_1$$

  in image I_2.

- This is independent of the distance of points from the camera!

- $\mathbf{H} = \mathbf{K}_2\mathbf{R}_2\mathbf{R}_1^\top\mathbf{K}_1^{-1}$ is a 3×3 homography matrix known as a *conjugate rotation*.

- We will soon consider methods to estimate the parameters of $\mathbf{H}$ (not the individual $\mathbf{K}$'s and $\mathbf{R}$'s).

$$\tilde{u}_2 = \underset{3\times 3}{H}\,\tilde{u}_1$$

where it goes in image 2

pixel loc in img 1

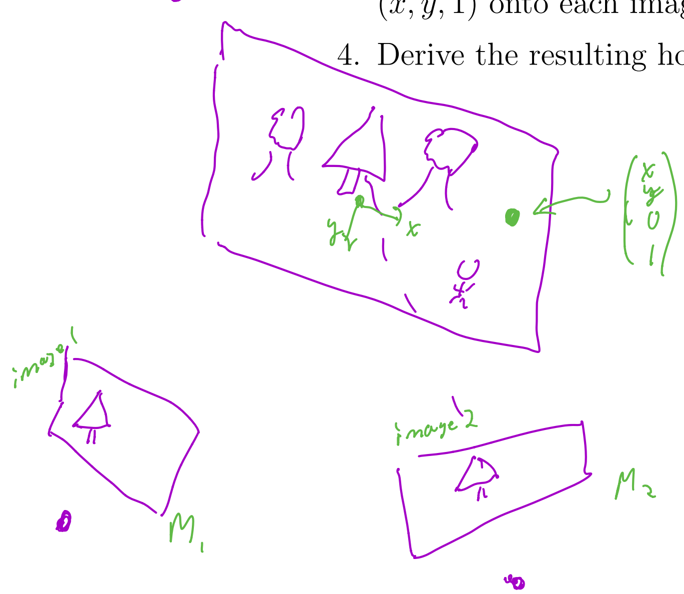
Multiple Images of a Planar Surface

- Suppose we are taking images of a planar surface
  - a wall, the earth (in limited view), a billboard,
  - almost anything when the distance from the camera is large relative to the distance variation
- Then, even if we move the camera (or cameras) in arbitrary ways, a pixel location $\tilde{\mathbf{u}}_1$ in image I_1 maps onto pixel location $\tilde{\mathbf{u}}_2$ in I_2 as

$$\tilde{\mathbf{u}}_2 = \mathbf{H}\tilde{\mathbf{u}}_1$$

  where $\mathbf{H}$ is a 3×3 homography.
- We do not need to know the parameters of the plane, only that the plane exists!
- We can prove this by assigning the x-y coordinate system in the world to coincide with the plane, so all points have $z = 0$. We'll go over this in class.
- Here is the outline of how we'll show this in class:
  1. Write
  $$\mathbf{M}_i = \begin{pmatrix}\mathbf{m}_{i,1} & \mathbf{m}_{i,2} & \mathbf{m}_{i,3} & \mathbf{m}_{i,4}\end{pmatrix}$$
  to describe the camera matrix or image $i = 1$ and image $i = 2$.
  2. Write any point on the plane in homogeneous coordinates as $\tilde{\mathbf{p}}^\top = (x, y, 0, 1)^\top$.
  3. Show that we can ignore the $z = 0$ value and the 3rd column of $\mathbf{M}_i$ for each camera i and therefore obtain a 3x3 homography mapping $(x, y, 1)$ onto each image.
  4. Derive the resulting homography between images.

