Homographies and Mosaics

© Jeffrey Martin (jeffrey-martin.com)

with a lot of slides stolen from
Steve Seitz and Rick Szeliski

15-463: Computational Photography
Alexei Efros, CMU, Fall 2006

Why Mosaic?

Are you getting the whole picture?

- Compact Camera FOV = 50 x 35°

Slide from Brown & Lowe

Why Mosaic?

Are you getting the whole picture?

- Compact Camera FOV = 50 x 35°
- Human FOV = 200 x 135°

Slide from Brown & Lowe

Why Mosaic?

Are you getting the whole picture?

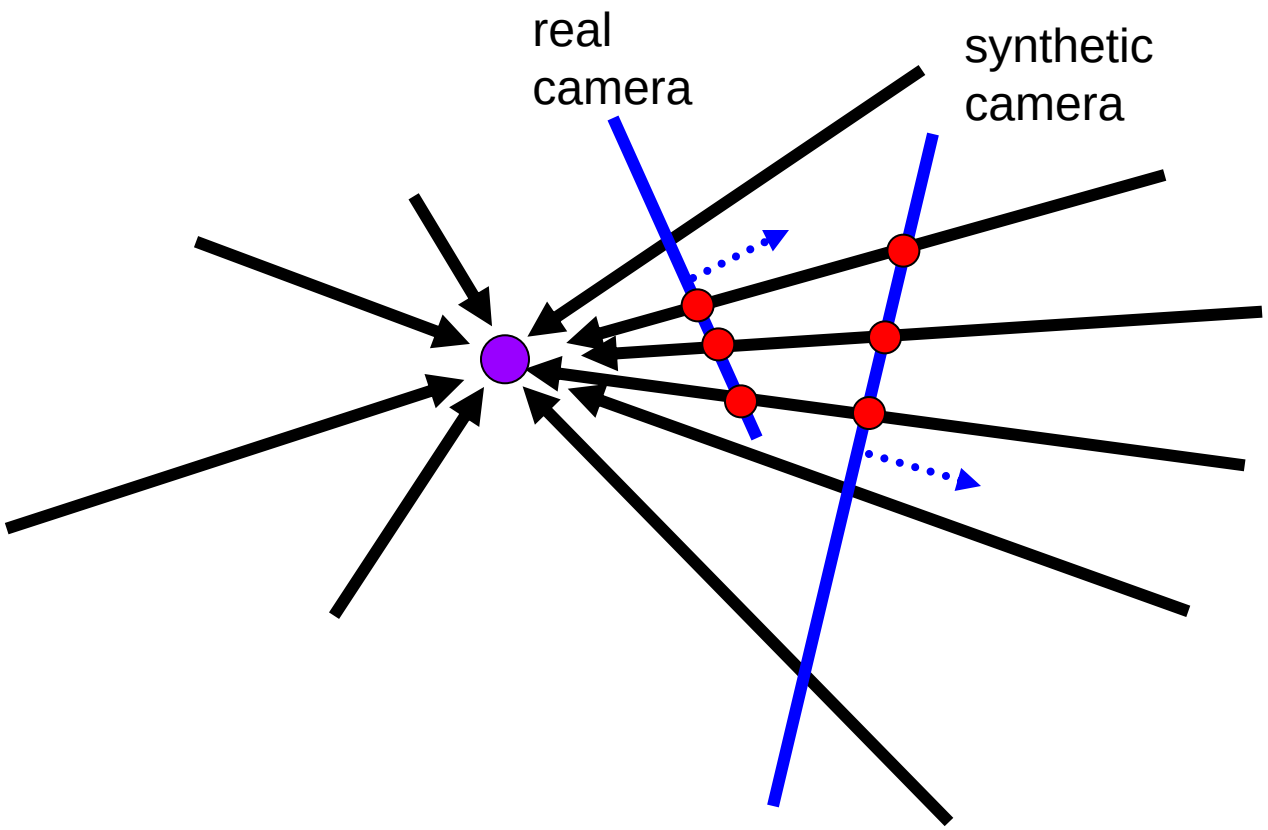
- Compact Camera FOV = 50 x 35°
- Human FOV = 200 x 135°
- Panoramic Mosaic = 360 x 180°

Slide from Brown & Lowe

Mosaics: stitching images together

virtual wide-angle camera

A pencil of rays contains all views



Can generate any synthetic camera view
as long as it has **the same center of projection**!

How to do it?

Basic Procedure

- Take a sequence of images from the same position
  - Rotate the camera about its optical center
- Compute transformation between second image and first
- Transform the second image to overlap with the first
- Blend the two together to create a mosaic
- If there are more images, repeat

…but **wait**, why should this work at all?

