PITT | SWANSON ENGINEERING
ELECTRICAL & COMPUTER

ECE 1390/2390 Image Processing and Computer Vision – Fall 2021

Multiple Views - Homographies and Projection

Ahmed Dallal

Reading and announcements

- FP 7.1 and 8 (all)
- HW 3:
  - due November 1st.
  - based upon last week and multiple view material
- Quiz 3 on Wednesday (10/20).
- Plagiarism!

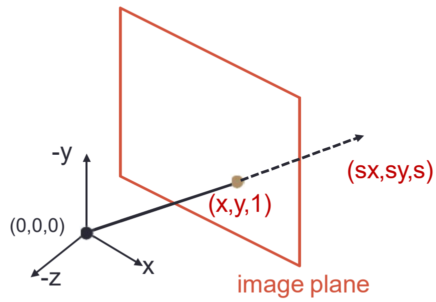
Two views…

- Projective transforms from image to image
- Some more projective geometry
  - Points and lines and planes
- Two arbitrary views of the same scene
  - Calibrated – “Essential Matrix”
  - Two uncalibrated cameras “Fundamental Matrix”
    - Gives epipolar lines

Image to image projections

The projective plane

- What is the geometric intuition of using homogenous coordinates ?
  - a point in the image is a *ray* in projective space



- Each *point* (x,y) on the plane (at z=1) is represented by a *ray* (sx,sy,s)
  – all points on the ray are equivalent: (x, y, 1) ≅ (sx, sy, s)

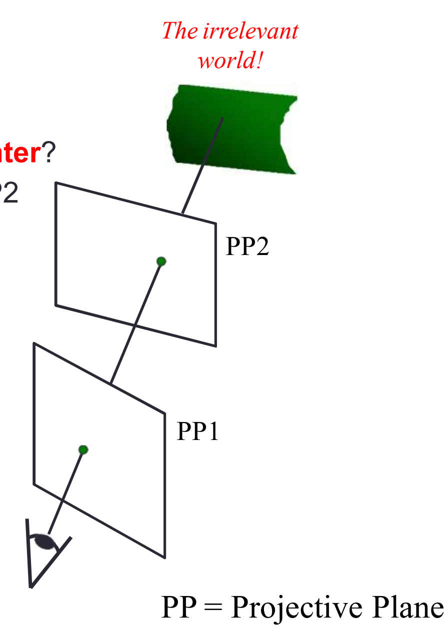
Image reprojection

- Basic question
  - How to relate two images from the **same camera center**?
    - how to map a pixel from projective plane PP1 to PP2

Answer

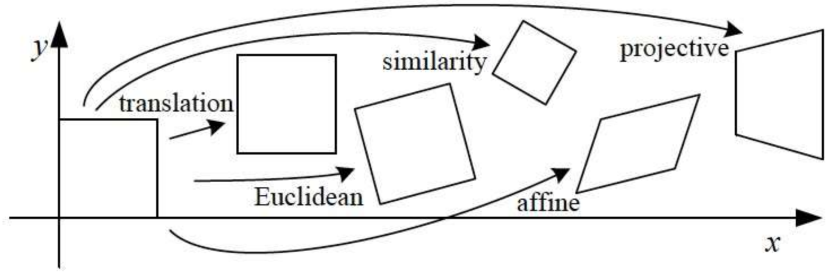
- Cast a ray through each pixel in PP1
- Draw the pixel where that ray intersects PP2

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

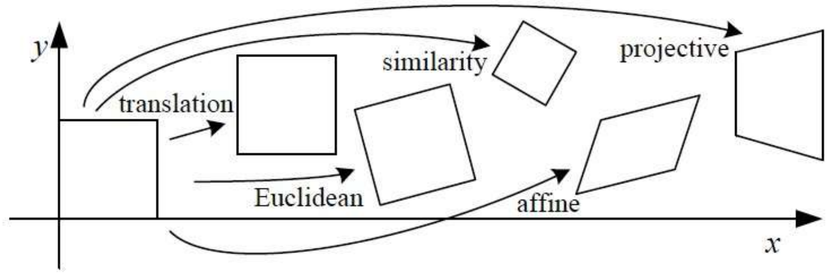


Source: Alyosha Efros

2D Transformations

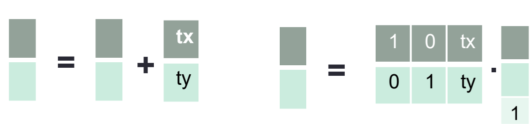


2D Transformations

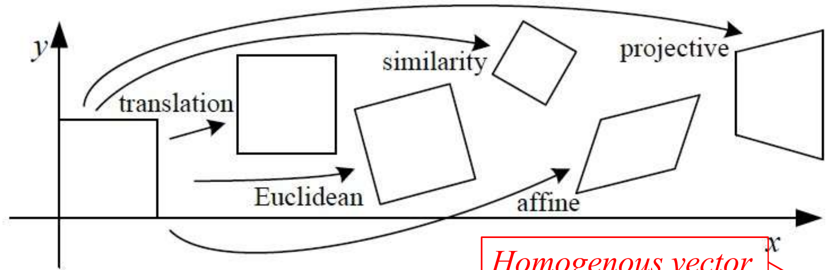


Example: translation

$$x' = x + t \qquad x' = \begin{bmatrix} I & t \end{bmatrix} \bar{x}$$



2D Transformations



Homogenous vector

Example: translation

$$x' = x + t \qquad x' = \begin{bmatrix} I & t \end{bmatrix} \bar{x} \qquad \bar{x}' = \begin{bmatrix} I & t \\ 0^T & 1 \end{bmatrix} \bar{x}$$

