CS4670: Computer Vision

Noah Snavely

Lecture 14: Panoramas

What's inside your fridge?
http://www.cs.washington.edu/education/courses/cse590ss/01wi/

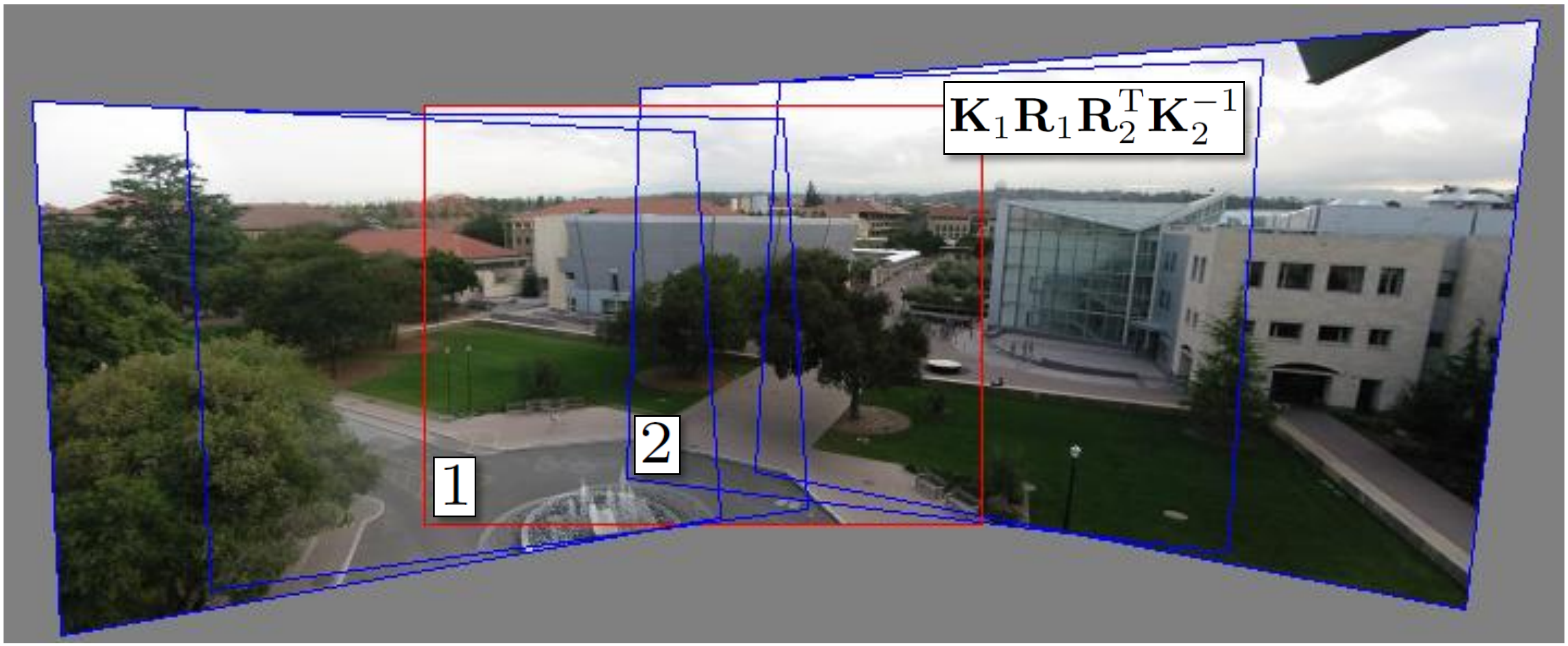
Reading

- Szeliski Chapter 9

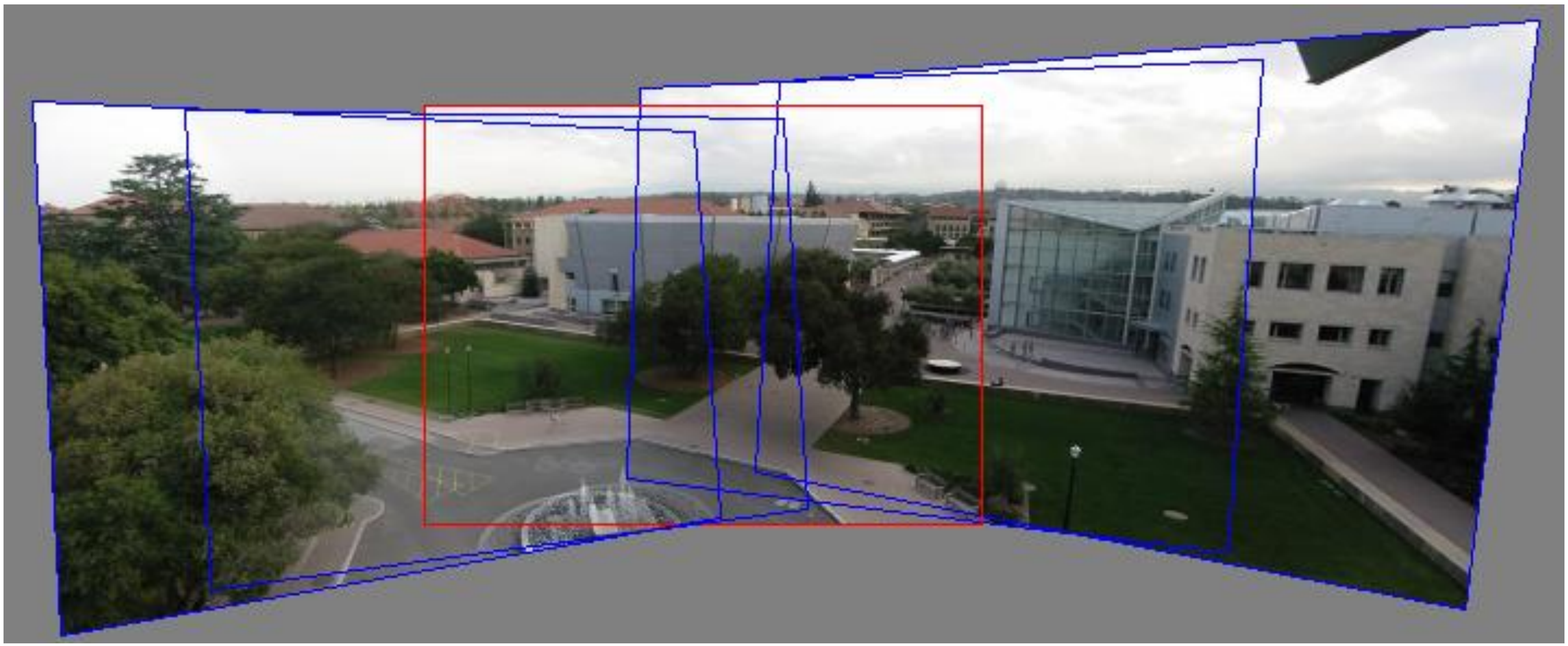
Announcements

- Project 2a due today, 8:59pm
- Project 2b out today
- Take-home prelim after Fall break