Eq'n of plane
$z=0$



$x M_{i1} + y M_{i2} + M_{i4}$

$$\tilde{u}_i = M_i \begin{pmatrix}x\\y\\0\\1\end{pmatrix} = \begin{pmatrix}m_{i1} & m_{i2} & \cancel{m_{i3}} & m_{i4}\end{pmatrix}\begin{pmatrix}x\\y\\\cancel{0}\\1\end{pmatrix}$$

$$= \underbrace{\begin{pmatrix}m_{i1} & m_{i2} & m_{i4}\end{pmatrix}}_{M_i'}\underbrace{\begin{pmatrix}x\\y\\1\end{pmatrix}}_{\tilde{p}'}$$

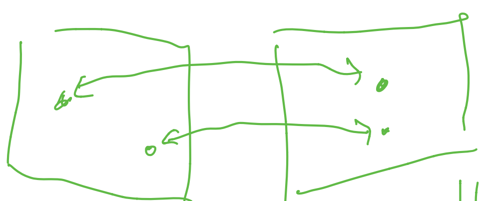
$$\underset{}{\tilde{u}_i} = \underset{3\times3}{M_i'}\ \underset{3\times1}{\tilde{p}_i}$$

$$\tilde{u}_1 = M_1'\tilde{p}' \qquad \tilde{u}_2 = M_2'\tilde{p}'$$

$$\tilde{u}_2 = \underbrace{M_2' M_1'^{-1}}_{H}\tilde{u}_1$$

When Can We Build a Montage?

- We now have the two cases: rotation about the optical center, and images of planar surface.
- In each case, the images are related by a planar homography:
  - Given any point location in one image and the 3x3 homography matrix **H**, we can compute the location of the corresponding point in the other image.
- The depths of the world coordinate points need not be known in order to map between images — only **H** must be known.
- In other words, a single 3x3 matrix dictates the (correct) mapping of an entire image onto another image.
- When want to build a mosaic of images, the goal of estimation will be to extract the parameters of this **H** matrix.
- In some cases, affine and even similarity transformations can serve as a good approximation to **H**.