$$\begin{bmatrix} \cdot \\ \cdot \end{bmatrix} = \begin{bmatrix} \cdot \\ \cdot \end{bmatrix} + \begin{bmatrix} tx \\ ty \end{bmatrix} \qquad \begin{bmatrix} \cdot \\ \cdot \end{bmatrix} = \begin{bmatrix} 1 & 0 & tx \\ 0 & 1 & ty \end{bmatrix} \cdot \begin{bmatrix} \cdot \\ \cdot \\ 1 \end{bmatrix} \qquad \begin{bmatrix} \cdot \\ \cdot \\ \cdot \end{bmatrix} = \begin{bmatrix} 1 & 0 & tx \\ 0 & 1 & ty \\ 0 & 0 & 1 \end{bmatrix} \cdot \begin{bmatrix} \cdot \\ \cdot \\ \cdot \end{bmatrix}$$

[BTW: Now we can chain transformations]

Projective Transformations

- *Projective* transformations: for 2D images it's a 3x3 matrix applied to homogenous coordinates

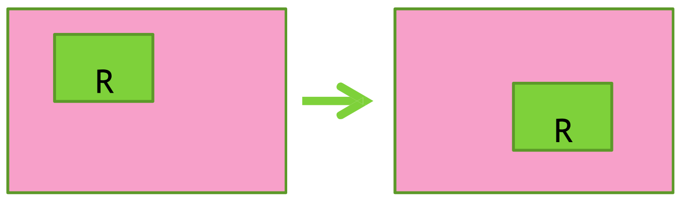
$$\begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \begin{bmatrix} x \\ y \\ w \end{bmatrix}$$

Special Projective Transformations

- Translation

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} 1 & 0 & t_x \\ 0 & 1 & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

- Preserves:
  - Lengths/Areas
  - Angles
  - Orientation
  - Lines



Quiz 1

Suppose I told you the transform from image A to image B is a ***translation***. How many pairs of corresponding points would you need to know to compute the transformation?

a) 3

b) 1

c) 2

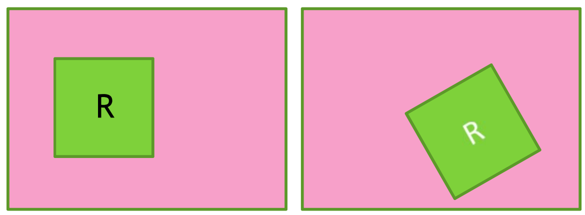
d) 4

Special Projective Transformations

- Euclidean (Rigid body)

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & t_x \\ \sin(\theta) & \cos(\theta) & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

- Preserves:
  - Lengths/Areas
  - Angles
  - Lines

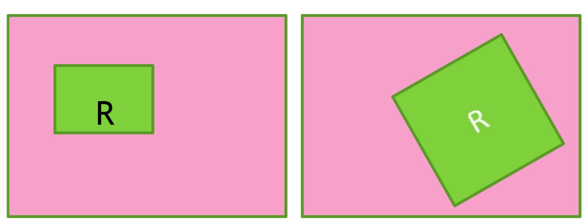


Special Projective Transformations

- Similarity (trans, rot, scale) transform

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a\cos(\theta) & -a\sin(\theta) & t_x \\ a\sin(\theta) & a\cos(\theta) & t_y \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

- Preserves:
  - Ratios of Areas
  - Angles
  - Lines

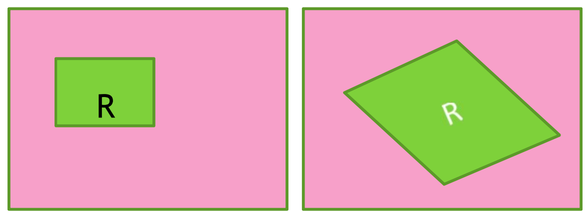


Special Projective Transformations

- Affine transform

$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

- Preserves:
  - Parallel lines
  - Ratio of Areas
  - Lines



Quiz2

Suppose I told you the transform from image A to image B is ***affine***. How many pairs of corresponding points would you need to know to compute the transformation?

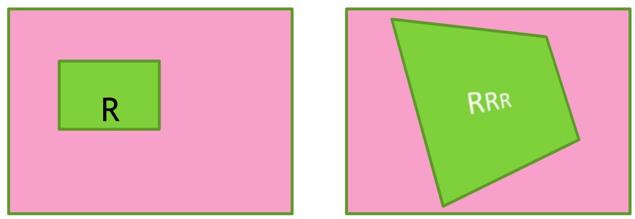
a) 3
b) 1
c) 2
d) 4

Projective Transformations

- General projective transform (or Homography)

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

- Preserves:
  - Lines
  - Also cross ratios (maybe later)



Projective Transformations

- Remember, these are homogeneous coordinates

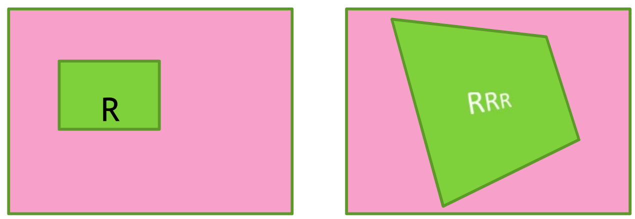
$$\begin{bmatrix} x' \\ y' \\ 1 \end{bmatrix} \cong \begin{bmatrix} sx' \\ sy' \\ s \end{bmatrix} = \begin{bmatrix} a & b & c \\ d & e & f \\ g & h & i \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix}$$