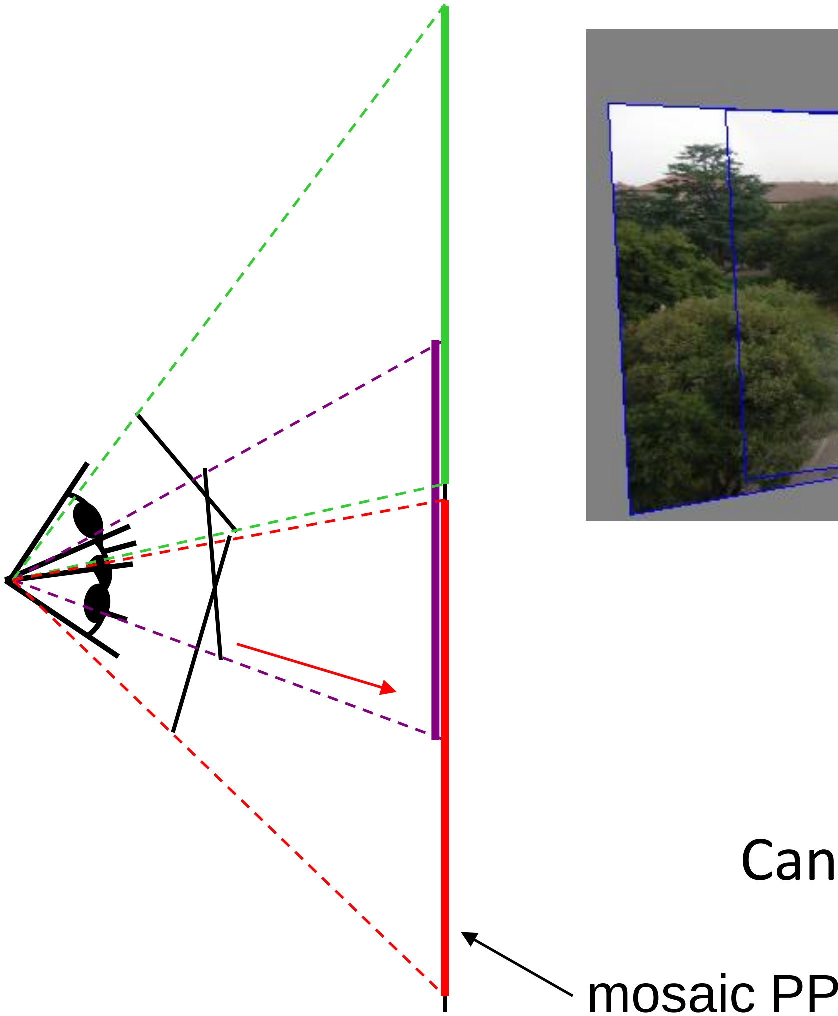
Image alignment

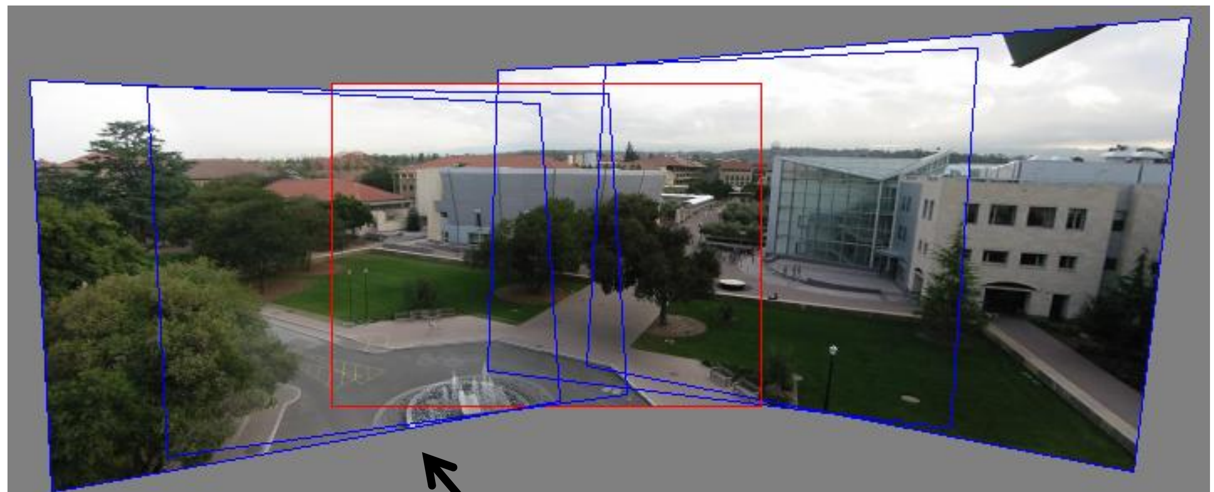


Can we use homography to create a 360 panorama?



Last time: projecting images onto a common plane



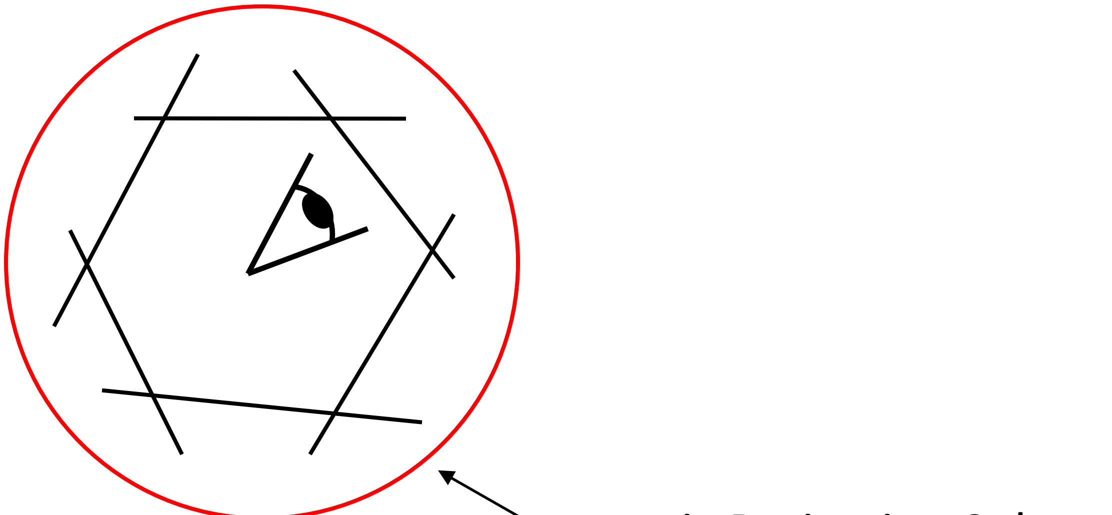


each image is warped with a homography $\mathbf{H}$

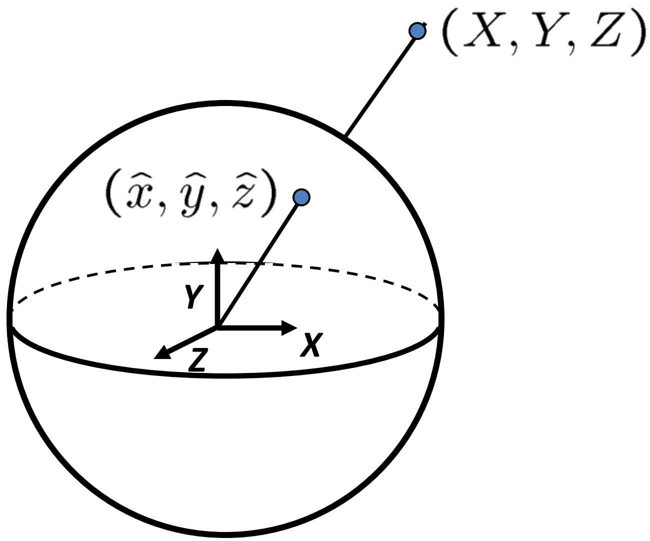
Can’t create a 360 panorama this way…

Panoramas

- What if you want a 360° field of view?



