Computer Vision: Panoramas

Nicolás Pérez de la Blanca,
nicolas@decsai.ugr.es

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

Reading

- Szeliski Chapter 9

Panoramas

- Now we know how to create panoramas!
- Given two images:
  - Step 1: Detect features
  - Step 2: Match features
  - Step 3: Compute a homography using RANSAC
  - Step 4: Combine the images together (somehow)
- What if we have more than two images?

Homographies and Mosaics

15-463: Computational Photography
Alexei Efros, CMU, Fall 2011

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

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

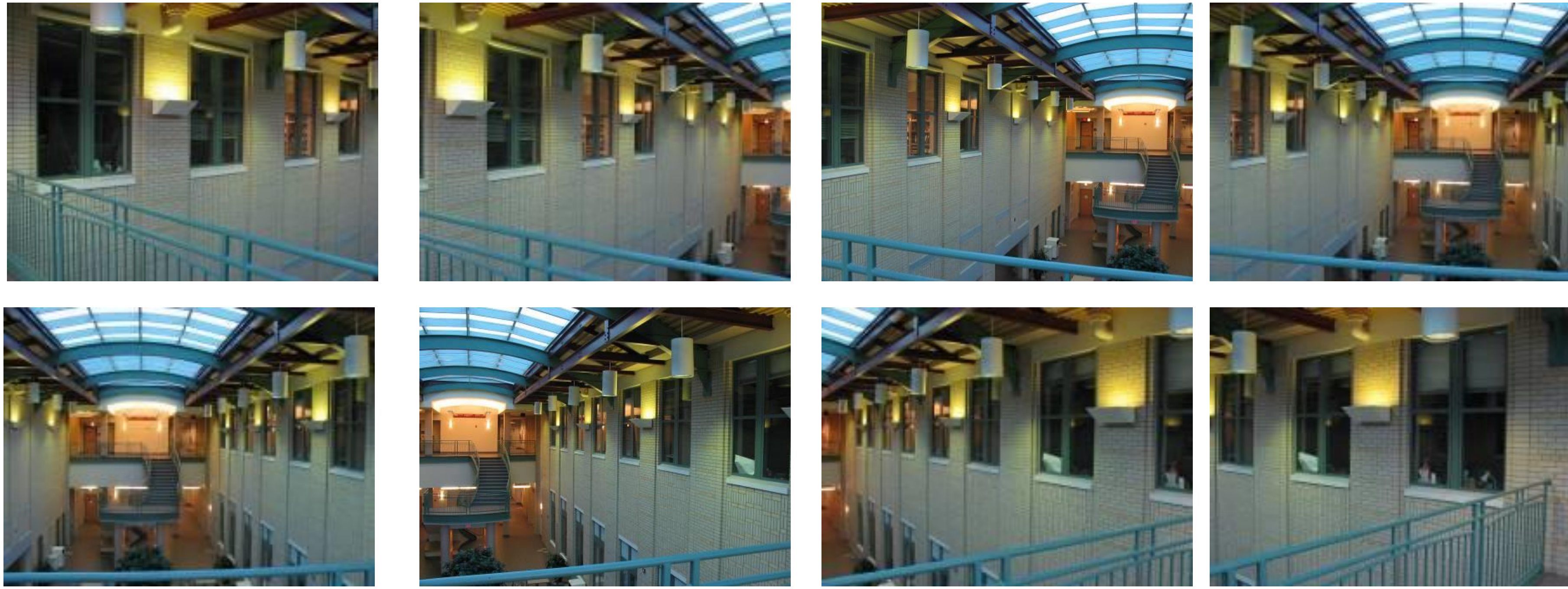
Why Mosaic?