Projective Transformations

- General projective transform (or Homography)

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

- Preserves:
  - Lines
  - Also cross ratios (maybe later)



Quiz3

Suppose I told you the transform from image A to image B is a ***homography***. How many pairs of corresponding points would you need to know to compute the transformation?

a) 3
b) 8
c) 2
d) 4

Homographies and mosaics

Projective Transformations

Projective transformations: for 2D images it's a 3x3 matrix applied to homogenous coordinates

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

Application: Simple mosaics

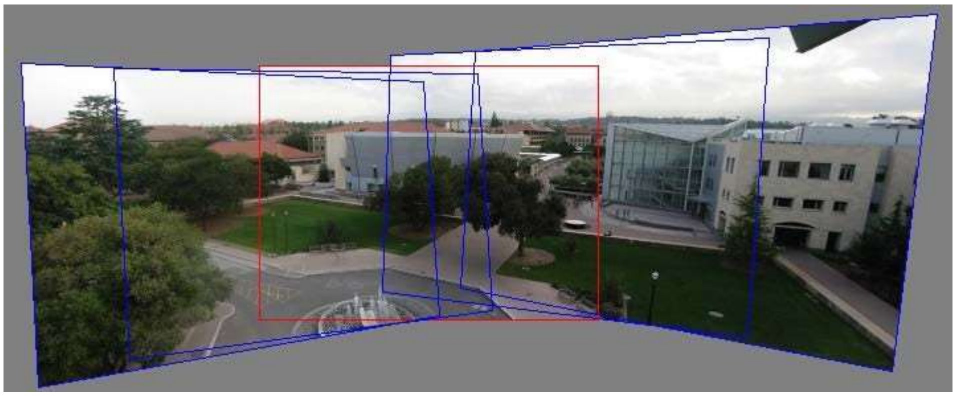


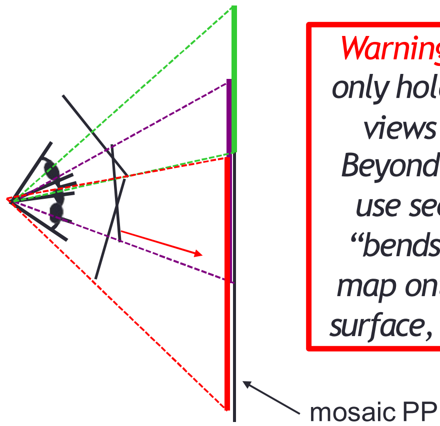
Image from http://graphics.cs.cmu.edu/courses/15-463/2010_fall/

How to stitch together a panorama (a.k.a. mosaic)?

- 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)

Source: Steve Seitz

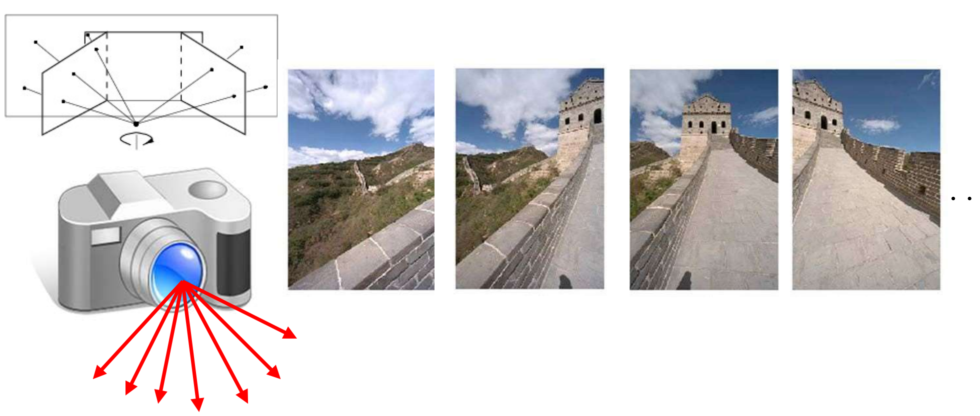
Image reprojection



Warning: This model only holds for angular views up to 180°. Beyond that need to use sequence that "bends the rays" or map onto a different surface, say, a cylinder.

- The mosaic has a natural interpretation in 3D
  - The images are reprojected onto a common plane
  - The mosaic is formed on this plane

Mosaics

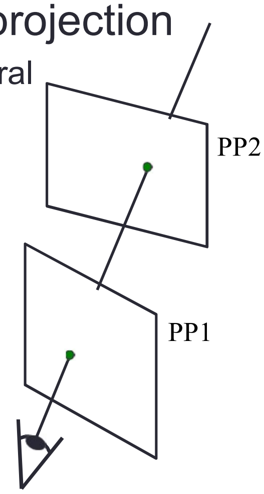


Obtain a wider angle view by combining multiple images ***all of which are taken from the same camera center***.