- What about the 3D geometry of the scene?
- Why aren’t we using it?

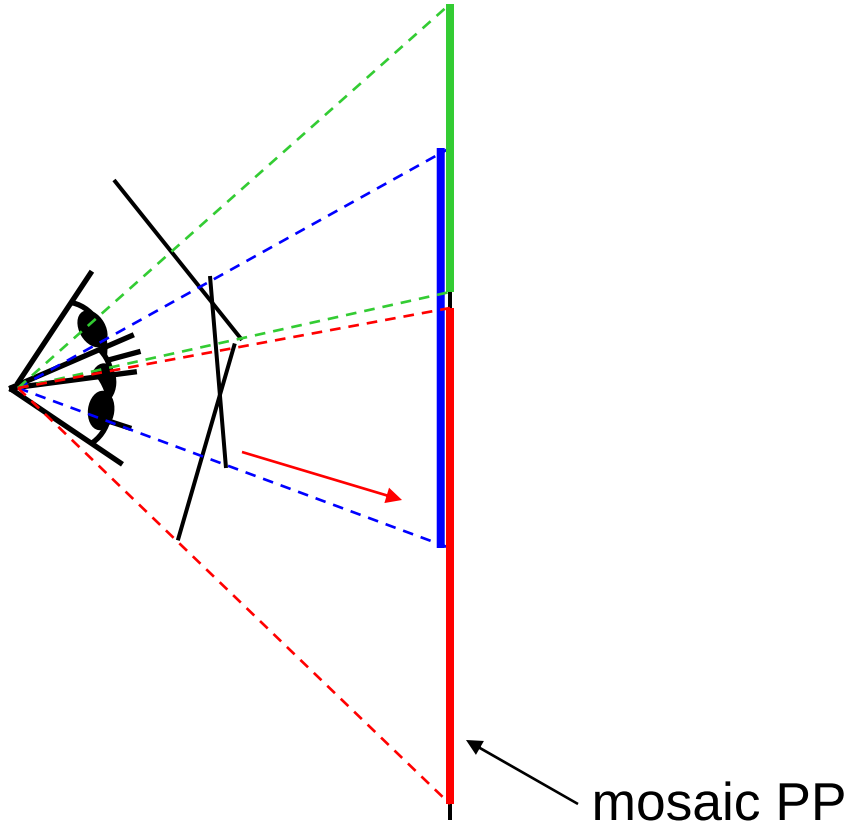
Aligning images

left on top

right on top

Translations are not enough to align the images

Image reprojection



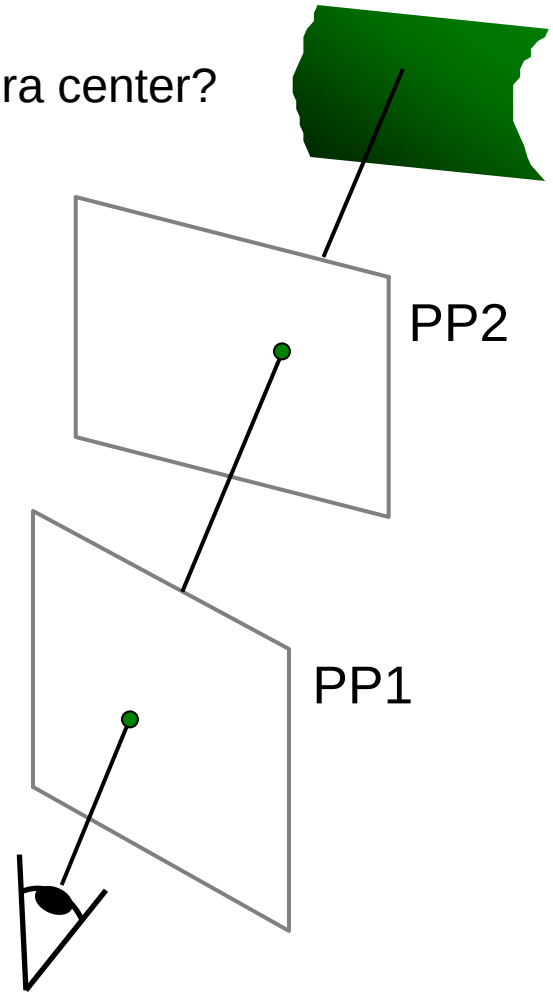
The mosaic has a natural interpretation in 3D

- The images are reprojected onto a common plane
- The mosaic is formed on this plane
- Mosaic is a *synthetic wide-angle camera*

Image reprojection

Basic question

- How to relate two images from the same camera center?
  - how to map a pixel from PP1 to PP2