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

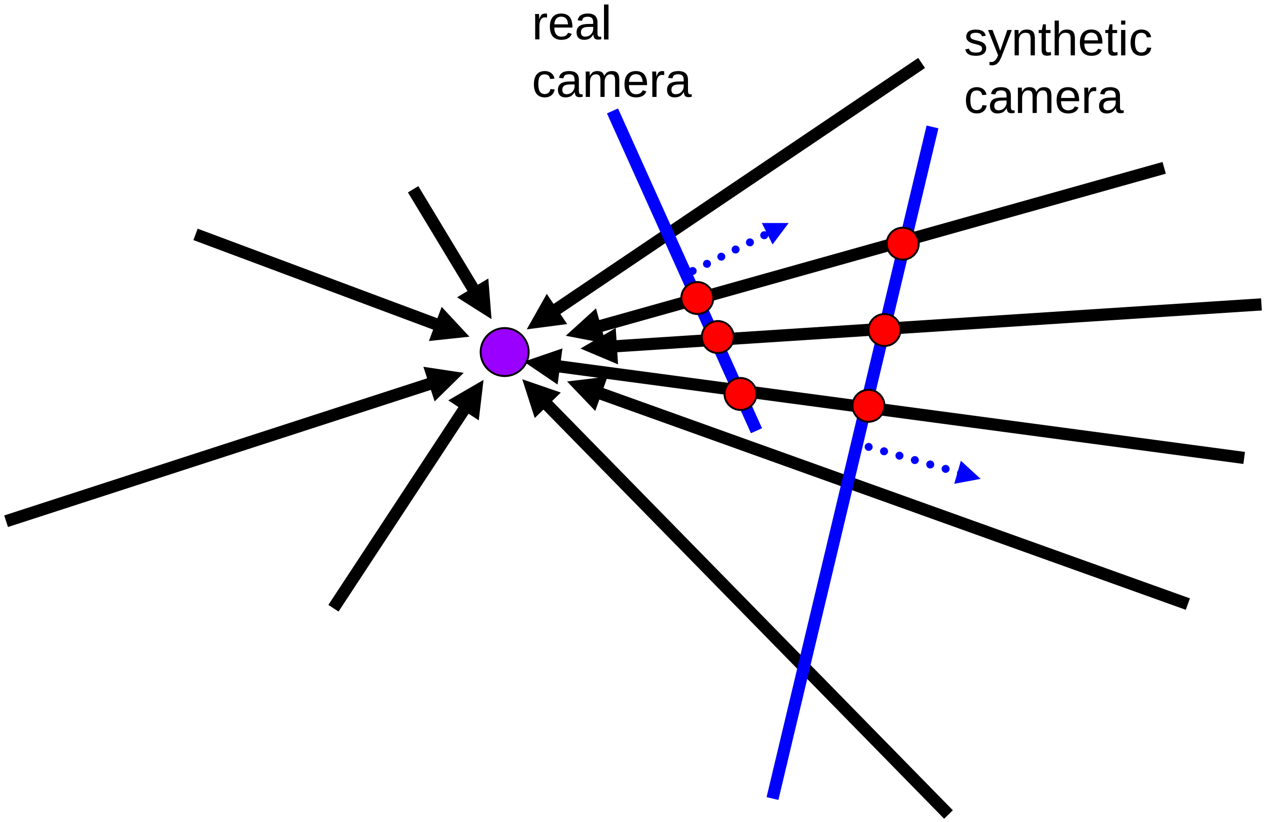
Naïve Stitching

left on top

right on top

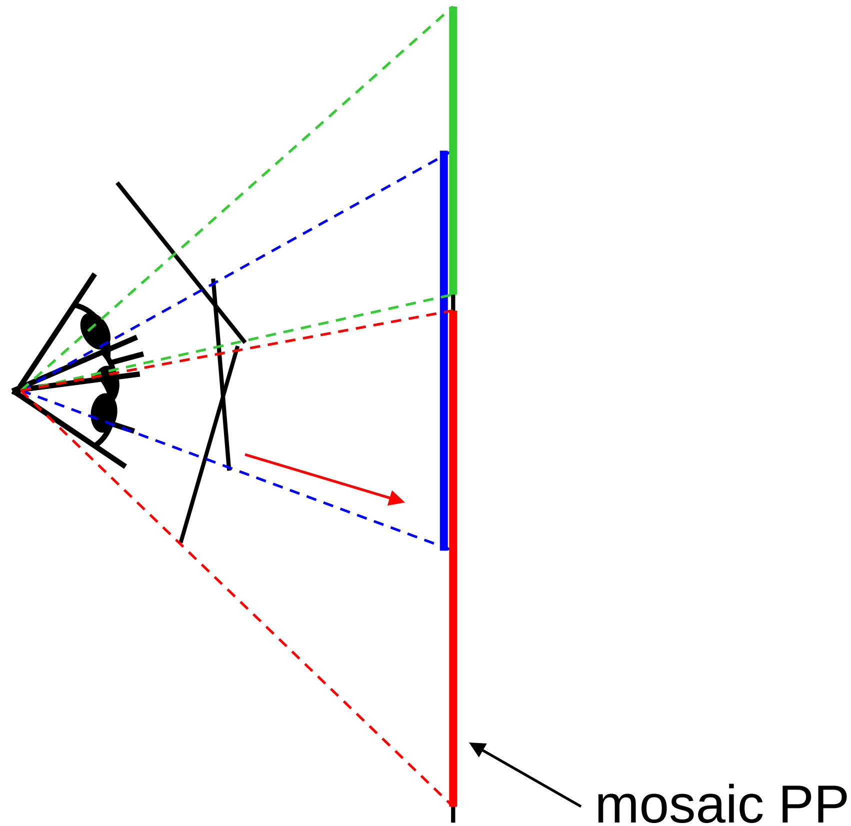
Translations are not enough to align the images

A pencil of rays contains all views



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

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*

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

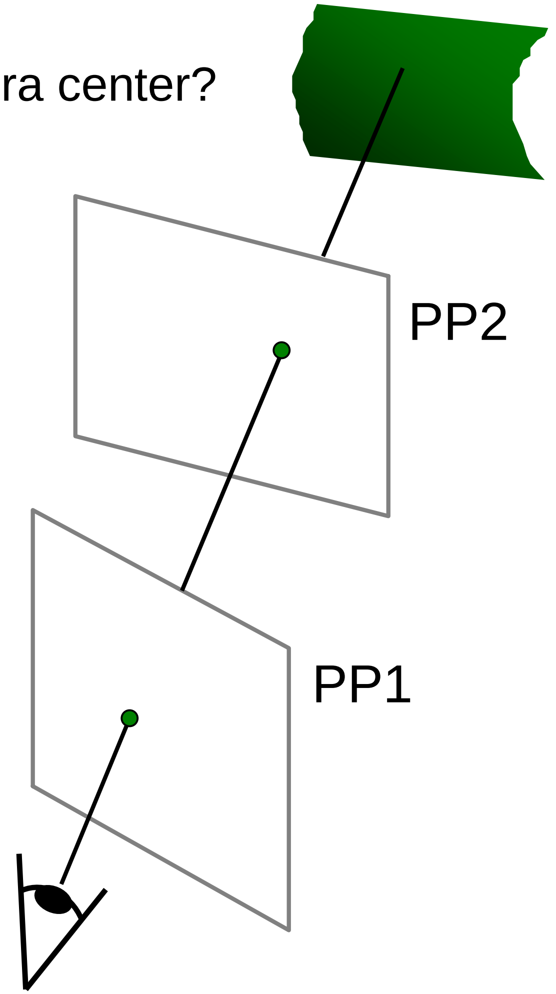
…but **wait**, why should this work at all?

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

Image reprojection

Basic question

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



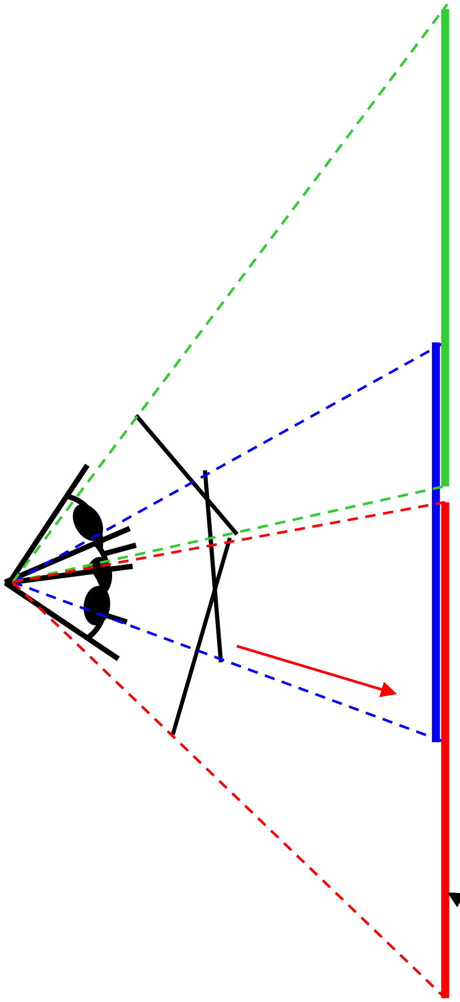
Answer

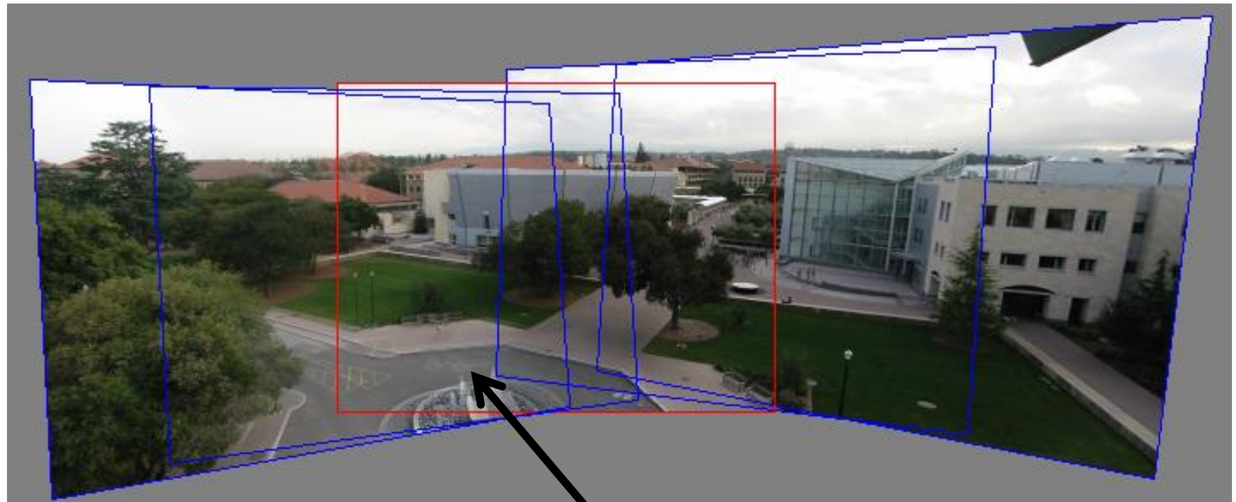
- Cast a ray through each pixel in PP1
- Draw the pixel where that ray intersects PP2

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

Rotational Mosaics





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

Can we say something more about rotational mosaics?