Image reprojection: Homography

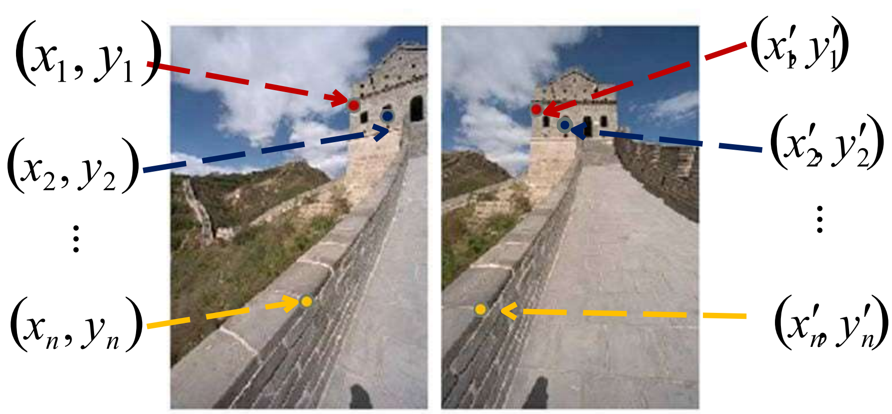
- A projective transform is a mapping between any two PPs with the same center of projection
  - rectangle should map to arbitrary quadrilateral
  - parallel lines aren't
  - but must preserve straight lines
  - called **Homography**

$$\underbrace{\begin{bmatrix} wx' \\ wy' \\ w \end{bmatrix}}_{\mathbf{p'}} = \underbrace{\begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \end{bmatrix}}_{\mathbf{H}} \underbrace{\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}}_{\mathbf{p}}$$



Source: Alyosha Efros

Homography



To **compute** the homography given pairs of corresponding points in the images, we need to set up an equation where the parameters of **H** are the unknowns…

Solving for homographies

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

- Can set scale factor i=1. So, there are 8 unknowns.
- Set up a system of linear equations $\mathbf{Ah} = \mathbf{b}$
- where vector of unknowns h = [a,b,c,d,e,f,g,h]T
- Need at least 4 points for 8 eqs, but the more the better…
- Solve for h. If overconstrained, solve using least-squares:

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

- Look familiar? (If don't set $\boldsymbol{i}$ to 1 can use SVD)

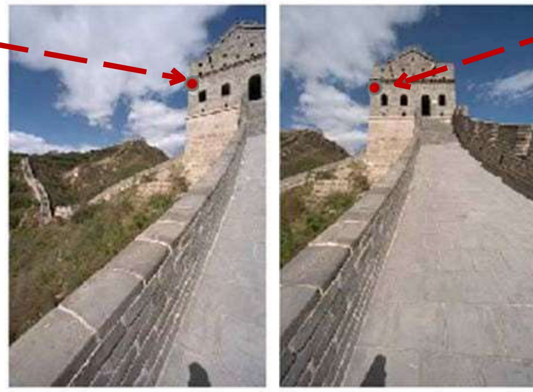
Solving for homographies – homogeneous

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

Just like we did for the extrinsics, multiply through, and divide out by w. Gives two homogeneous equations per point.
Solve using SVD just like before. This is the cool way.

Apply Homography

(x, y)



$\left({}^{wx'}/_{w}, {}^{wy'}/_{w} \right)$

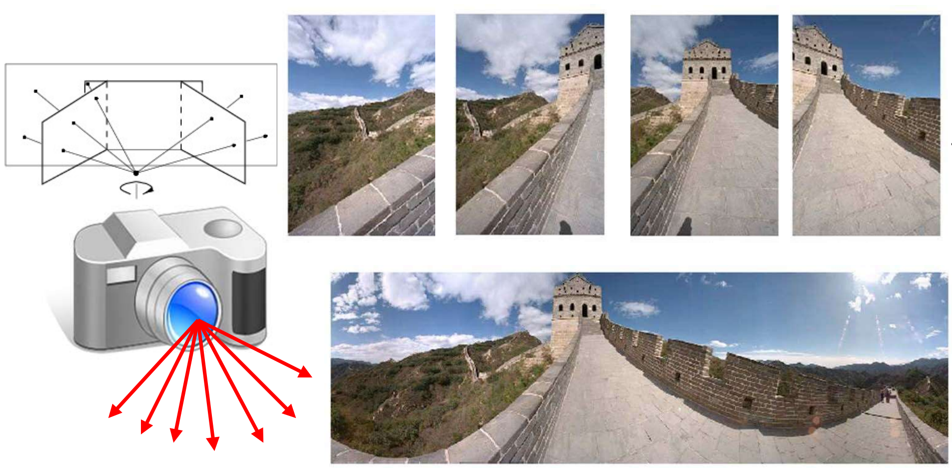
$= (x', y')$

To **apply** a given homography **H**

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

$$\underbrace{\begin{bmatrix} wx' \\ wy' \\ w \end{bmatrix}}_{\mathbf{p'}} = \underbrace{\begin{bmatrix} * & * & * \\ * & * & * \\ * & * & * \end{bmatrix}}_{\mathbf{H}} \underbrace{\begin{bmatrix} x \\ y \\ 1 \end{bmatrix}}_{\mathbf{p}}$$

Mosaics



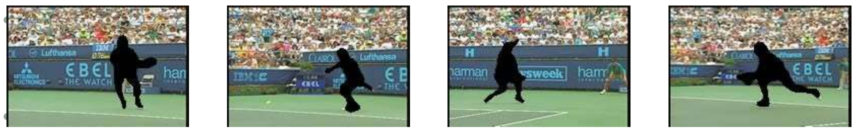
. . .