Answer

- Cast a ray through each pixel in PP1
- Draw the pixel where that ray intersects PP2

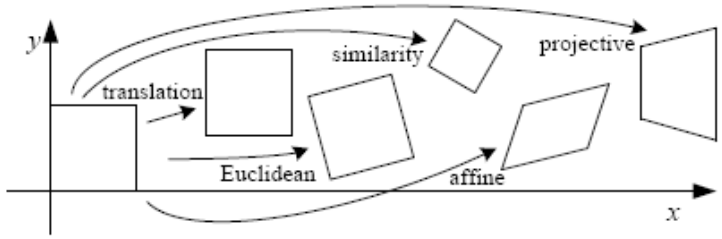
But don’t we need to know the geometry of the two planes in respect to the eye?

Observation:
Rather than thinking of this as a 3D reprojection, think of it as a 2D **image warp** from one image to another

Back to Image Warping

Which t-form is the right one for warping PP1 into PP2?

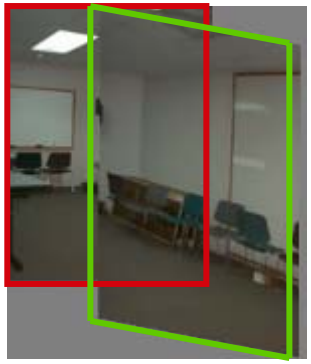
e.g. translation, Euclidean, affine, projective



Translation

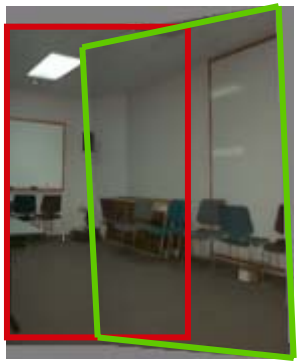
2 unknowns

Affine



6 unknowns

Perspective



8 unknowns

Homography

A: Projective – mapping between any two PPs with the same center of projection

- rectangle should map to arbitrary quadrilateral
- parallel lines aren't
- but must preserve straight lines
- same as: project, rotate, reproject

called Homography

$$\begin{bmatrix} wx' \\ wy' \\ w \end{bmatrix} = \begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

$$\mathbf{p'} \qquad \mathbf{H} \qquad \mathbf{p}$$

To apply a homography **H**

- Compute **p'** = **Hp** (regular matrix multiply)
- Convert **p'** from homogeneous to image coordinates

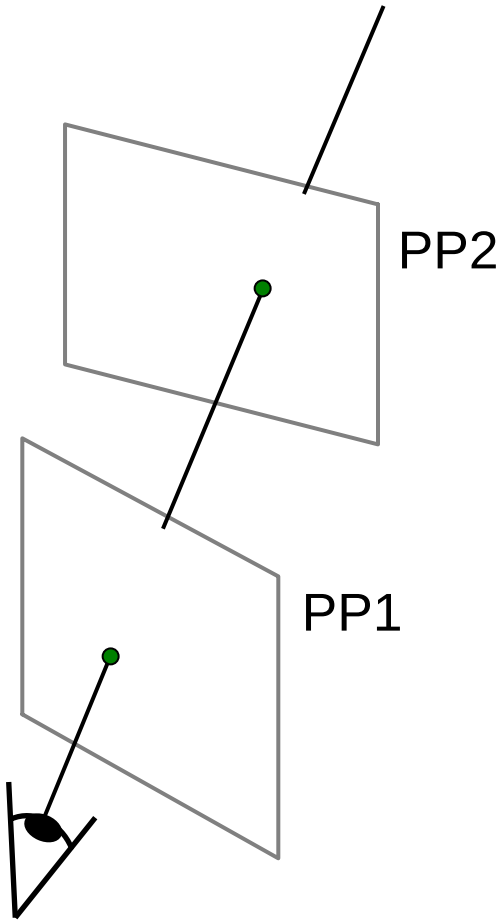
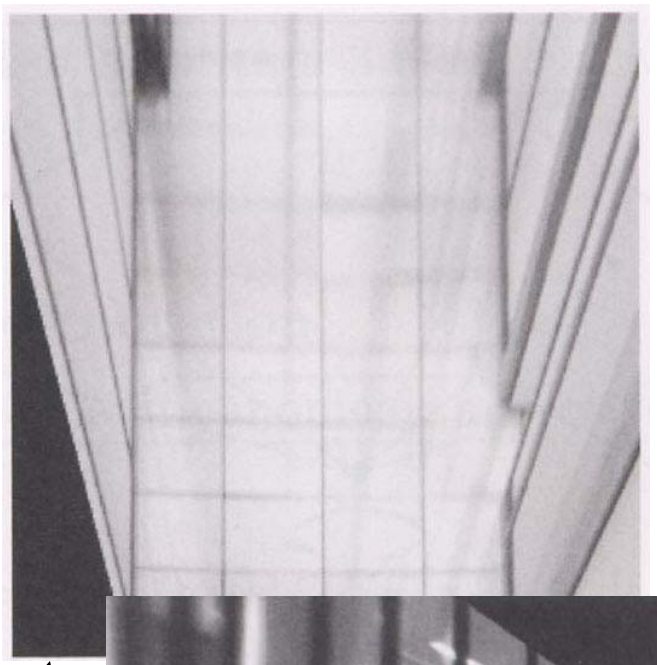
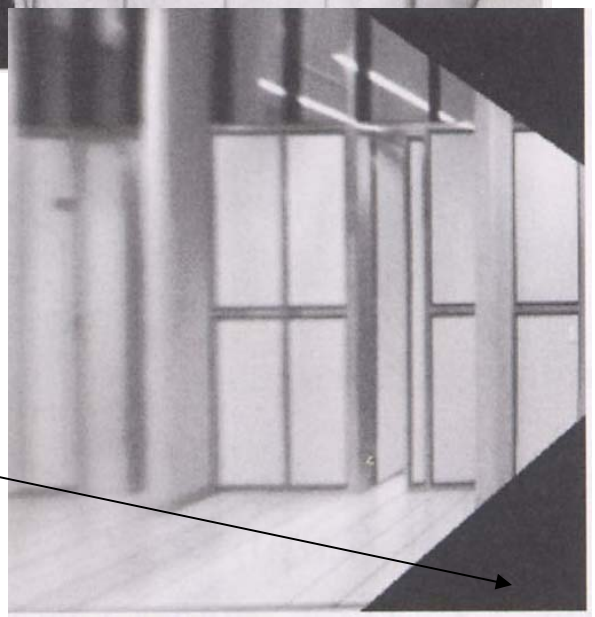