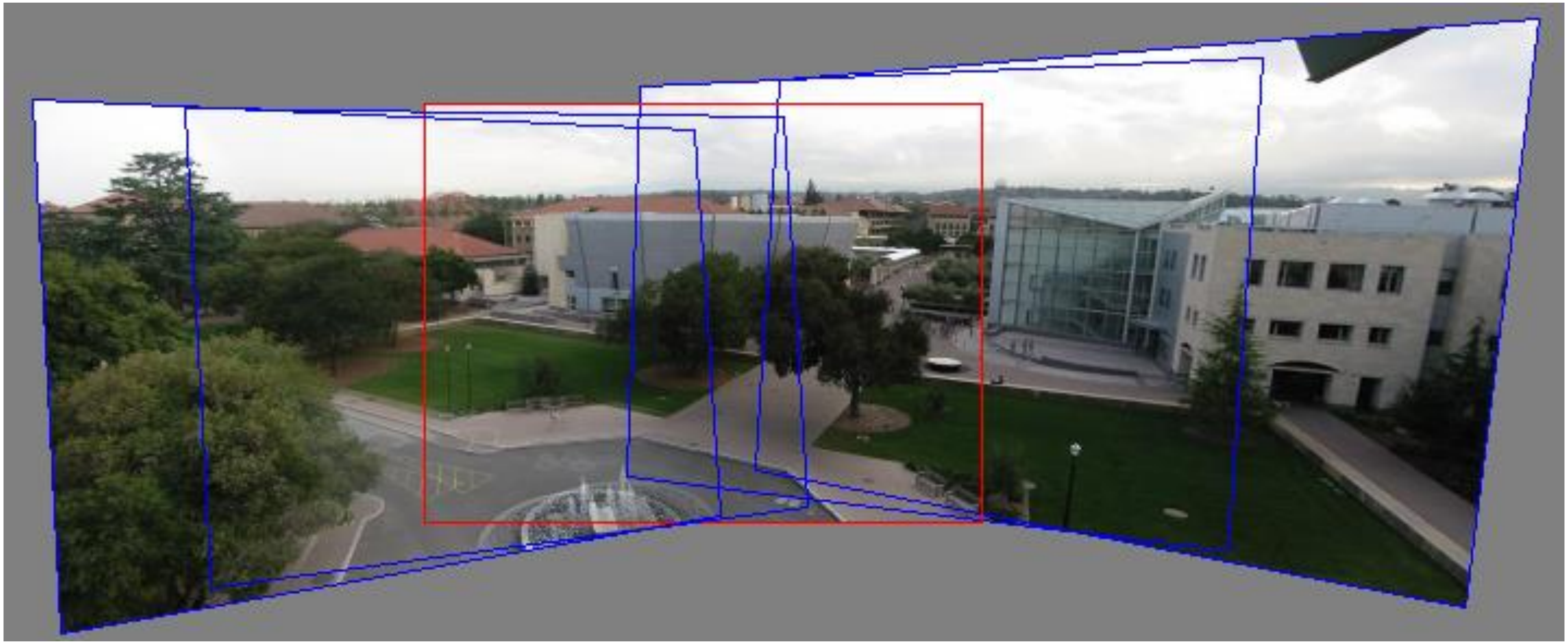
i.e. can we further constrain our H?

Algorithm for simple planar Mosaic

1. Order the image sequence from left to right according to its common areas
2. Define a large canvas for the image Mosaic
3. Select a central image as the Mosaic reference image
4. Move the reference image to the canvas using an homography
5. Compute the homography between each two adjacent images.
6. Using (5) compute the homography from each image and the canvas
7. Transport each image into the canvas using the corresponding homography

Different orders in the image transport produces mosaic color variations.

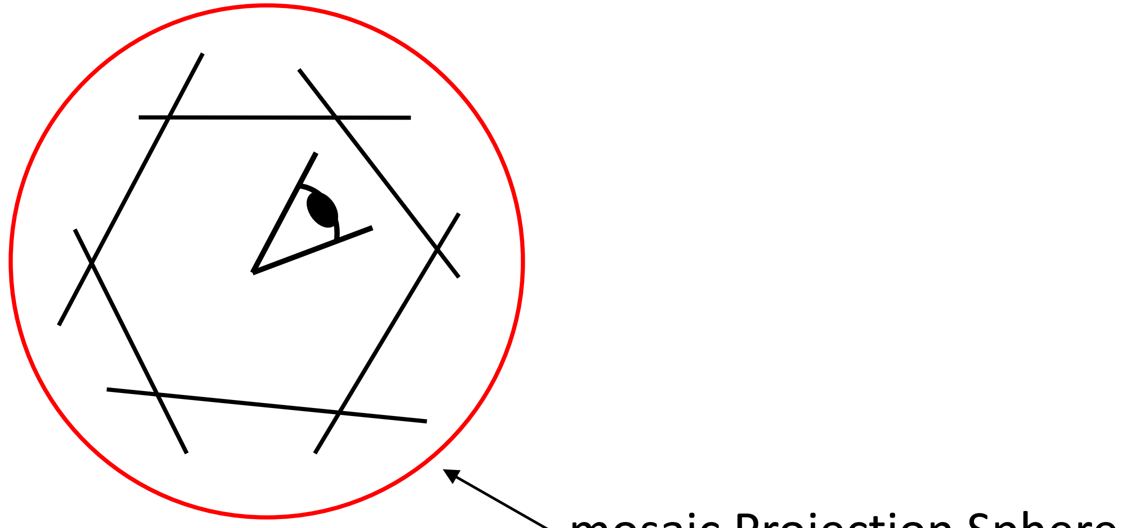
Can we use homographies to create a 360 panorama?