image from S. Seitz

Combine images with the computed homographies…

Mosaics for Video Coding

- Convert masked images into a background sprite for content-based coding



=

Homographies and 3D planes

- Remember this:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} m_{00} & m_{01} & m_{02} & m_{03} \\ m_{10} & m_{11} & m_{12} & m_{13} \\ m_{20} & m_{21} & m_{22} & m_{23} \end{bmatrix} \begin{bmatrix} X \\ Y \\ Z \\ 1 \end{bmatrix}$$

- Suppose the 3D points are on a plane:

$$aX + bY + cZ + d = 0$$

Homographies and 3D planes

- On the plane *[a b c d]* can replace Z:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} m_{00} & m_{01} & m_{02} & m_{03} \\ m_{10} & m_{11} & m_{12} & m_{13} \\ m_{20} & m_{21} & m_{22} & m_{23} \end{bmatrix} \begin{bmatrix} X \\ Y \\ (aX + bY + d)/(-c) \\ 1 \end{bmatrix}$$

This column multiplies X, Y and a constant term, so its effect can be represented using the other columns

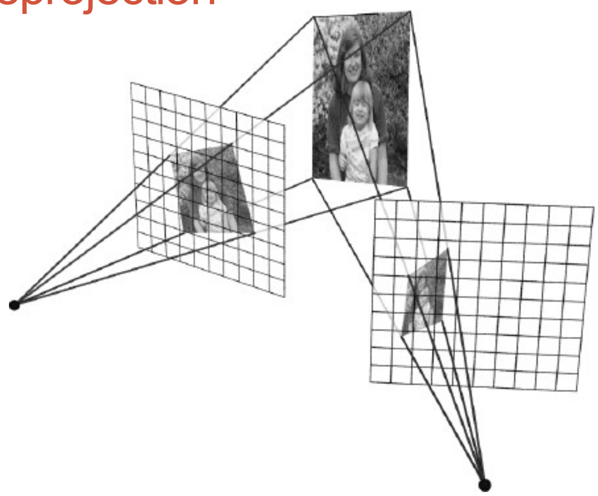
Homographies and 3D planes

- So, can put the Z coefficients into the others:

$$\begin{bmatrix} u \\ v \\ 1 \end{bmatrix} = \begin{bmatrix} m'_{00} & m'_{01} & 0 & m'_{03} \\ m'_{10} & m'_{11} & 0 & m'_{13} \\ m'_{20} & m'_{21} & 0 & m'_{23} \end{bmatrix} \begin{bmatrix} X \\ Y \\ (aX + bY + d)/(-c) \\ 1 \end{bmatrix}$$

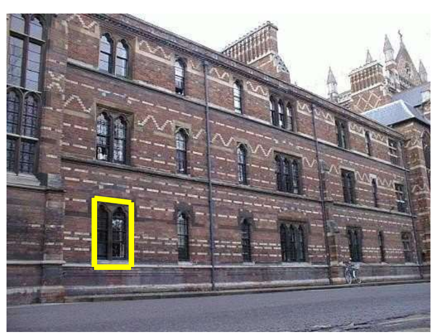
3x3
Homography!

Image reprojection



- Mapping between planes is a homography. Whether a plane in the world to the image or between image planes.

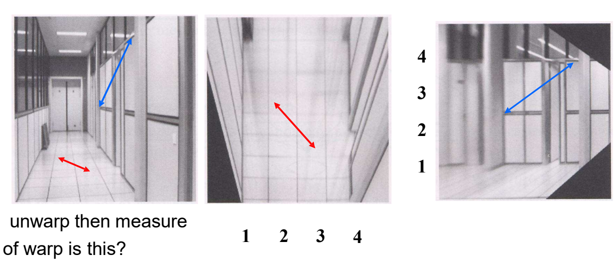
Rectifying slanted views



Rectifying slanted views

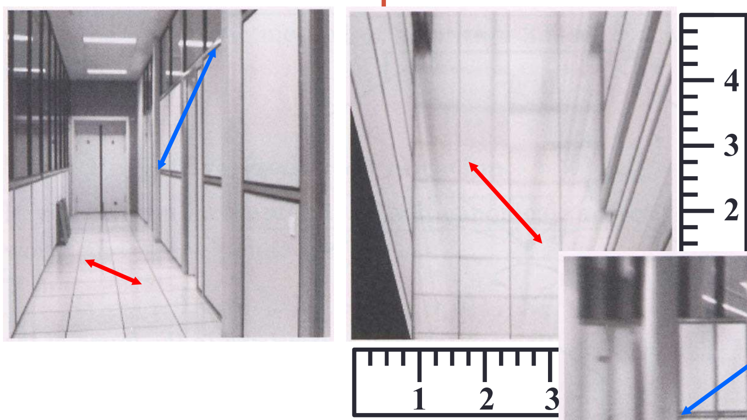
front-to-parallel

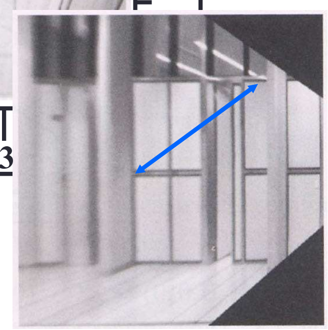
Measurements on planes



Approach: unwarp then measure
What kind of warp is this?
Homography…

Measurements on planes



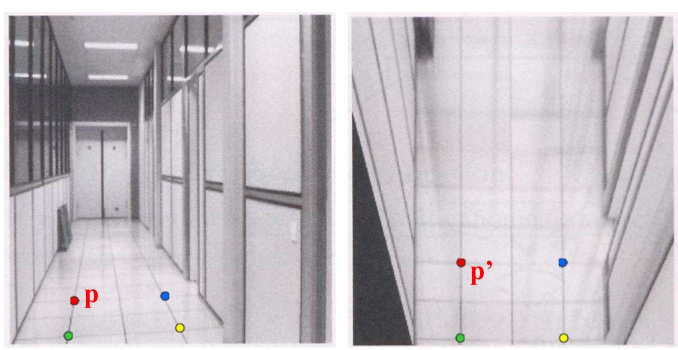


Approach: unwarp then measure

What kind of warp is this?