Image warping with homographies





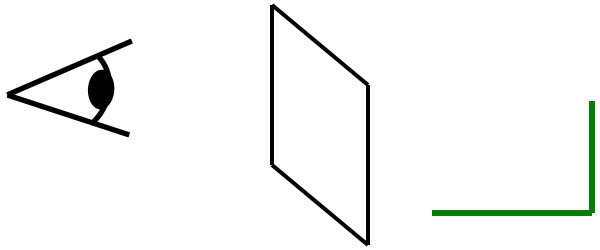
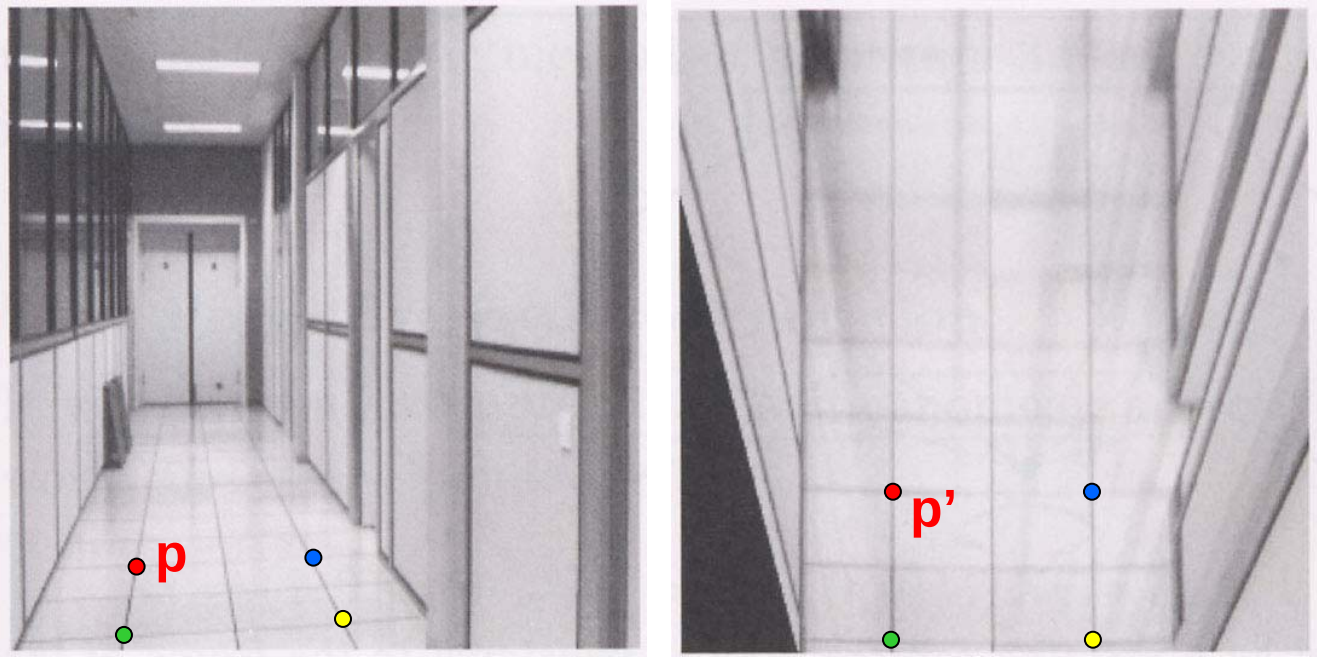