Spherical projection



unit sphere

- Map 3D point (X,Y,Z) onto sphere

$$(\hat{x},\hat{y},\hat{z}) = \frac{1}{\sqrt{X^2+Y^2+Z^2}}(X,Y,Z)$$

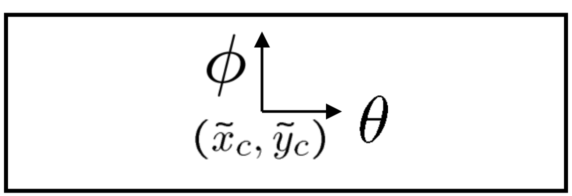
- Convert to spherical coordinates

$$(sin\theta cos\phi, sin\phi, cos\theta cos\phi) = (\hat{x},\hat{y},\hat{z})$$

- Convert to spherical image coordinates

$$(\tilde{x},\tilde{y}) = (s\theta, s\phi) + (\tilde{x}_c,\tilde{y}_c)$$

  - s defines size of the final image
    - » often convenient to set s = camera focal length



unwrapped sphere

Spherical image

Spherical reprojection

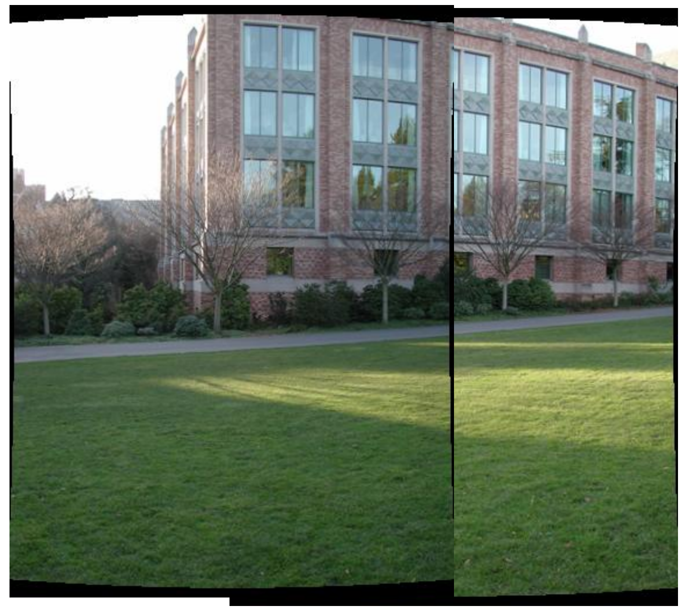
input **f = 200 (pixels)** **f = 400** **f = 800**

- Map image to spherical coordinates
  - need to know the focal length

Aligning spherical images

- Suppose we rotate the camera by θ about the vertical axis
  - How does this change the spherical image?

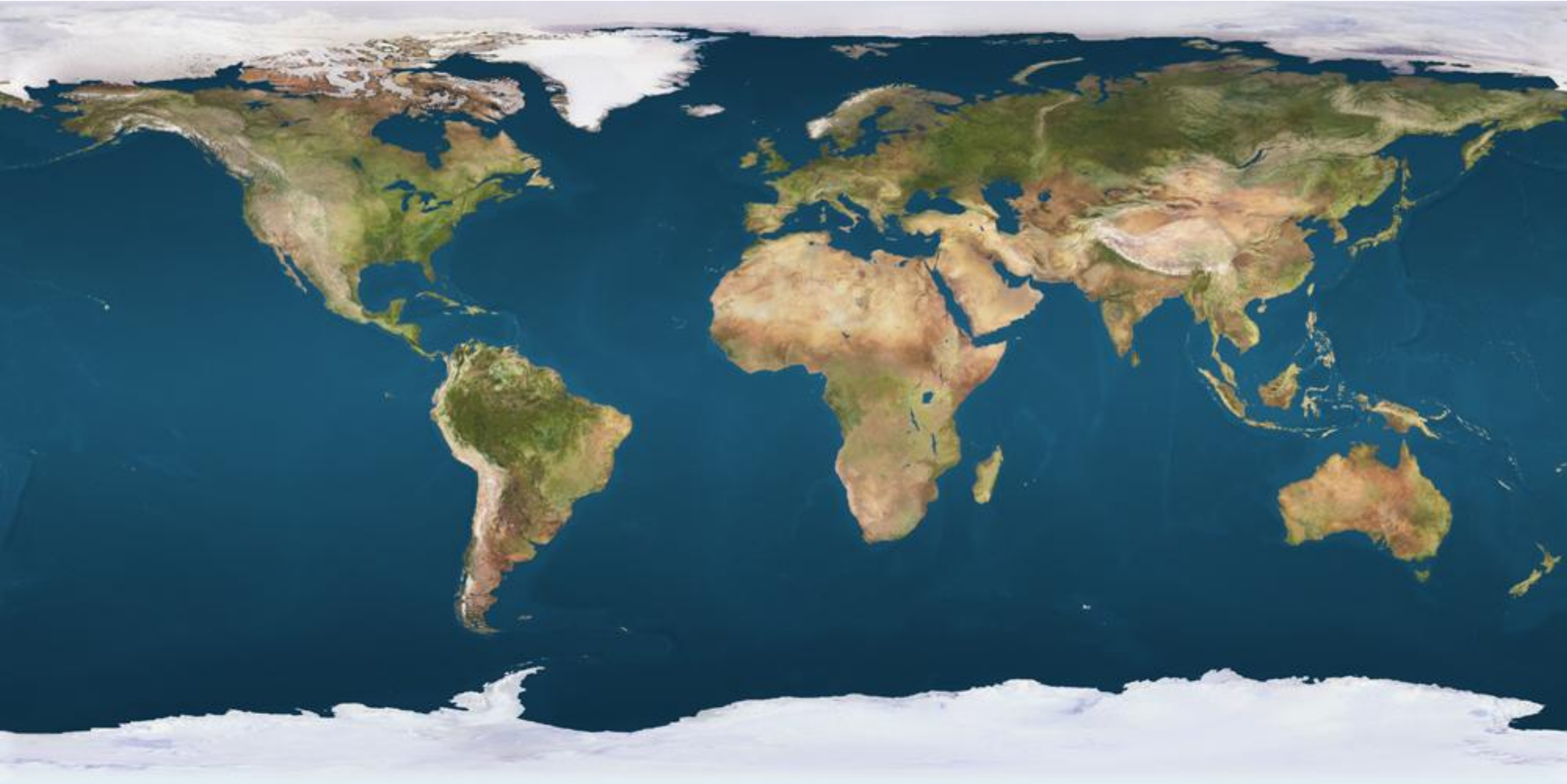
Aligning spherical images



- Suppose we rotate the camera by θ about the vertical axis
  - How does this change the spherical image?
    - Translation by θ
  - This means that we can align spherical images by translation

Unwrapping a sphere

Credit: JHT's Planetary Pixel Emporium



Spherical panoramas

Microsoft Lobby: http://www.acm.org/pubs/citations/proceedings/graph/258734/p251-szeliski