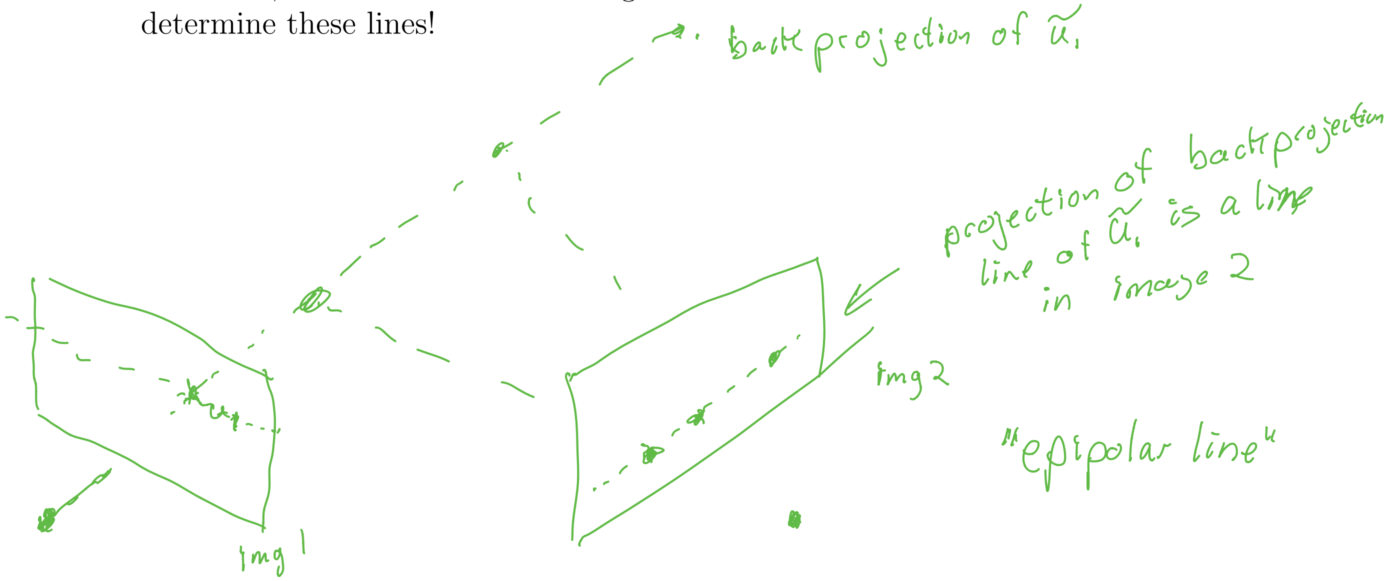


if we know H exists then we form matches and estimate params of H

General Case of Two Images

When the optical centers of the cameras do not coincide...

- The backprojection of a point p_1 in I_1 is a line of possible points in the world.
- When this line is "projected" down into image I_2, it forms a line in I_2.
  - This line is called the "epipolar line"
- More generally, this epipolar line in I_2 corresponds to another line, back in I_1, that contains p_1.
- Of course, we must know something about the two cameras in order to determine these lines!



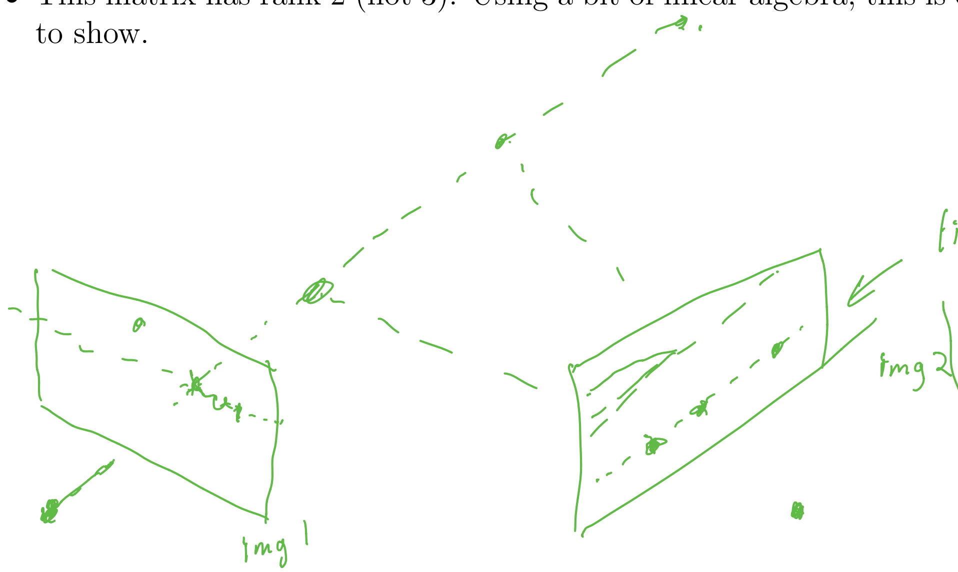
The Fundamental Matrix

F is 3x3

- For each pair of images I_1 and I_2, there exists a special 3×3 matrix $\mathbf{F}$ called the *fundamental matrix*
- If $\tilde{\mathbf{x}}_1$ is the homogeneous coordinate vector for a point from I_1, then the epipolar line in I_2 has (implicit form) parameters

$$\mathbf{a}_2 = \mathbf{F}\cancel{\tilde{\mathbf{x}}}_1.\ \tilde{u}_1$$

- This fundamental matrix may be computed from the camera matrices or estimated when there are at least 7 known matching points between the images.
- This matrix has rank 2 (not 3). Using a bit of linear algebra, this is easy to show.



line parameters

$\tilde{a}_2 = F\tilde{u}_1$

$\tilde{u}_2^T F \tilde{u}_1 = 0$

Two View Summary

There are only two cases for relating the image projections of a point in the world:

- Notation: point coordinates are $\tilde{\mathbf{x}}_1$ in I_1 and $\tilde{\mathbf{x}}_2$ in I_2.
- When the cameras have the same optic center or the image points are of a planar surface, then there is a 3×3 homography matrix $\mathbf{H}$ (rank 3 and unique up to a scale factor) such that

$$\tilde{\mathbf{x}}_2 = \mathbf{H}\tilde{\mathbf{x}}_1$$

- Otherwise, there is a 3×3, rank-2 matrix $\mathbf{F}$ such that

$$\tilde{\mathbf{x}}_2^\top \mathbf{F} \tilde{\mathbf{x}}_1 = 0.$$

$F x_1 = \tilde{a}_2$ line params in img 2

$F^T \tilde{x}_2 = \tilde{a}_1$ line params in img 1

- In the first case, **H determines** the matching point location for $\tilde{\mathbf{x}}_1$ in image I_2.
- In the second case, **F constrains** the matching point location in image I_2 to be along a line.

Estimation Problems

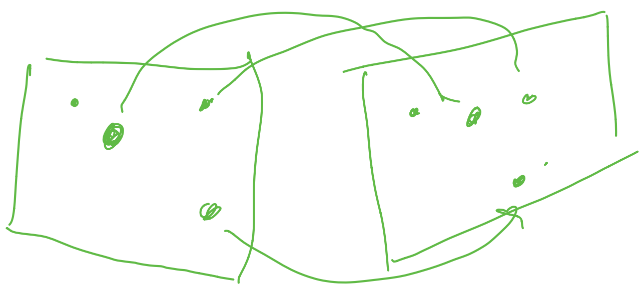
We now have three matrices for which we would like to know the parameters

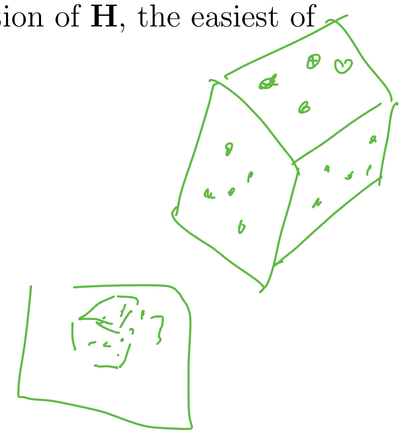
- 3×4 camera matrix $\mathbf{M}$ (plus radial lens distortion terms) for a single camera
- 3×3 homography matrix $\mathbf{H}$ for two important special cases of two cameras
- 3×3 fundamental matrix $\mathbf{F}$ for the general case
- An important consideration to reinforce your understanding is to think about how many pairs of matching point locations are needed to determine each of these three matrices.

Overview of Solving These Estimation Problems

Two primary steps:

- 1. Establish "correspondences"
  - Between features in two images (for **H** or for **F**)
  - Between points on a known "calibration target" and points in an image.
- 2. Estimate parameters based on these correspondences:
  - Use the defining equations for **H**, **F** and **M** to write a least-squares objective function.
- In the next set of notes, we will focus on estimation of **H**, the easiest of these.





Correspondences for Camera Calibration: Brief Overview

Figure 1: Image from `cmp.felk.cvut.cz`

- A calibration target is created, usually a three-dimensional object, that has a known (machined) shape and precisely measured markings.
- The target is placed in a fixed position, so the markings have known positions in 3D.
- Extract features in images of the target, e.g.
  - Edges (with subpixel accuracy) and then fit lines
  - Intersect lines to form corners.
- Use corners and knowledge of target to establish 3d-to-2d correspondences.

Summary

- Lens distortion as a quadratic or quartic function of the image coordinates.
- The effect of camera movement or using two cameras:
  - Homography matrix $\mathbf{H}$ for mapping between images taken of a planar surface or taken by a rotating (not translating) camera.
  - Fundamental matrix $\mathbf{F}$ for establishing epipolar lines in the general case.
- Estimation problems:
  - Establish correspondences and derive constraints to estimate $\mathbf{H}$, $\mathbf{F}$ and $\mathbf{M}$.