image plane in front

black area
where no pixel
maps to

Image rectification



To unwarp (rectify) an image

- Find the homography **H** given a set of **p** and **p’** pairs
- How many correspondences are needed?
- Tricky to write H analytically, but we can solve for it!
  - Find such H that “best” transforms points p into p’
  - Use least-squares!

Least Squares Example

Say we have a set of data points (X1,X1'), (X2,X2'), (X3,X3'), etc. (e.g. person's height vs. weight)

We want a nice compact formula (a line) to predict X's from Xs: Xa + b = X'

We want to find a and b

How many (X,X') pairs do we need?

$$X_1 a + b = X_1'$$
$$X_2 a + b = X_2'$$

$$\begin{bmatrix} X_1 & 1 \\ X_2 & 1 \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} X_1' \\ X_2' \end{bmatrix}$$

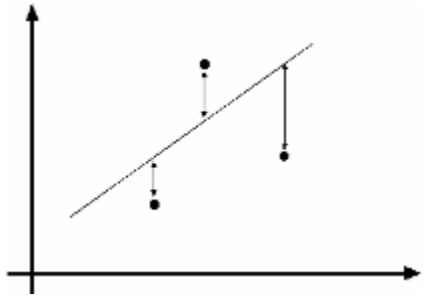
Ax=B

What if the data is noisy?

$$\begin{bmatrix} X_1 & 1 \\ X_2 & 1 \\ X_3 & 1 \\ \dots & \dots \end{bmatrix} \begin{bmatrix} a \\ b \end{bmatrix} = \begin{bmatrix} X_1' \\ X_2' \\ X_3' \\ \dots \end{bmatrix}$$

overconstrained

$$\min \|Ax - B\|^2$$



Solving for homographies

$$\mathbf{p'} = \mathbf{Hp}$$

$$\begin{bmatrix} wx' \\ wy' \\ w \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Can set scale factor *i*=1. So, there are 8 unkowns.

Set up a system of linear equations:

$$\mathbf{Ah = b}$$

where vector of unknowns h = [a,b,c,d,e,f,g,h]^T

Need at least 8 eqs, but the more the better…

Solve for h. If overconstrained, solve using least-squares:

$$\min \left\| Ah - b \right\|^2$$

Can be done in Matlab using "\" command

- see "help lmdivide"