Different projections are possible

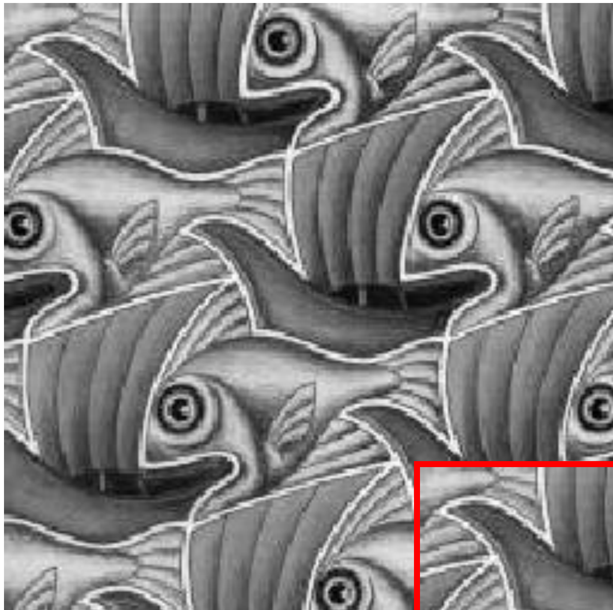
Blending

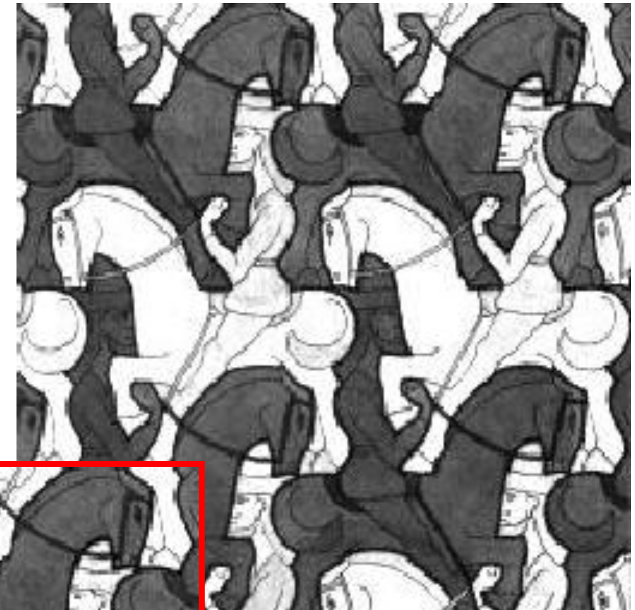
- We’ve aligned the images – now what?