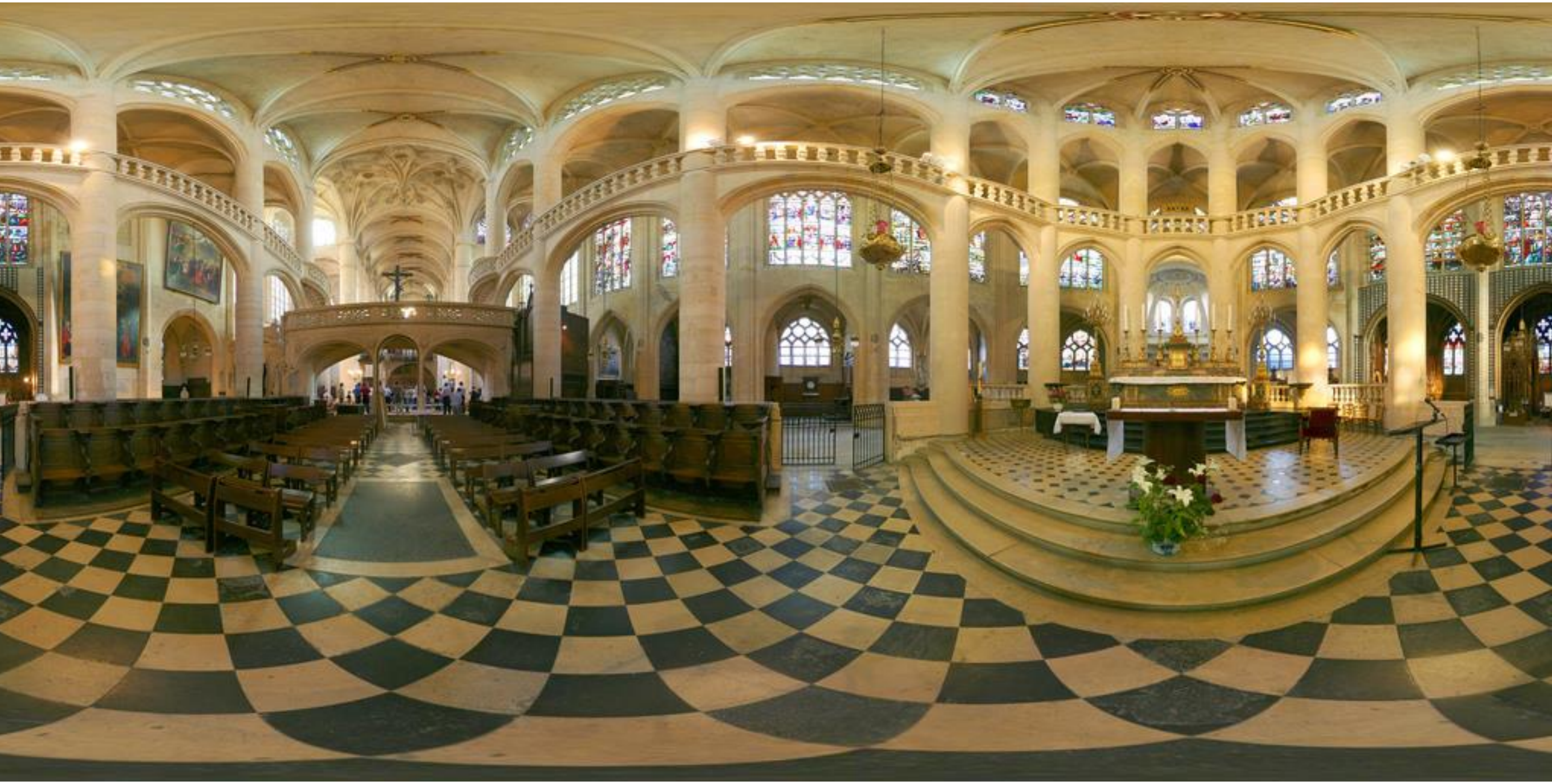
- In order to figure this out, we need to learn what a **camera** is

Panoramas

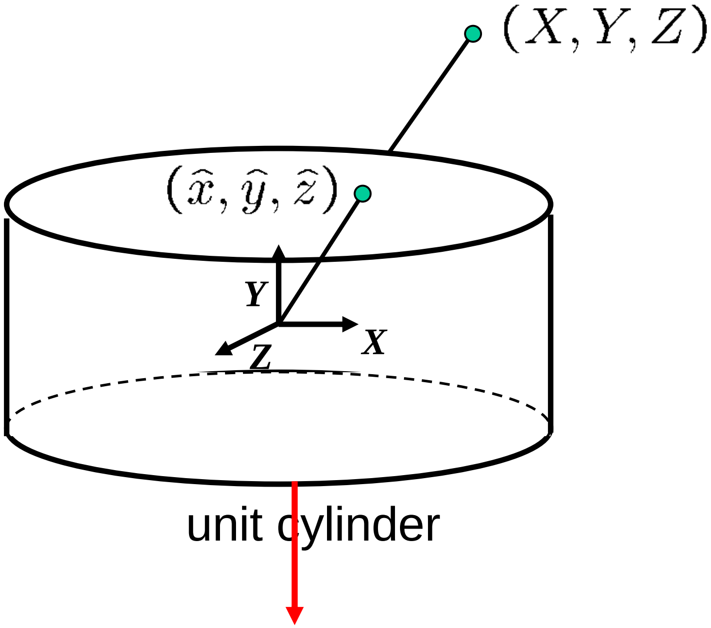
- What if you want a 360° field of view?



360 panorama



Cylindrical projection



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

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

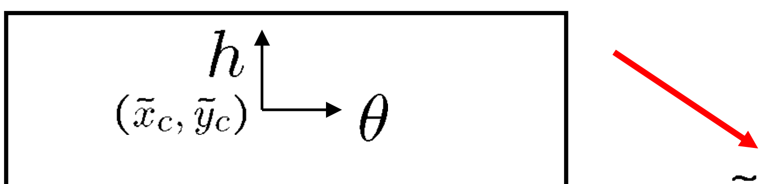
- Convert to cylindrical coordinates

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

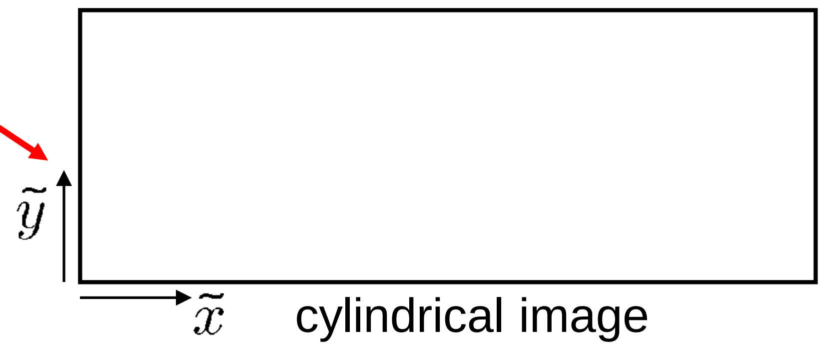
- Convert to cylindrical image coordinates

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

  – s defines size of the final image



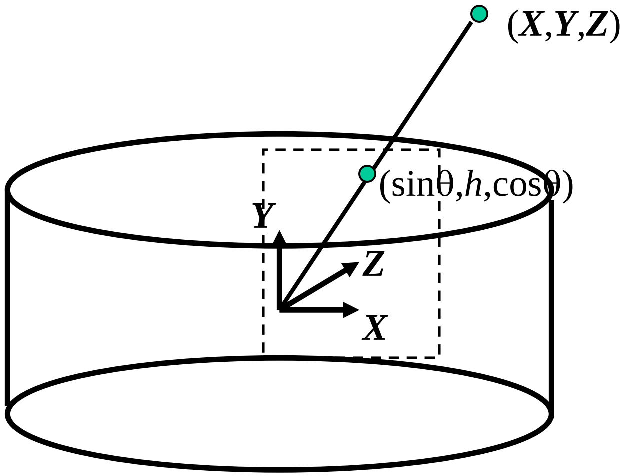
unwrapped cylinder



cylindrical image

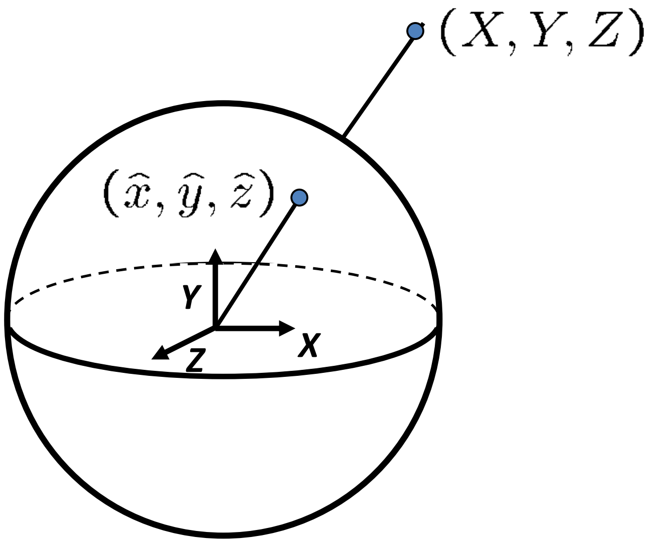
Cylindrical warping

Given focal length f and image center (x_c,y_c)