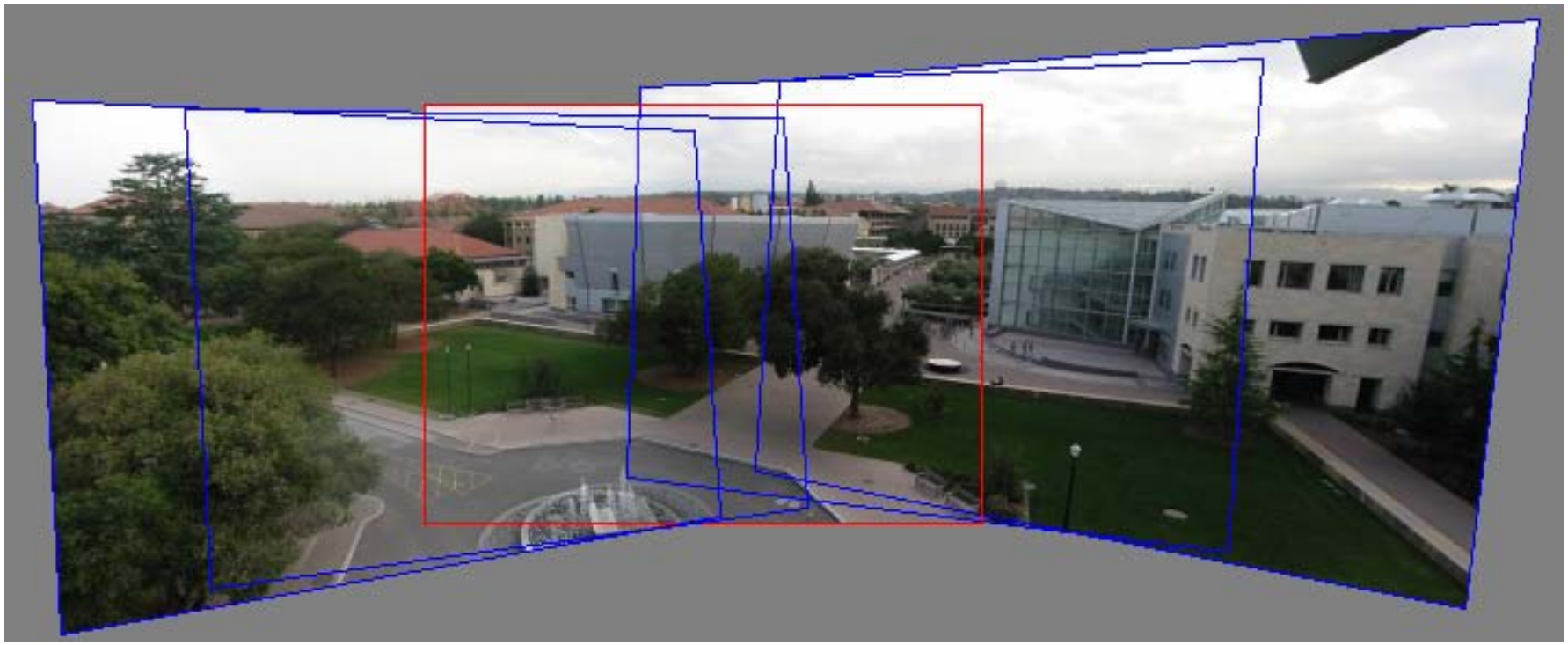
Fun with homographies

Original image

St.Petersburg
photo by A. Tikhonov

Virtual camera rotations

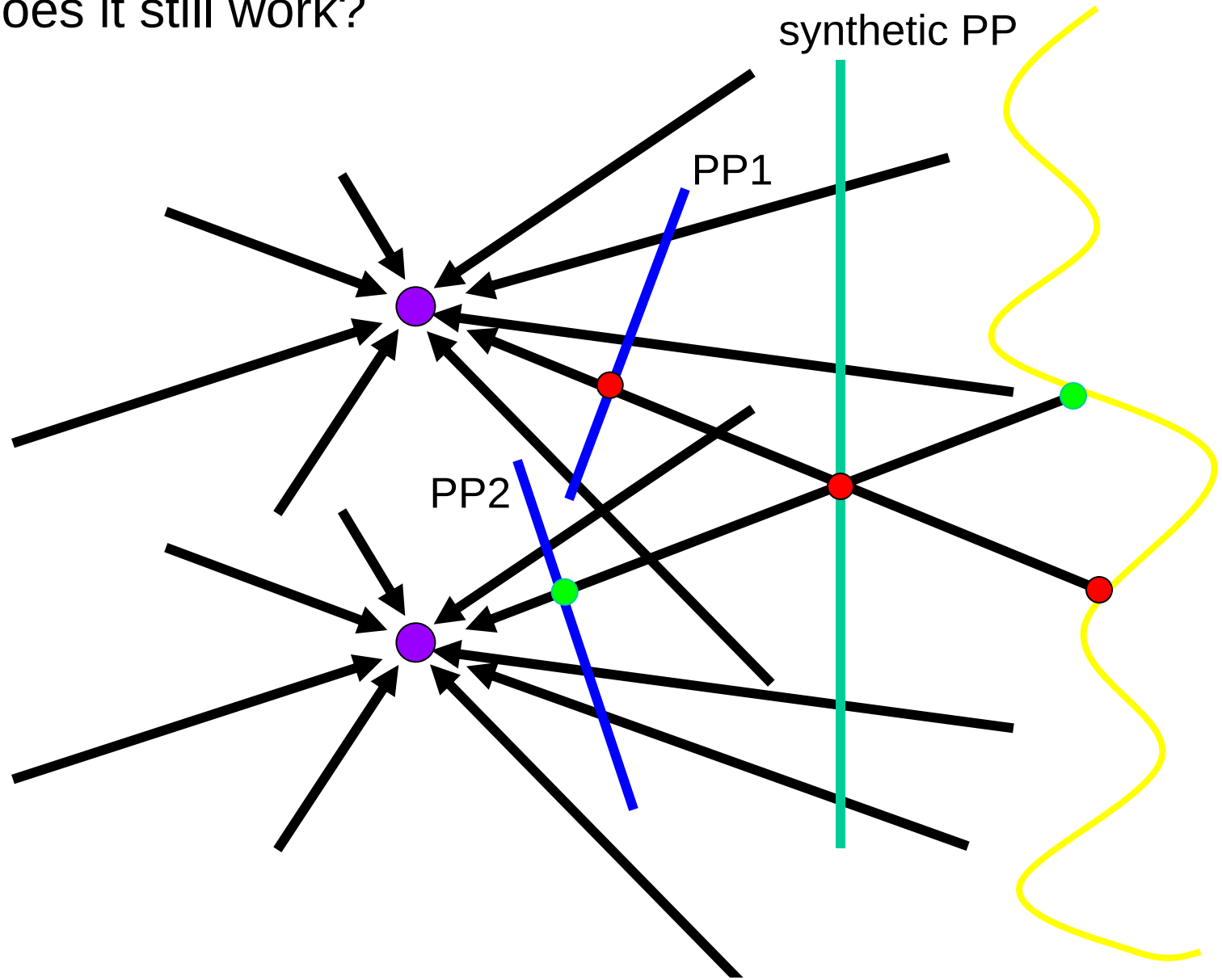