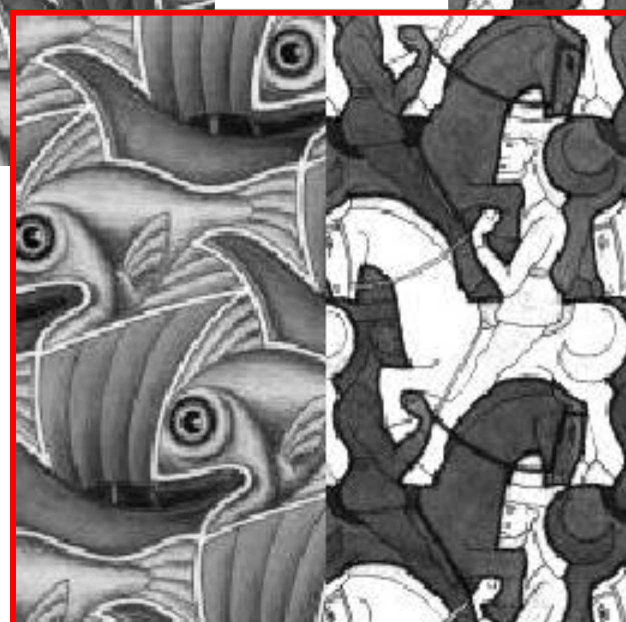
Blending

- Want to seamlessly blend them together

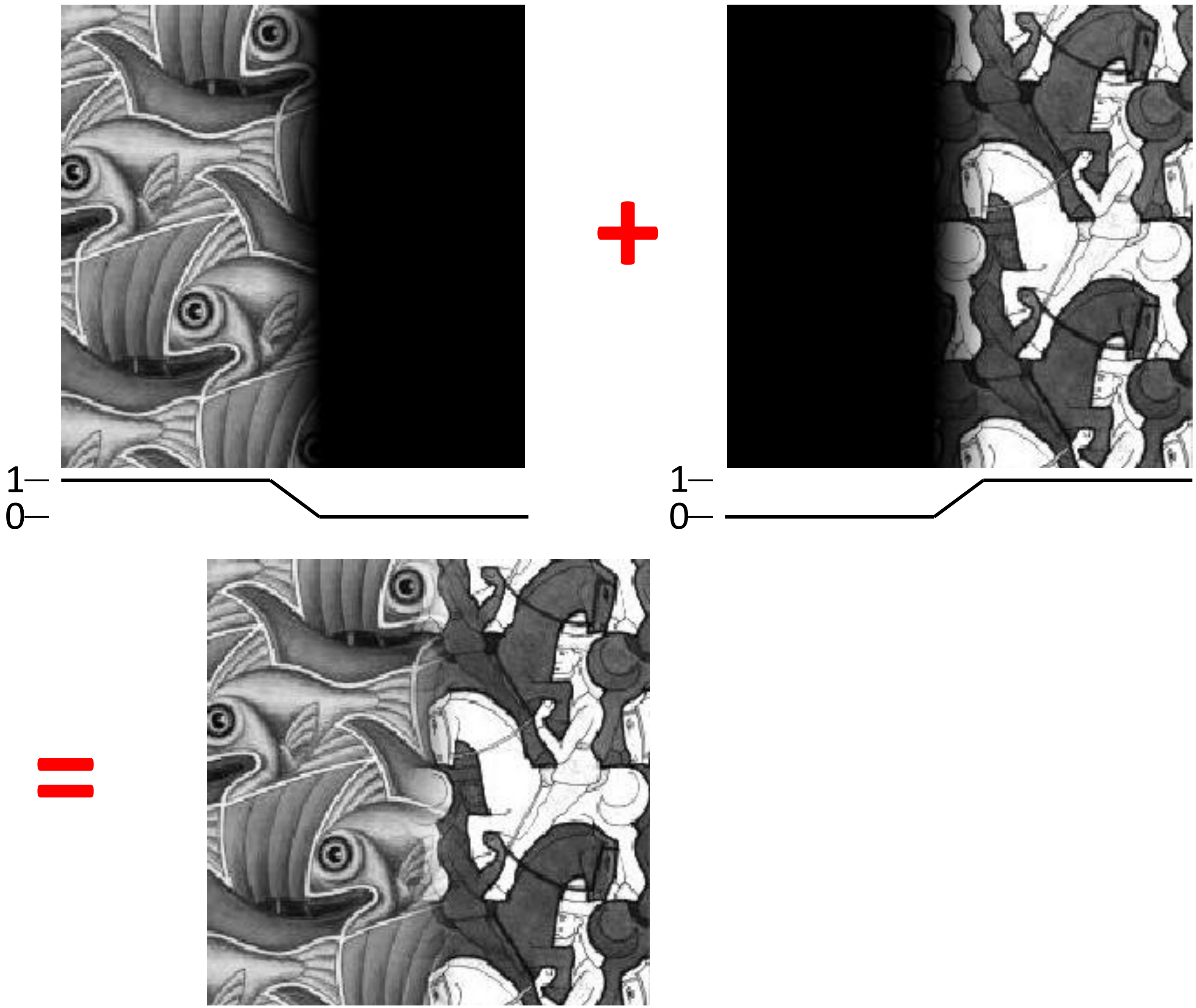
Image Blending



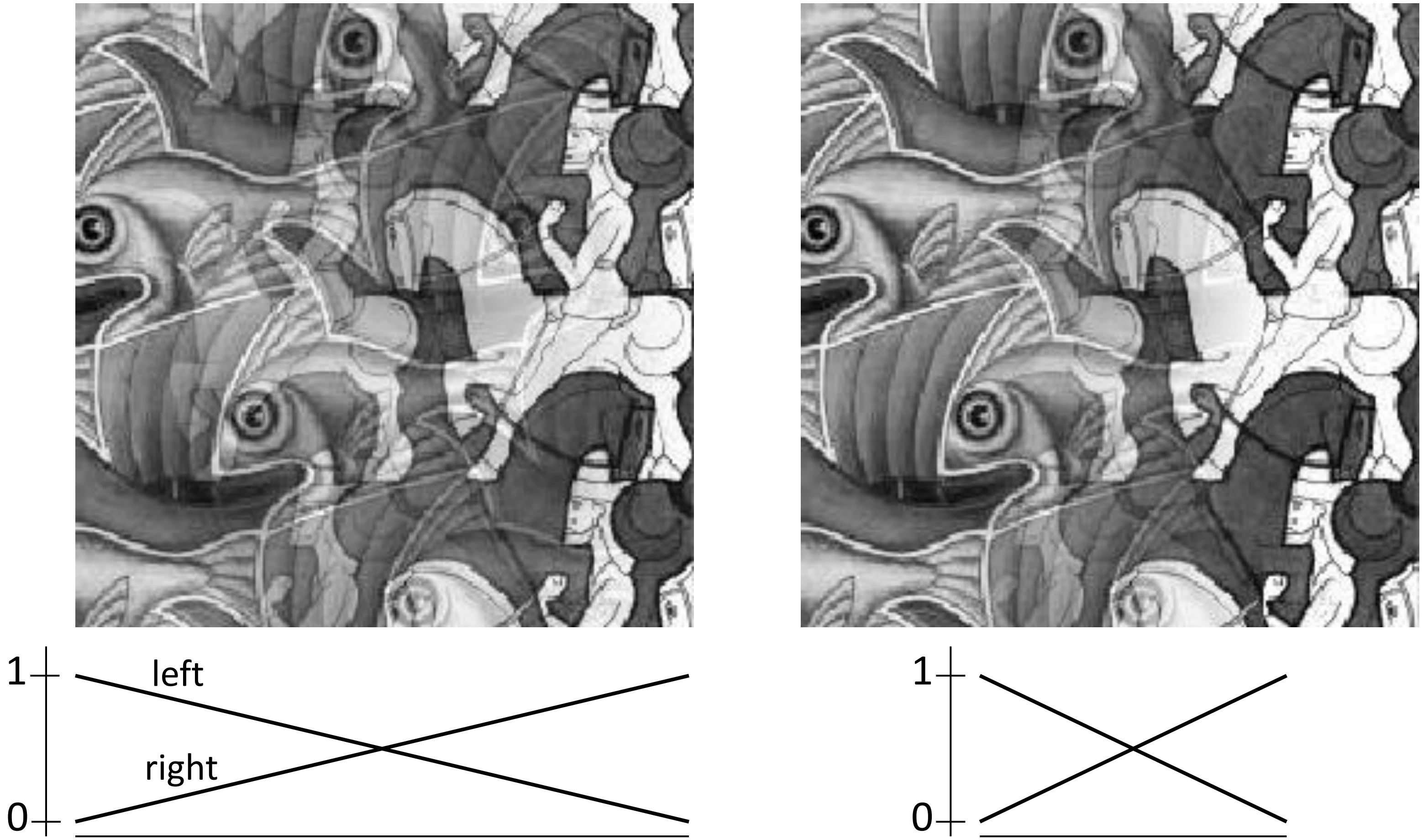




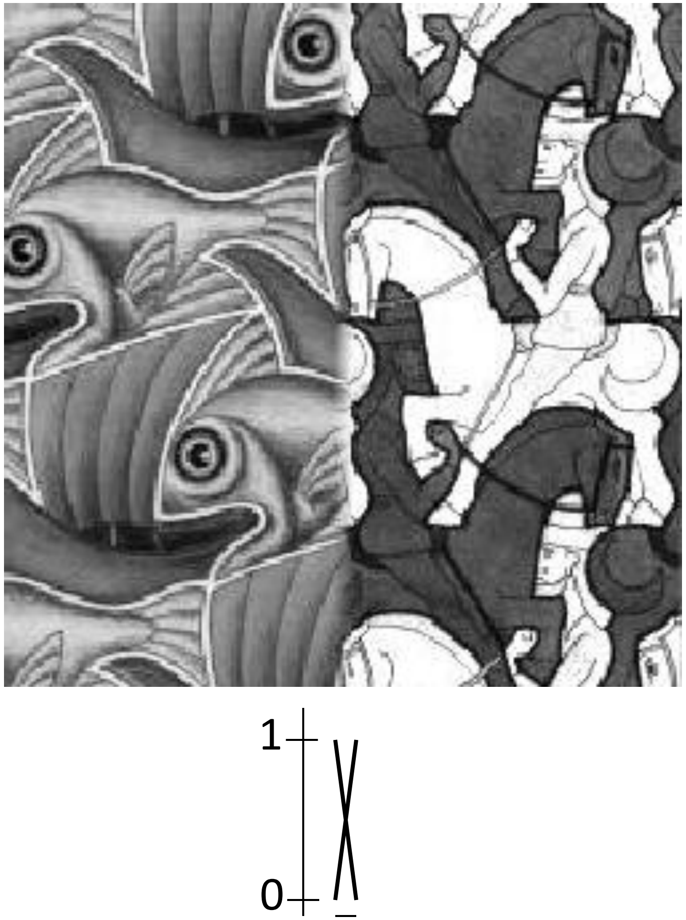
Feathering

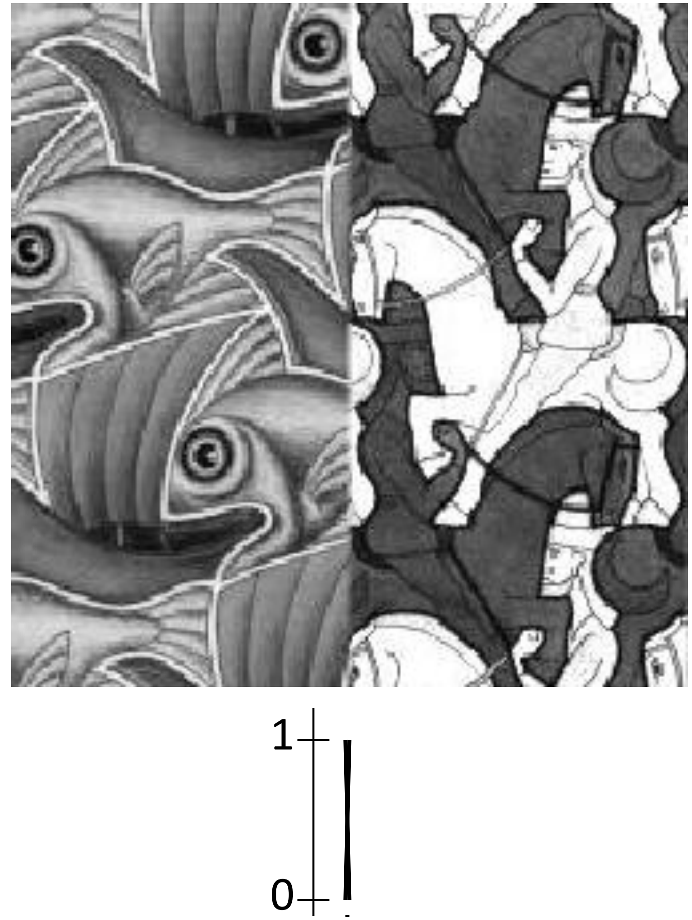


Effect of window size

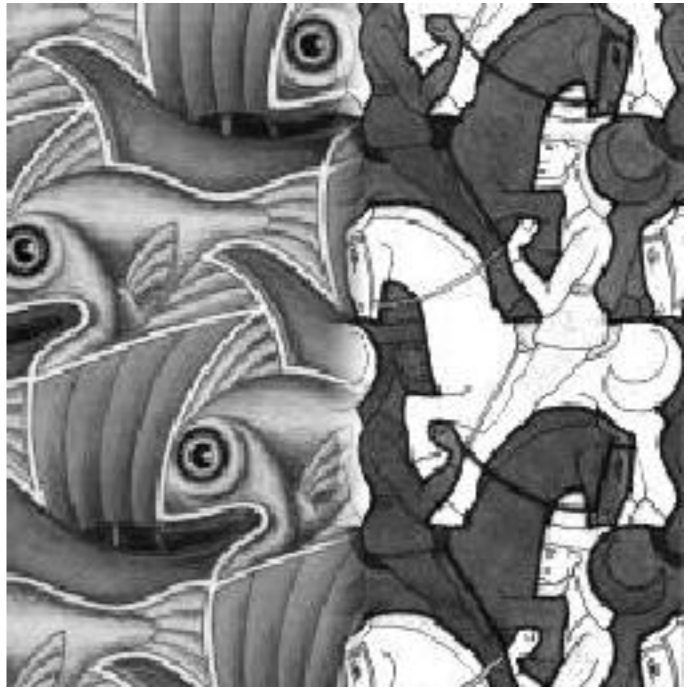


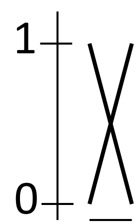