Homography…

Image rectification



A planar rectangular grid in the scene. Map it into a rectangular grid in the image.

Some other images of rectangular grids…

Same pixels – via a homography

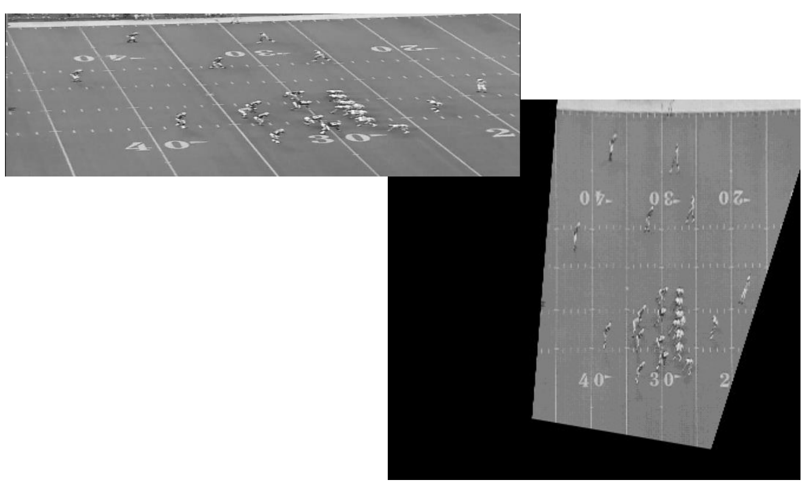
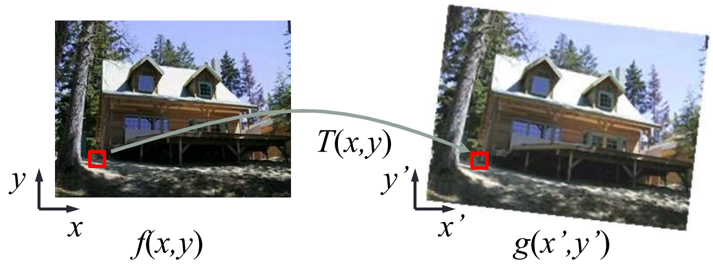


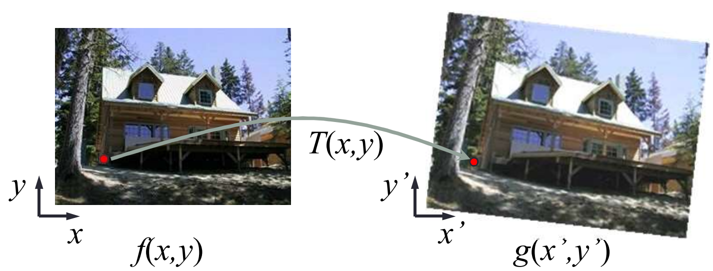
Image warping



Given a coordinate transform and a source image $f(x,y)$, how do we compute a transformed image $g(x',y') = f(T(x,y))$?

Slide from Alyosha Efros, CMU

Forward warping

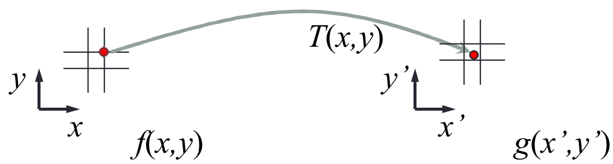


- Send each pixel $f(x,y)$ to its corresponding location
- $(x',y') = T(x,y)$ in the second image

Q: what if pixel lands "between" two pixels?

Slide from Alyosha Efros, CMU

Forward warping



- Send each pixel $f(x,y)$ to its corresponding location
- $(x',y') = T(x,y)$ in the second image

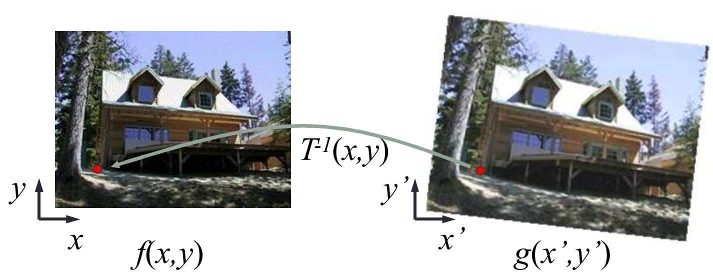
Q: what if pixel lands "between" two pixels?