Panoramas



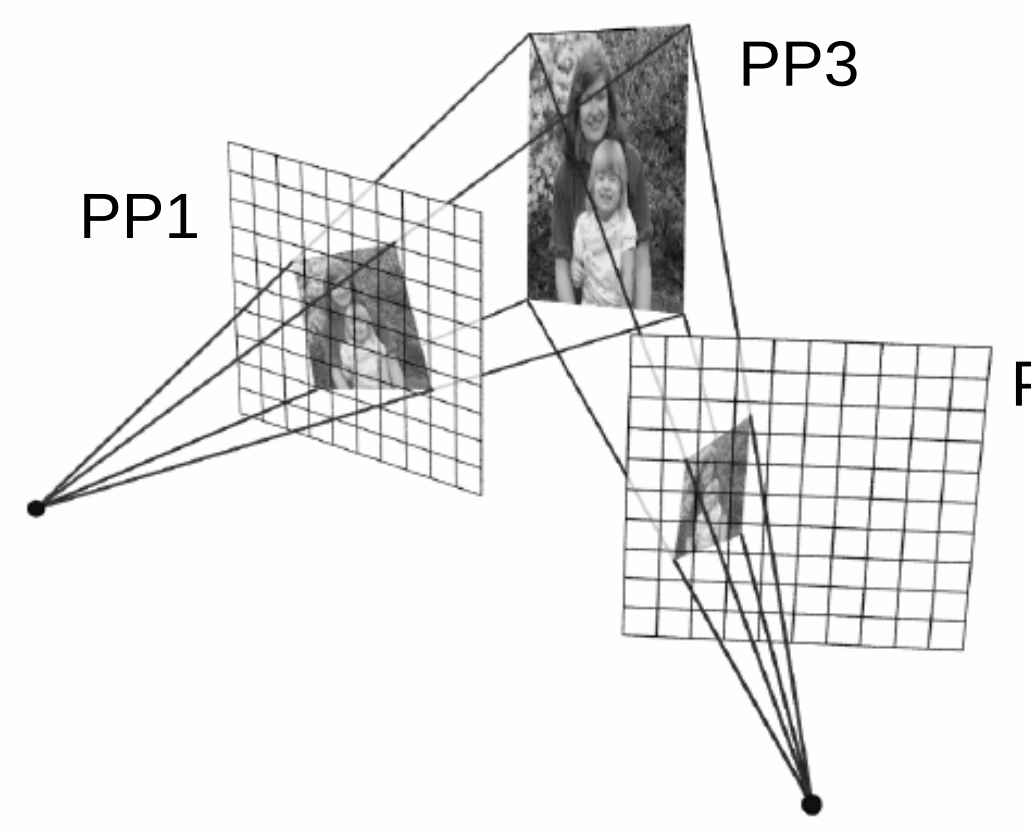
1. Pick one image (red)
2. Warp the other images towards it (usually, one by one)
3. blend

changing camera center

Does it still work?