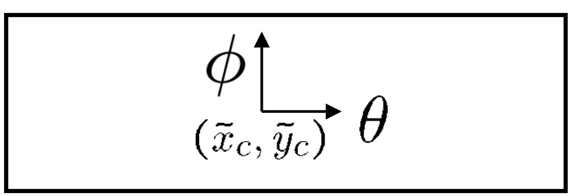


$$
\begin{aligned}
\theta &= (x_{cyl} - x_c)/f \\
h &= (y_{cyl} - y_c)/f \\
\hat{x} &= \sin\theta \\
\hat{y} &= h \\
\hat{z} &= \cos\theta \\
x &= f\hat{x}/\hat{z} + x_c \\
y &= f\hat{y}/\hat{z} + y_c
\end{aligned}
$$

Spherical projection



unit sphere



unwrapped 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

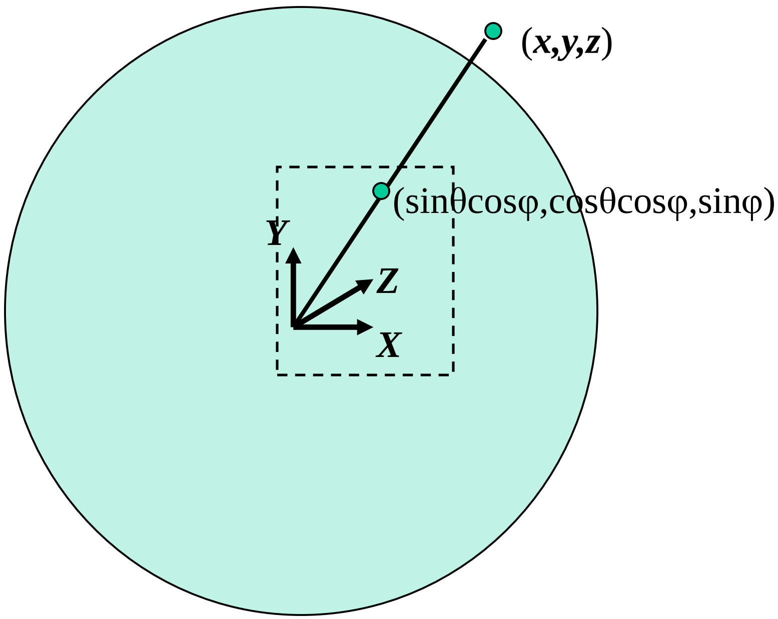
$$(\tilde{x}, \tilde{y}) = (s\theta, s\phi) + (\tilde{x}_c, \tilde{y}_c)$$

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

Spherical image

Spherical warping

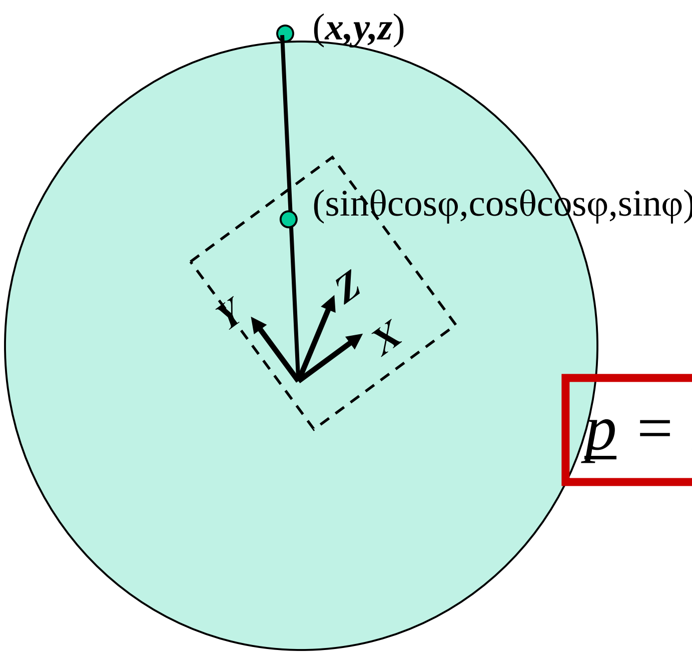
Given focal length *f* and image center (x_c,y_c)



$$
\begin{aligned}
\theta &= (x_{cyl} - x_c)/f \\
\varphi &= (y_{cyl} - y_c)/f \\
\hat{x} &= \sin\theta \cos\varphi \\
\hat{y} &= \sin\varphi \\
\hat{z} &= \cos\theta \cos\varphi \\
x &= f\hat{x}/\hat{z} + x_c \\
y &= f\hat{y}/\hat{z} + y_c
\end{aligned}
$$

3D rotation

Rotate image before placing on unrolled sphere



$$\boxed{\underline{p} = \boldsymbol{R}\, p}$$

$$
\begin{aligned}
\theta &= (x_{cyl} - x_c)/f \\
\varphi &= (y_{cyl} - y_c)/f \\
\hat{x} &= \sin\theta \cos\varphi \\
\hat{y} &= \sin\varphi \\
\hat{z} &= \cos\theta \cos\varphi \\
x &= f\underline{\hat{x}}/\underline{\hat{z}} + x_c \\
y &= f\underline{\hat{y}}/\underline{\hat{z}} + y_c
\end{aligned}
$$

Radial distortion

Correct for "bending" in wide field of view lenses

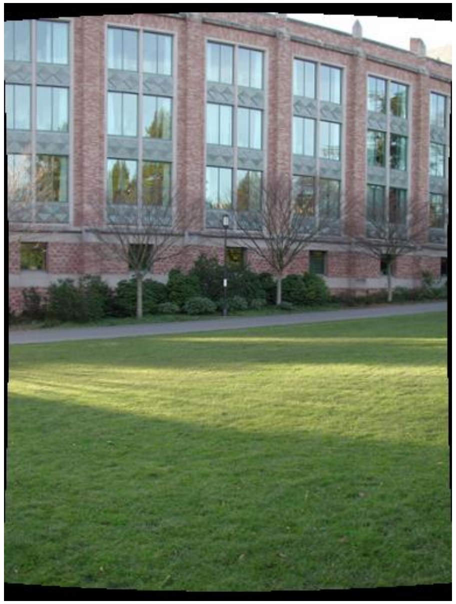
$$
\begin{aligned}
\widehat{r}^2 &= \widehat{x}^2 + \widehat{y}^2 \\
\widehat{x}' &= \widehat{x}/(1 + \kappa_1 \widehat{r}^2 + \kappa_2 \widehat{r}^4) \\
\widehat{y}' &= \widehat{y}/(1 + \kappa_1 \widehat{r}^2 + \kappa_2 \widehat{r}^4) \\
x &= f\widehat{x}'/\widehat{z} + x_c \\
y &= f\widehat{y}'/\widehat{z} + y_c
\end{aligned}
$$