Effect of window size





Good window size

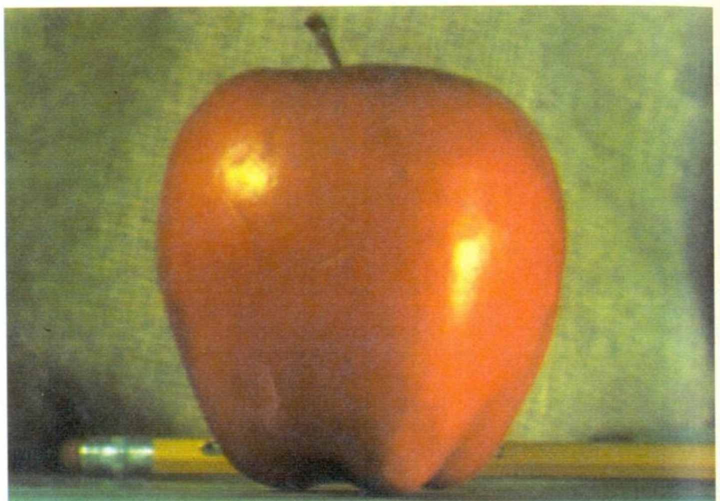




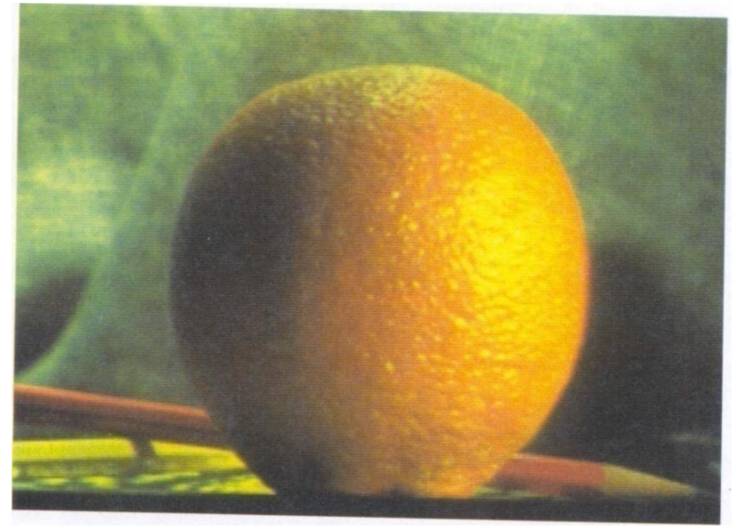
"Optimal" window: smooth but not ghosted

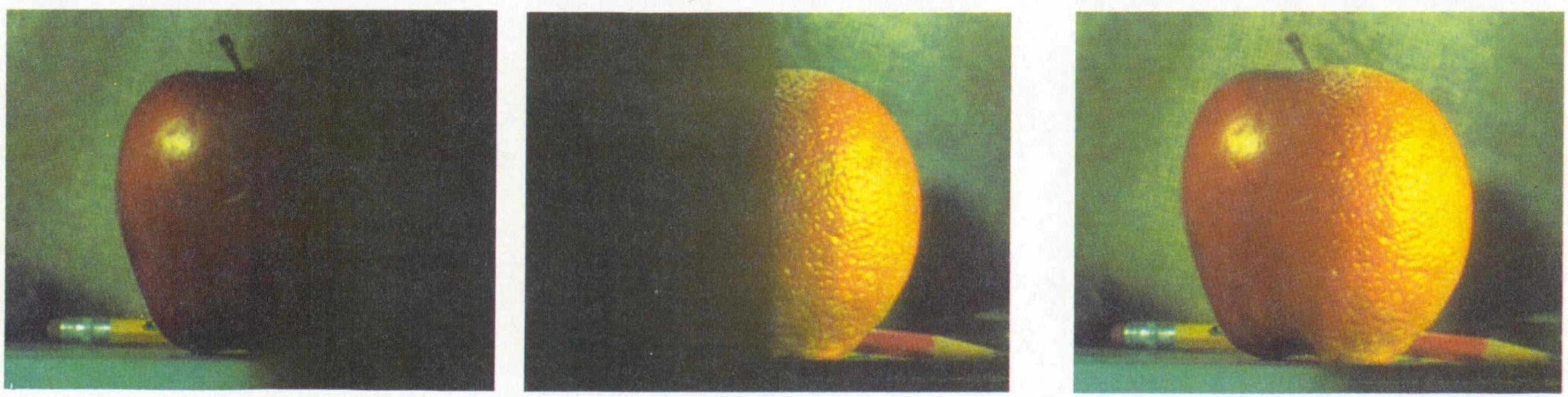
- Doesn't always work...

Pyramid blending



(a)





(d) (h) (l)

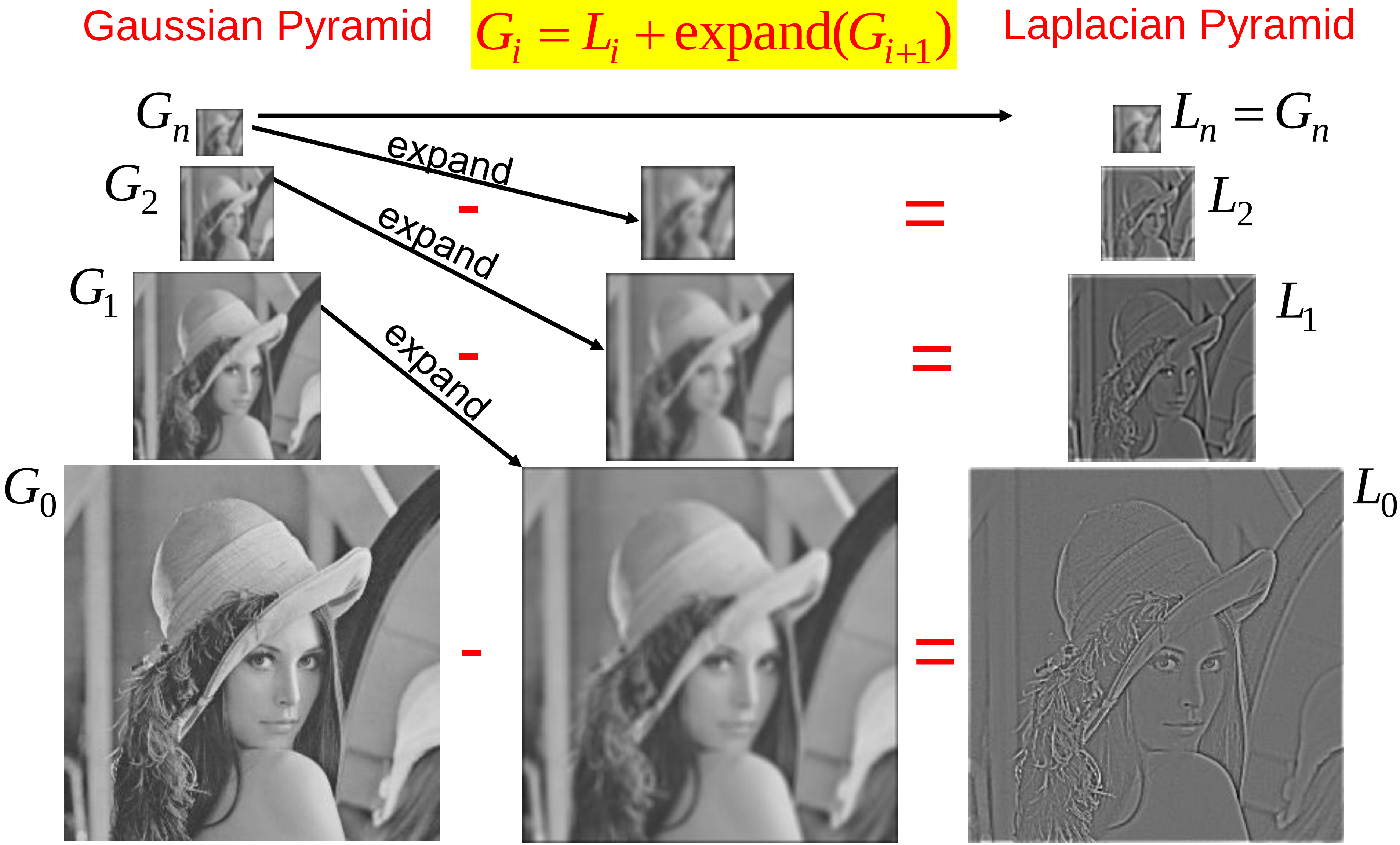
Create a Laplacian pyramid, blend each level