A: distribute color among neighboring pixels (x',y')

- Known as "splatting"

Slide from Alyosha Efros, CMU

Inverse warping



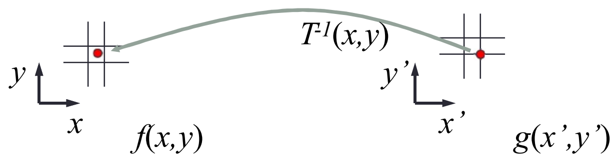
Get each pixel $g(x',y')$ from its corresponding location

$(x,y) = T^{-1}(x',y')$ in the first image

Q: what if pixel comes from "between" two pixels?

Slide from Alyosha Efros, CMU

Inverse warping



Get each pixel $g(x',y')$ from its corresponding location

$(x,y) = T^{-1}(x',y')$ in the first image

Q: what if pixel comes from "between" two pixels?

A: *Interpolate* color value from neighbors

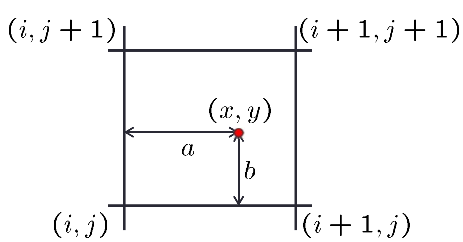
– nearest neighbor, bilinear…

Slide from Alyosha Efros, CMU

```
>> help interp2
```

Bilinear interpolation

Sampling at *f(x,y):*



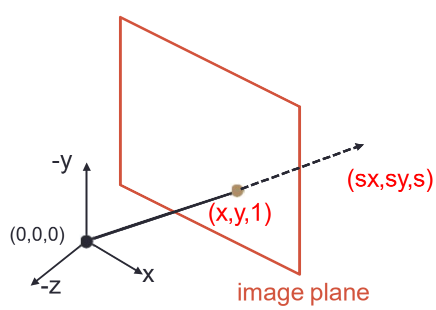
$$
\begin{aligned}
f(x,y) = \; & (1-a)(1-b) \quad f[i,j] \\
& +a(1-b) \quad f[i+1,j] \\
& +ab \quad f[i+1,j+1] \\
& +(1-a)b \quad f[i,j+1]
\end{aligned}
$$

Slide from Alyosha Efros, CMU

Projective geometry

Recall: The projective plane

- What is the geometric intuition of using homogenous coordinates ?
  - a point in the image is a *ray* in projective space



- Each *point* (x,y) on the plane is represented by a *ray* (sx,sy,s)
  - all points on the ray are equivalent: (x, y, 1) ≅ (sx, sy, s)

Homogeneous coordinates

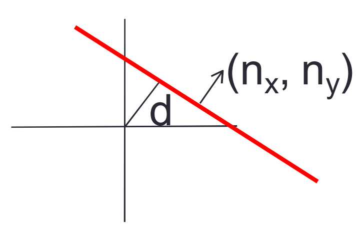
2D Points:

$$p = \begin{bmatrix} x \\ y \end{bmatrix} \longrightarrow p' = \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} \qquad p' = \begin{bmatrix} x' \\ y' \\ w' \end{bmatrix} \longrightarrow p = \begin{bmatrix} x'/w' \\ y'/w' \end{bmatrix}$$

Homogeneous coordinates

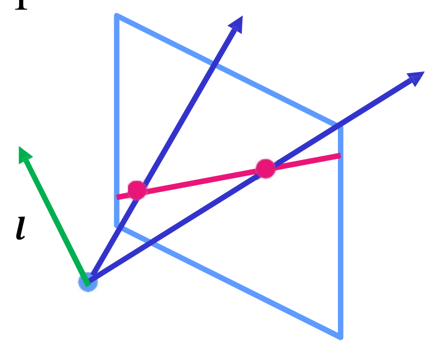
2D Lines: $ax + by + c = 0$

$$\begin{bmatrix} a & b & c \end{bmatrix} \begin{bmatrix} x \\ y \\ 1 \end{bmatrix} = 0 \qquad l = \begin{bmatrix} a & b & c \end{bmatrix} \Rightarrow \begin{bmatrix} n_x & n_y & -d \end{bmatrix}$$



Projective lines

What does a line in the image correspond to in projective space?

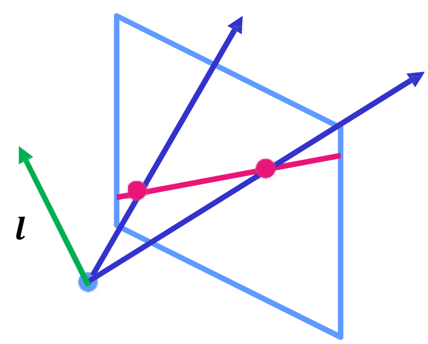


Projective lines

A line is a *plane* of rays through origin define by the normal $l = (a, b, c)$

All rays (x,y,z) satisfying:

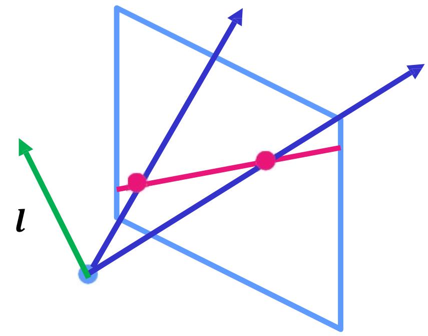
$$ax + by + cz = 0$$



Projective lines

In vector notation:

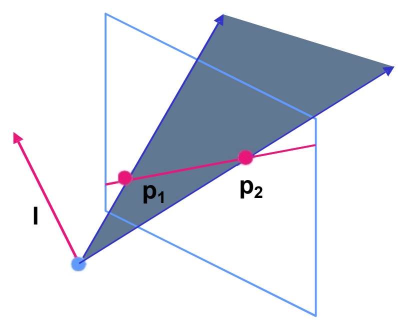
$$0 = \underbrace{\begin{bmatrix} a & b & c \end{bmatrix}}_{l} \underbrace{\begin{bmatrix} x \\ y \\ z \end{bmatrix}}_{p}$$



A line is also represented as a homogeneous 3-vector!