Spherical reprojection

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

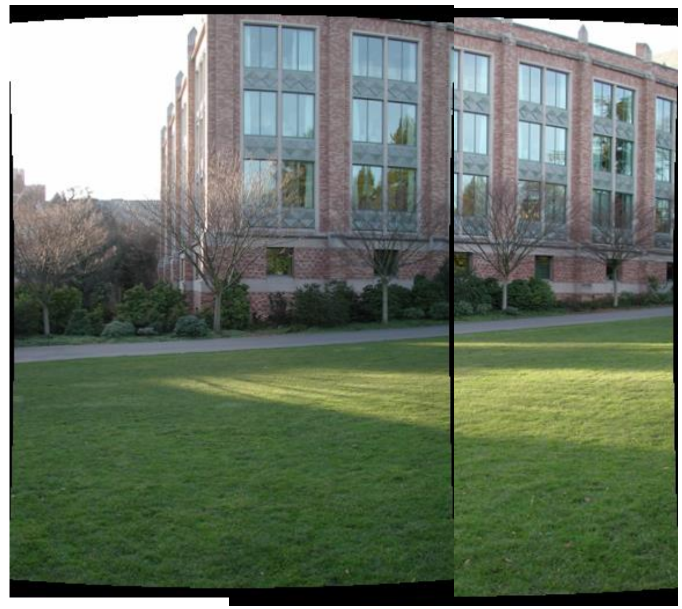
- Map image to spherical coordinates
  - need to know the focal length

Aligning spherical images



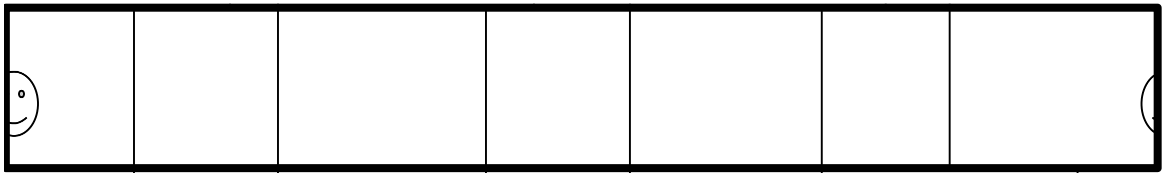
- Suppose we rotate the camera by θ about the vertical axis
  - How does this change the spherical image?

Aligning spherical images



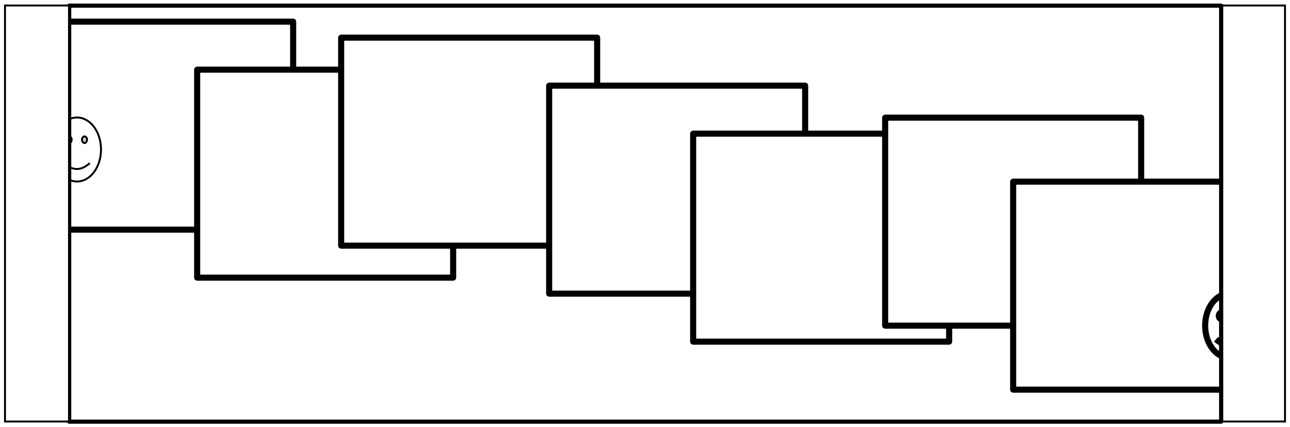
- Suppose we rotate the camera by θ about the vertical axis
  - How does this change the spherical image?
    - Translation by θ
  - This means that we can align spherical images by translation

Assembling the panorama



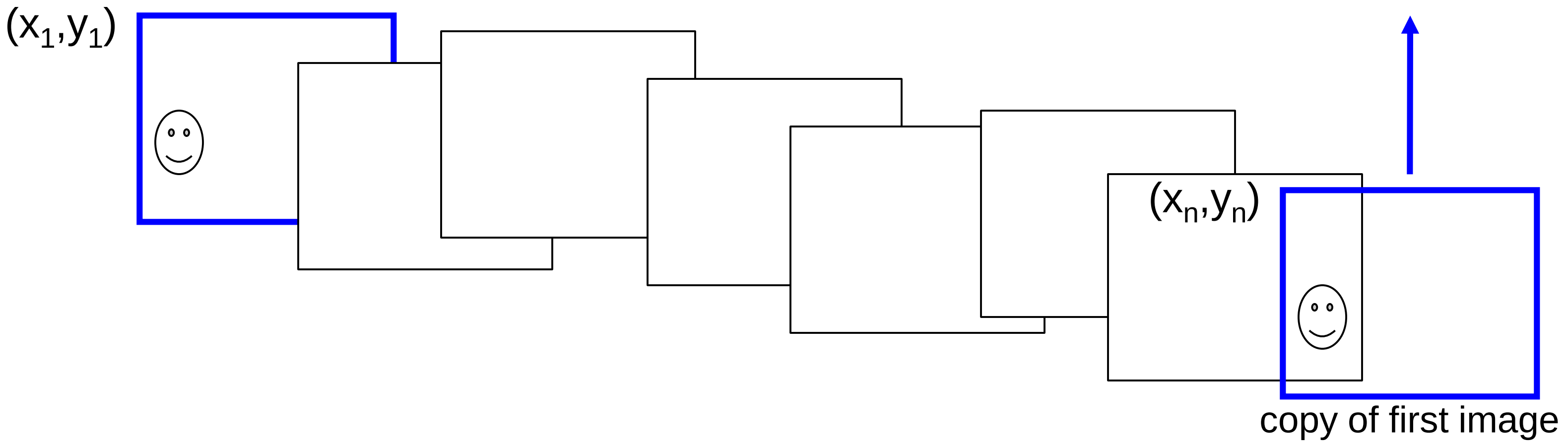
- Stitch pairs together, blend, then crop