- Burt, P. J. and Adelson, E. H., A multiresolution spline with applications to image mosaics, ACM Transactions on Graphics, 42(4), October 1983, 217-236.

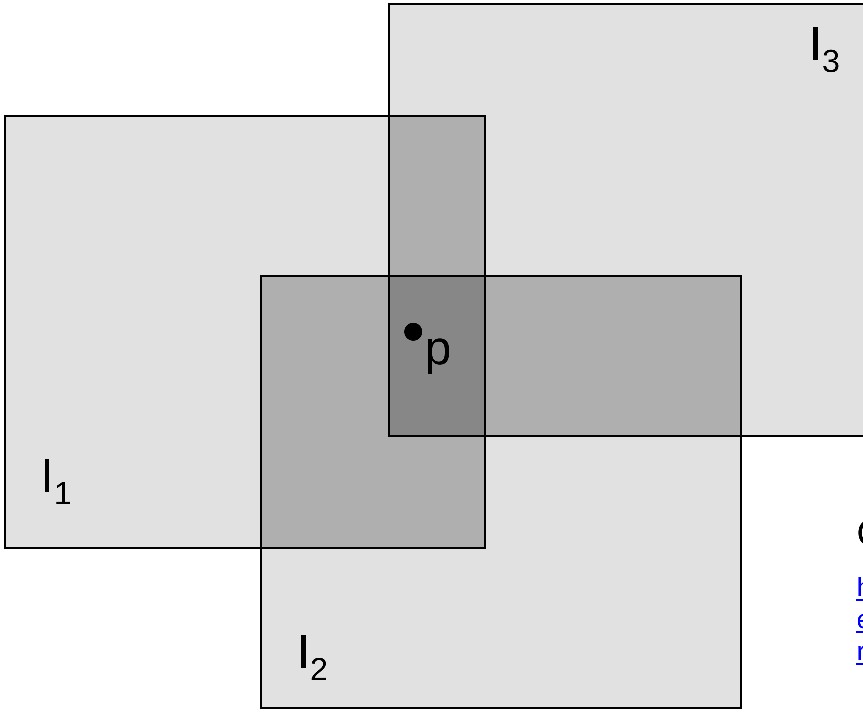
The Laplacian Pyramid

$$L_i = G_i - \text{expand}(G_{i+1})$$

$$G_i = L_i + \text{expand}(G_{i+1})$$



Alpha Blending



Optional: see Blinn (CGA, 1994) for details:

http://ieeexplore.ieee.org/iel1/38/7531/00310740.pdf?isNumber=7531&prod=JNL&arnumber=310740&arSt=83&ared=87&arAuthor=Blinn%2C+J.F.

Encoding blend weights: $I(x,y) = (\alpha R, \alpha G, \alpha B, \alpha)$

color at p = $$\frac{(\alpha_1 R_1,\ \alpha_1 G_1,\ \alpha_1 B_1) + (\alpha_2 R_2,\ \alpha_2 G_2,\ \alpha_2 B_2) + (\alpha_3 R_3,\ \alpha_3 G_3,\ \alpha_3 B_3)}{\alpha_1 + \alpha_2 + \alpha_3}$$

Implement this in two steps:

1. accumulate: add up the (α premultiplied) RGBα values at each pixel
2. normalize: divide each pixel's accumulated RGB by its α value

   Q: what if $\alpha = 0$?

Poisson Image Editing

- For more info: Perez et al, SIGGRAPH 2003
  - http://research.microsoft.com/vision/cambridge/papers/perez_siggraph03.pdf

Some panorama examples

Before Siggraph Deadline:
http://www.cs.washington.edu/education/courses/cse590ss/01wi/projects/project1/students/dougz/siggraph-hires.html

Some panorama examples

- Every image on Google Streetview

Magic: ghost removal

M. Uyttendaele, A. Eden, and R. Szeliski.
***Eliminating ghosting and exposure artifacts in image mosaics*.**
In Proceedings of the Interational Conference on Computer Vision and Pattern Recognition, volume 2, pages 509--516, Kauai, Hawaii, December 2001.

Magic: ghost removal

M. Uyttendaele, A. Eden, and R. Szeliski.
***Eliminating ghosting and exposure artifacts in image mosaics*.**
In Proceedings of the Interational Conference on Computer Vision and Pattern Recognition, volume 2, pages 509--516, Kauai, Hawaii, December 2001.

Other types of mosaics

- Can mosaic onto *any* surface if you know the geometry
  - See NASA's Visible Earth project for some stunning earth mosaics
    - http://earthobservatory.nasa.gov/Newsroom/BlueMarble/
    - Click for images…

Questions?

CS6670: Computer Vision

Noah Snavely

Lecture 14b: Single-view modeling

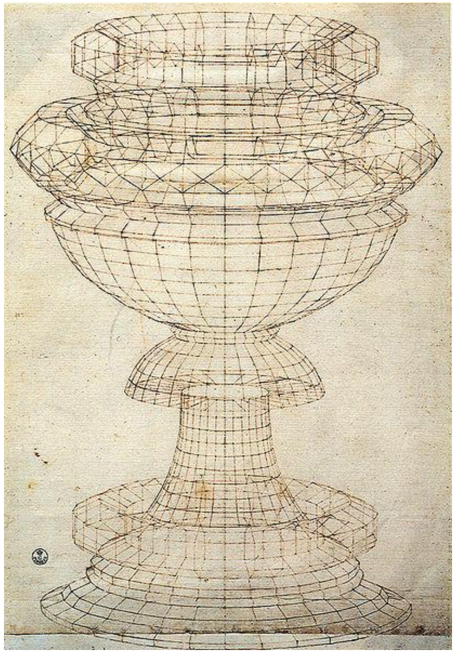
Projective geometry

Ames Room

- Readings
  - Mundy, J.L. and Zisserman, A., Geometric Invariance in Computer Vision, Appendix: Projective Geometry for Machine Vision, MIT Press, Cambridge, MA, 1992, **(read 23.1 - 23.5, 23.10)**
    - available online: http://www.cs.cmu.edu/~ph/869/papers/zisser-mundy.pdf

Projective geometry—what's it good for?