Point and line duality

- A line $\mathbf{l}$ is a homogeneous 3-vector
- It is $\perp$ to every point (ray) **p** on the line: $\mathbf{l}^T\mathbf{p}=0$
  i.e., perpendicular to the ray that defines that point



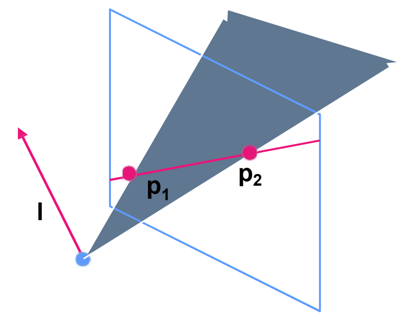
Point and line duality

What is the line $\mathbf{l}$ spanned by rays $\mathbf{p}_1$ and $\mathbf{p}_2$?

$\mathbf{l}$ is $\perp$ to $\mathbf{p}_1$ and $\mathbf{p}_2$

$\Rightarrow \quad \mathbf{l} = \mathbf{p}_1 \times \mathbf{p}_2$

$\mathbf{l}$ is the plane normal

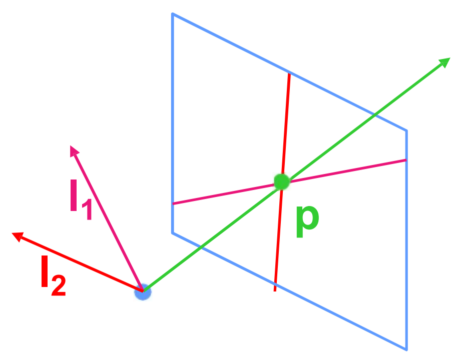


Point and line duality

What is the intersection of two lines $\mathbf{l}_1$ and $\mathbf{l}_2$?

$\mathbf{p}$ is $\perp$ to $\mathbf{l}_1$ and $\mathbf{l}_2 \quad \Rightarrow$

$\mathbf{p} = \mathbf{l}_1 \times \mathbf{l}_2$

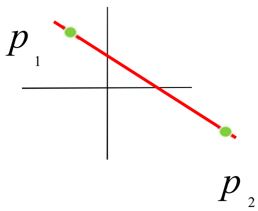


Points and lines are *dual* in projective space

- Given any formula, can switch the meanings of points and lines to get another formula

Homogeneous coordinates

Line joining two points:



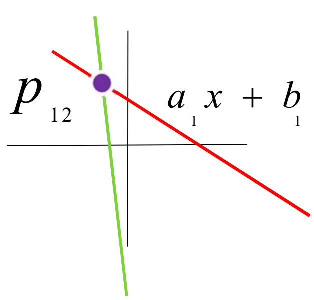
$ax + by + c = 0$

$$p_1 = [x_1 \quad y_1 \quad 1]$$
$$p_2 = [x_2 \quad y_2 \quad 1]$$

$$l = p_1 \times p_2$$

Homogeneous coordinates

Intersection between two lines:



$a_1 x + b_1 y + c_1 = 0$

$a_2 x + b_2 y + c_2 = 0$

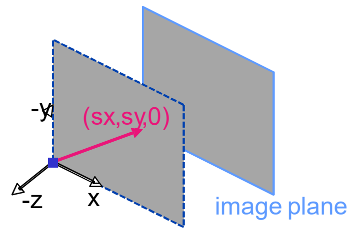
$$l_1 = [a_1 \quad b_1 \quad c_1]$$
$$l_2 = [a_2 \quad b_2 \quad c_2]$$

$$p_{12} = l_1 \times l_2$$

Ideal points and lines

Ideal point ("point at infinity")

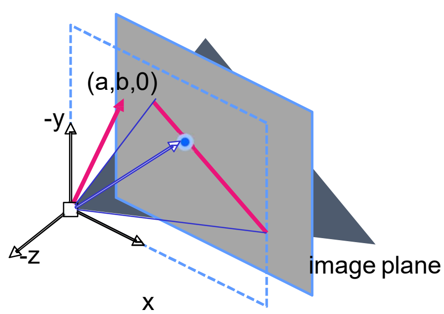
- $p \cong (x, y, 0)$ - ray parallel to image plane → intersection occurs at infinity!
- It has infinite image coordinates



Ideal points and lines

Ideal line

- $l \cong (a, b, 0)$ – normal is parallel to image plane
- Corresponds to a line in the image (finite coordinates)
  - –goes through image origin (*principle point*)



3D projective geometry

- These concepts generalize naturally to 3D
- Recall the equation of a plane:

$$aX + bY + cZ + d = 0$$

- Homogeneous coordinates

  Projective 3D points have four coords: p = (wX,wY,wZ,w)

3D projective geometry

- Duality
  - A plane N is also represented by a 4-vector N = (a,b,c,d)
  - Points and planes are dual in 3D: $N^T p = 0$
- Projective transformations
  - Represented by 4x4 matrices T: P' = TP