Problem: Drift



- Error accumulation
  - small errors accumulate over time

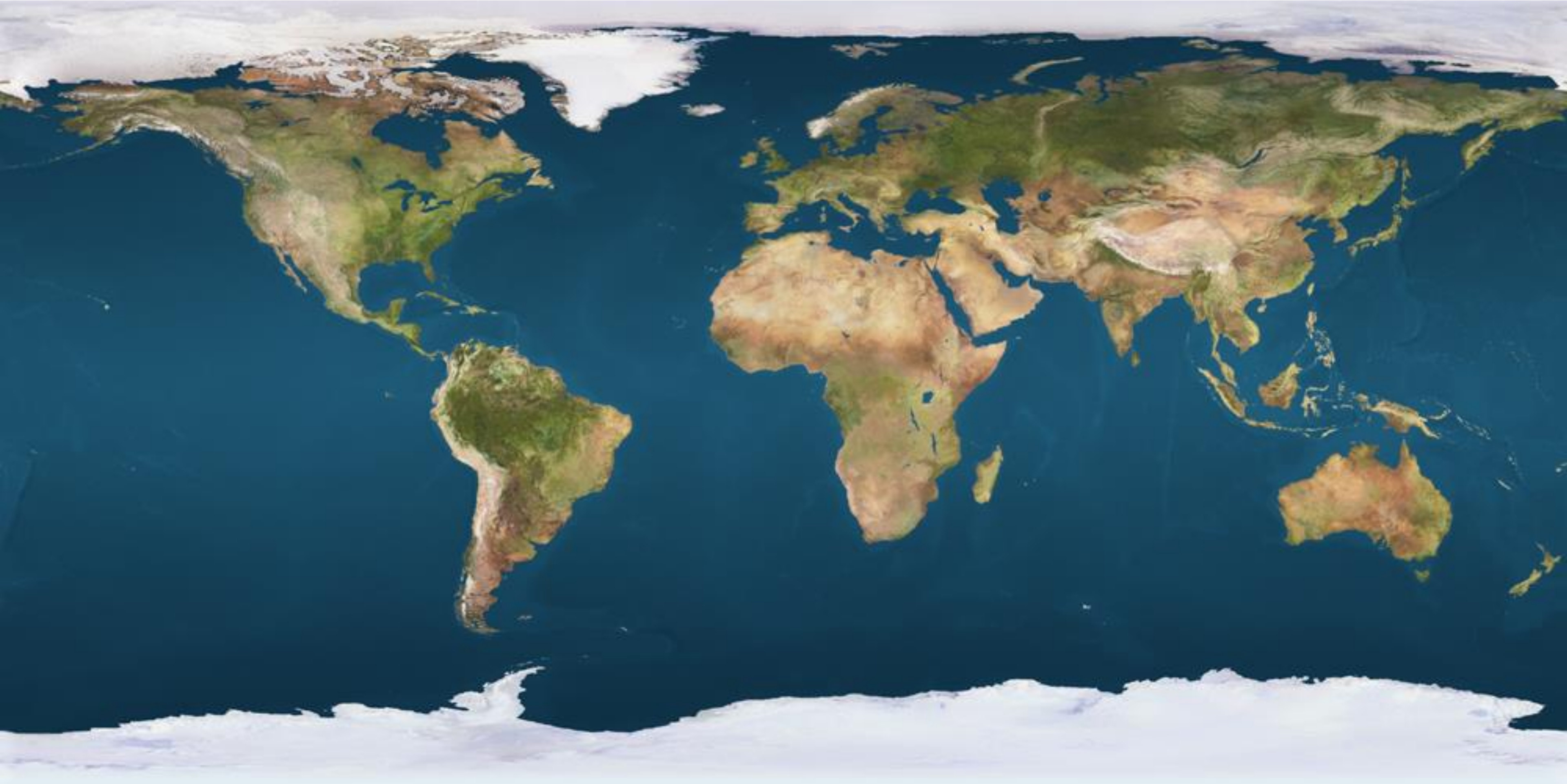
Problem: Drift



- Solution
  - add another copy of first image at the end
  - this gives a constraint: $y_n = y_1$
  - there are a bunch of ways to solve this problem
    - add displacement of $(y_1 - y_n)/(n - 1)$ to each image after the first
    - **apply an affine warp: $y' = y + ax$**
    - run a big optimization problem, incorporating this constraint
      - best solution, but more complicated
      - known as "bundle adjustment"

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

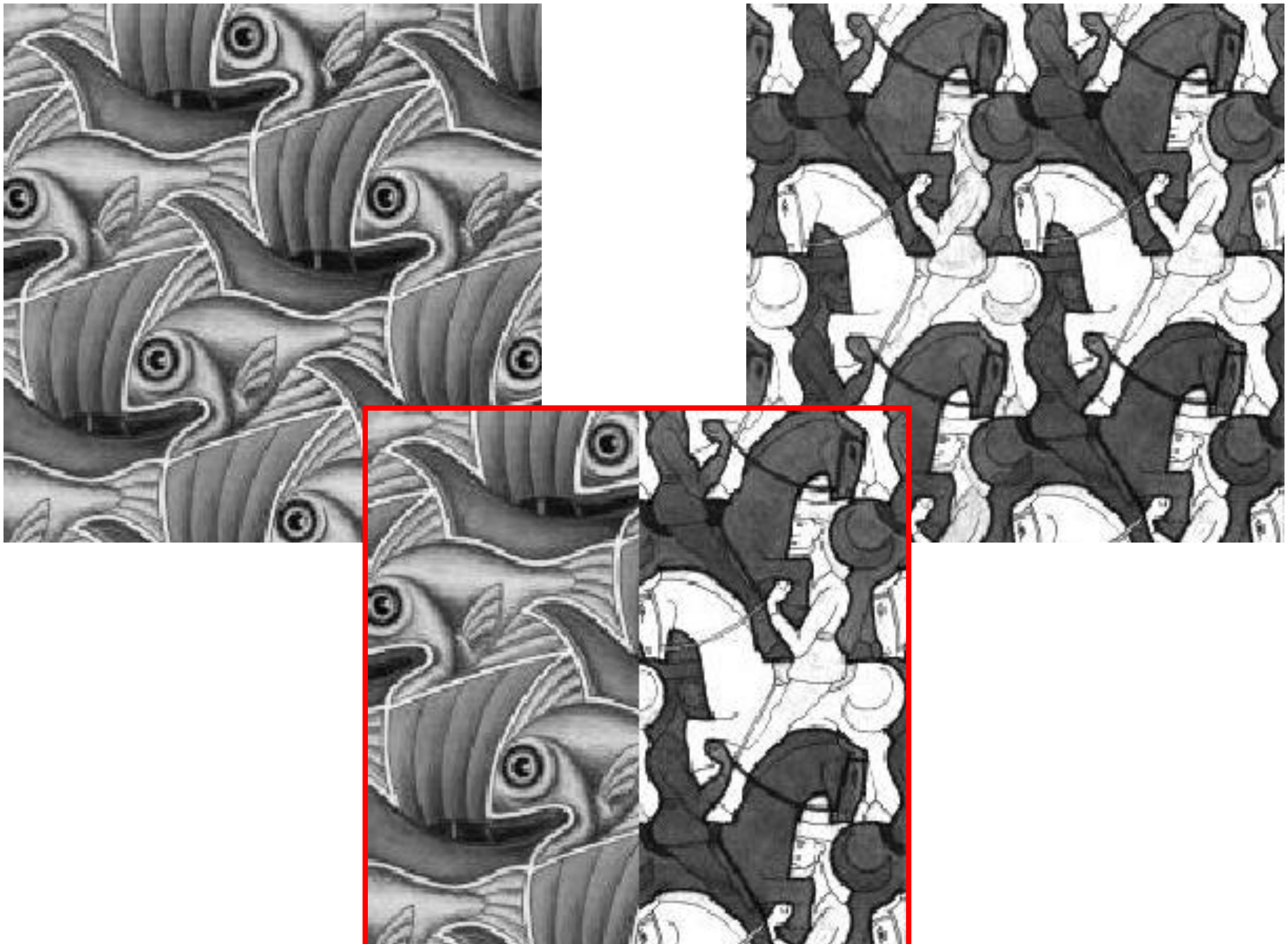
Blending

- We’ve aligned the images – now what?