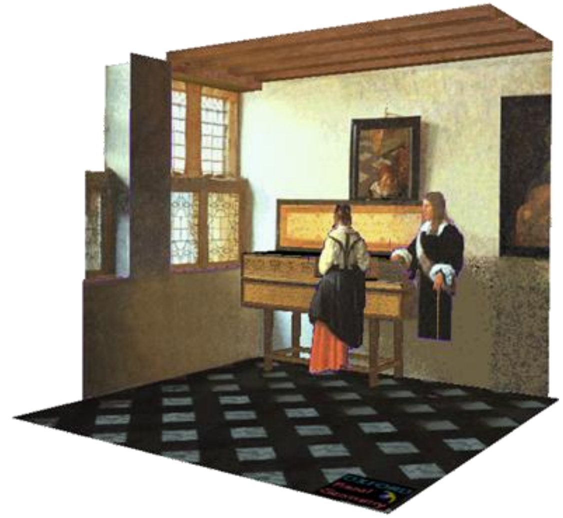
- Uses of projective geometry
  - Drawing
  - Measurements
  - Mathematics for projection
  - Undistorting images
  - Camera pose estimation
  - **Object recognition**



Paolo Uccello

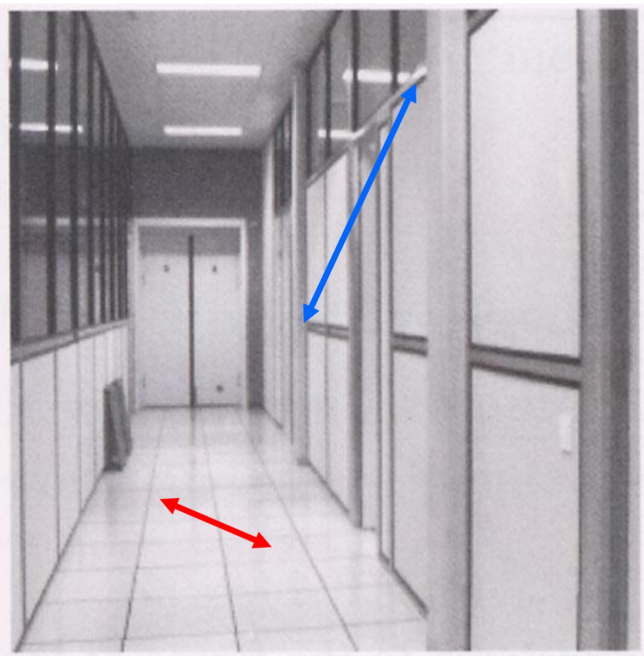
Applications of projective geometry

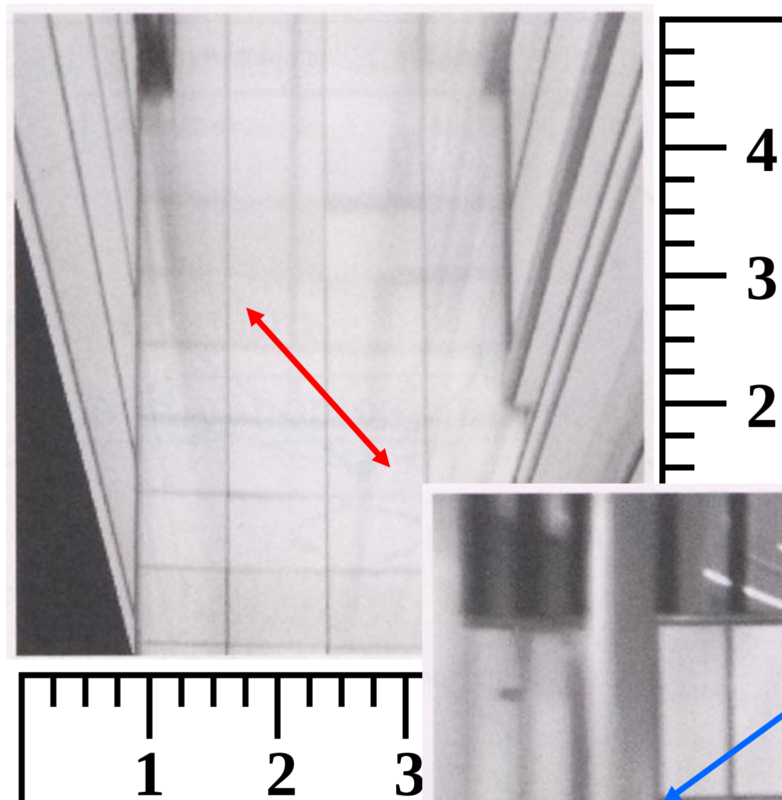
Vermeer's *Music Lesson*

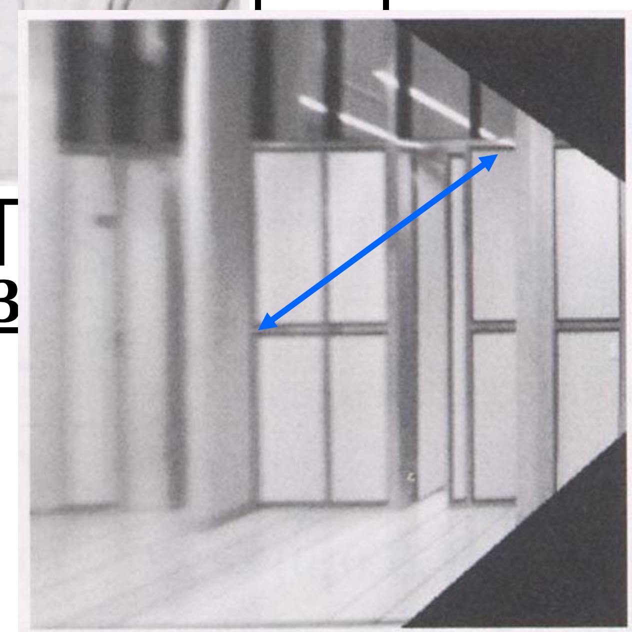


Reconstructions by Criminisi et al.

Measurements on planes





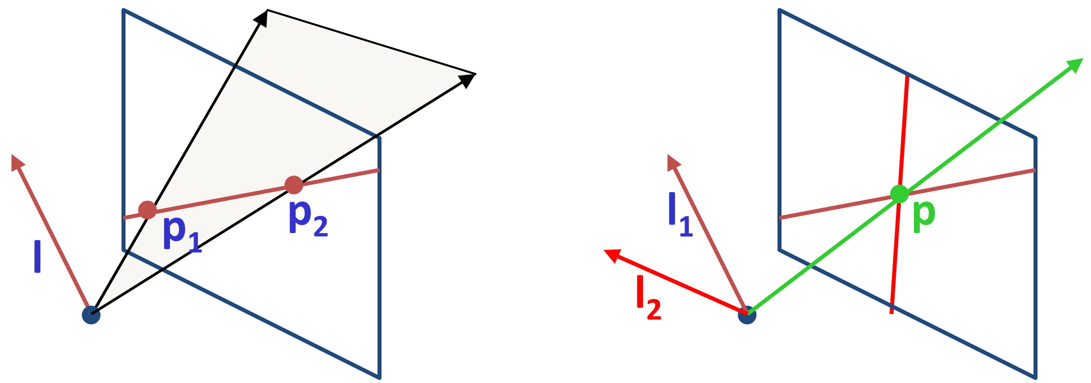


Approach: unwarp then measure

Point and line duality

– A line $\mathbf{l}$ is a homogeneous 3-vector

– It is $\perp$ to every point (ray) $\mathbf{p}$ on the line: $\mathbf{l}\ \mathbf{p}=0$



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 $\mathbf{p_2}$ $\Rightarrow$ $\mathbf{l} = \mathbf{p_1} \times \mathbf{p_2}$
- $\mathbf{l}$ can be interpreted as a *plane normal*

What is the intersection of two lines $\mathbf{l_1}$ and $\mathbf{l_2}$?

- $\mathbf{p}$ is $\perp$ to $\mathbf{l_1}$ and $\mathbf{l_2}$ $\Rightarrow$ $\mathbf{p} = \mathbf{l_1} \times \mathbf{l_2}$

Points and lines are *dual* in projective space