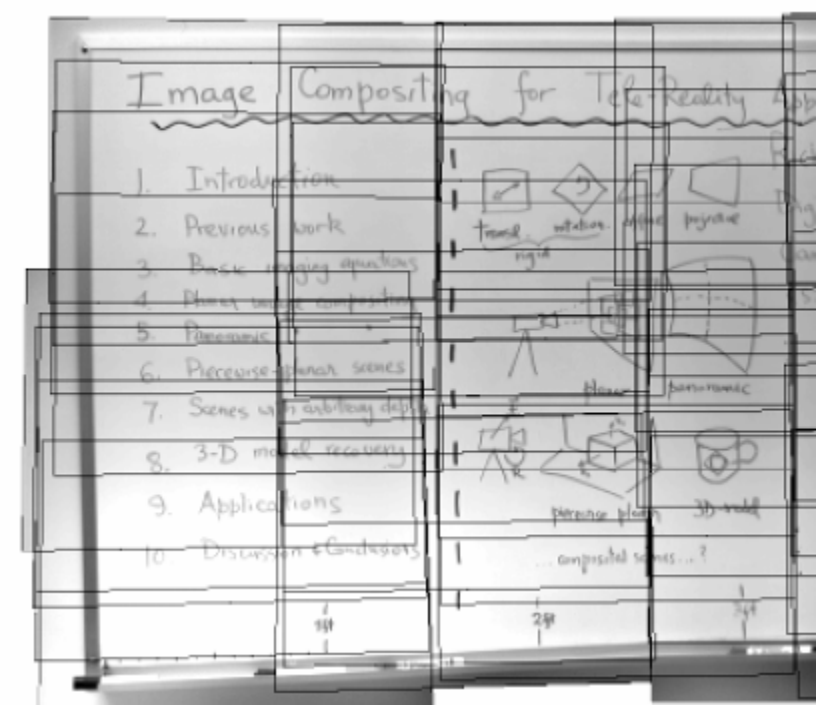
Planar scene (or far away)

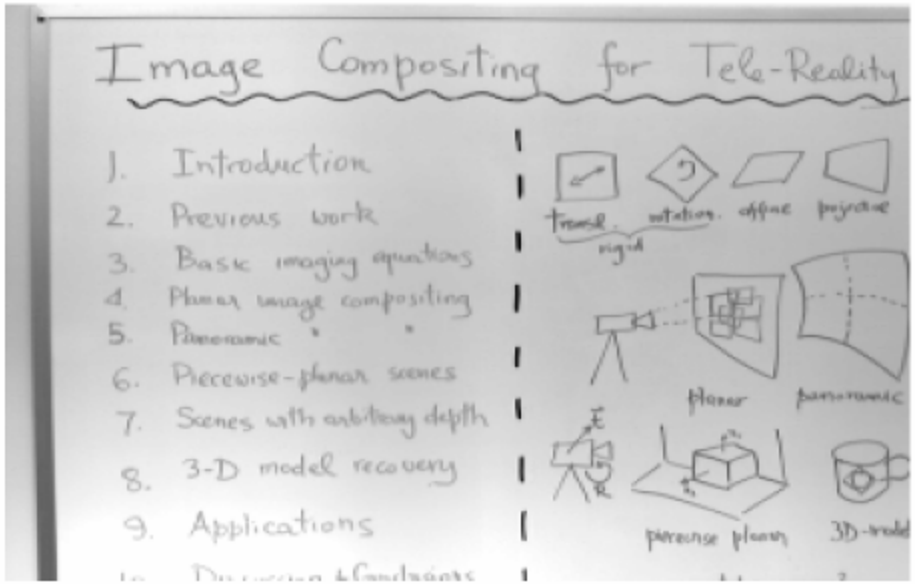


PP3 is a projection plane of both centers of projection, so we are OK!

This is how big aerial photographs are made

Planar mosaic





Programming Project #3

Homographies and Panoramic Mosaics

- Capture photographs (and possibly video)
  - Might want to use tripod
- Compute homographies (define correspondences)
  - will need to figure out how to setup system of eqs.
- (un)warp an image (undo perspective distortion)
- Produce 3 panoramic mosaics (with blending)
- Do some of the Bells and Whistles

Bells and Whistles

Blending and Compositing

- use homographies to combine images or video and images together in an interesting (fun) way. E.g.
  - put fake graffiti on buildings or chalk drawings on the ground
  - replace a road sign with your own poster
  - project a movie onto a building wall
  - etc.

Bells and Whistles

Capture creative/cool/bizzare panoramas

- Example from UW (by Brett Allen):

- Ever wondered what is happening inside your fridge while you are not looking?

Capture a 360 panorama (a bit tricky… talk in next next class)

Bells and Whistles

Video Panorama

- Capture two (or more) stationary videos (either from the same point, or of a planar/far-away scene). Compute homography and produce a video mosaic. Need to worry about synchronization (not too hard).
- e.g. capturing a football game from the sides of the stadium

Other interesting ideas?

- talk to me

From previous year's classes

Ben Hollis, 2004

Ben Hollis, 2004

Eunjeong Ryu (E.J), 2004

Matt Pucevich , 2004

Go Explore!

Ken Chu, 2004