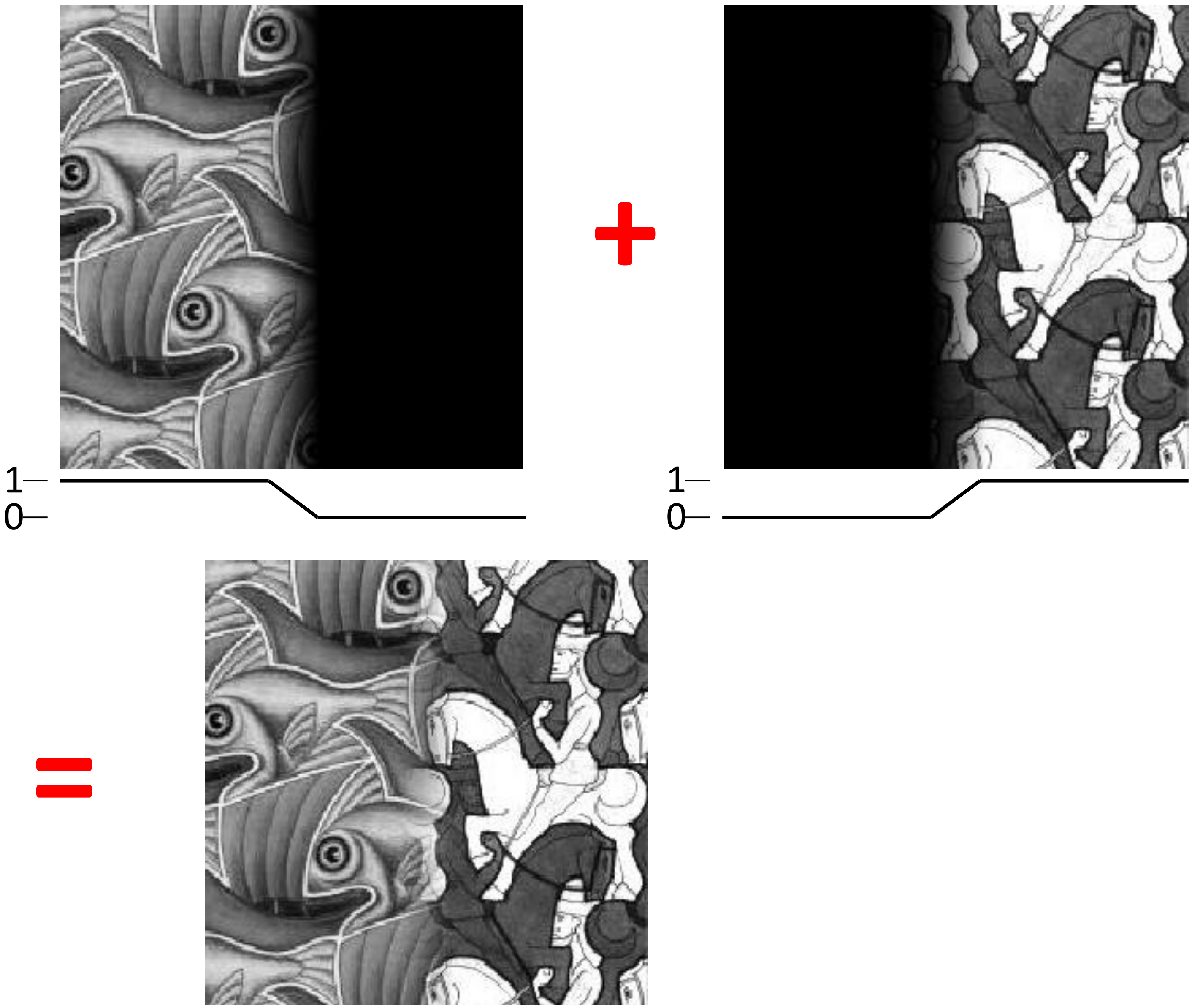
Blending

- Want to seamlessly blend them together

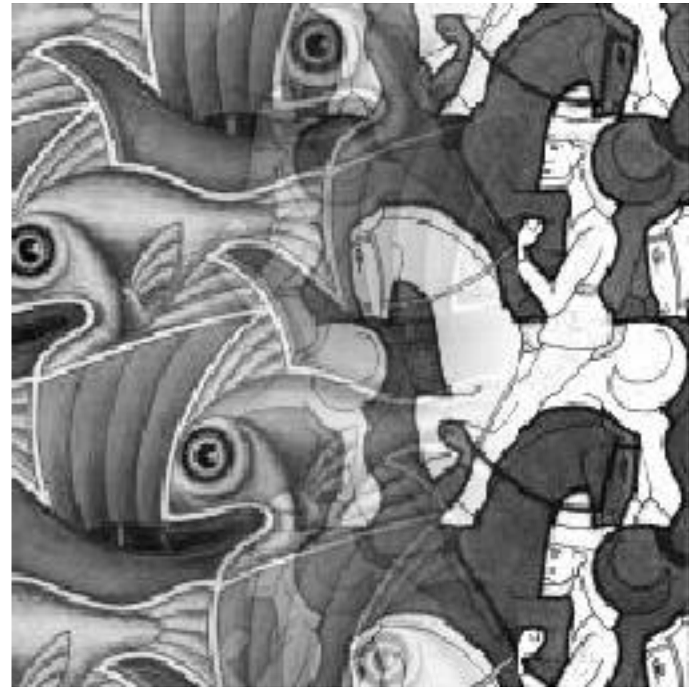
Image Blending

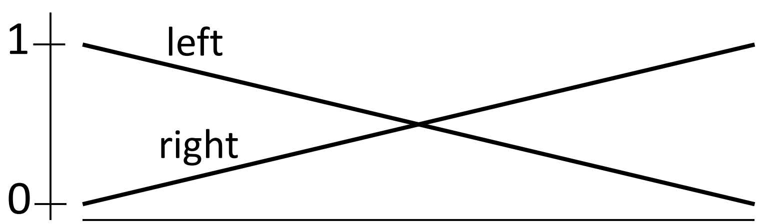


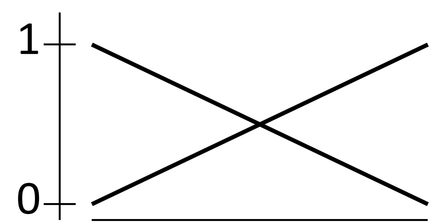
Feathering



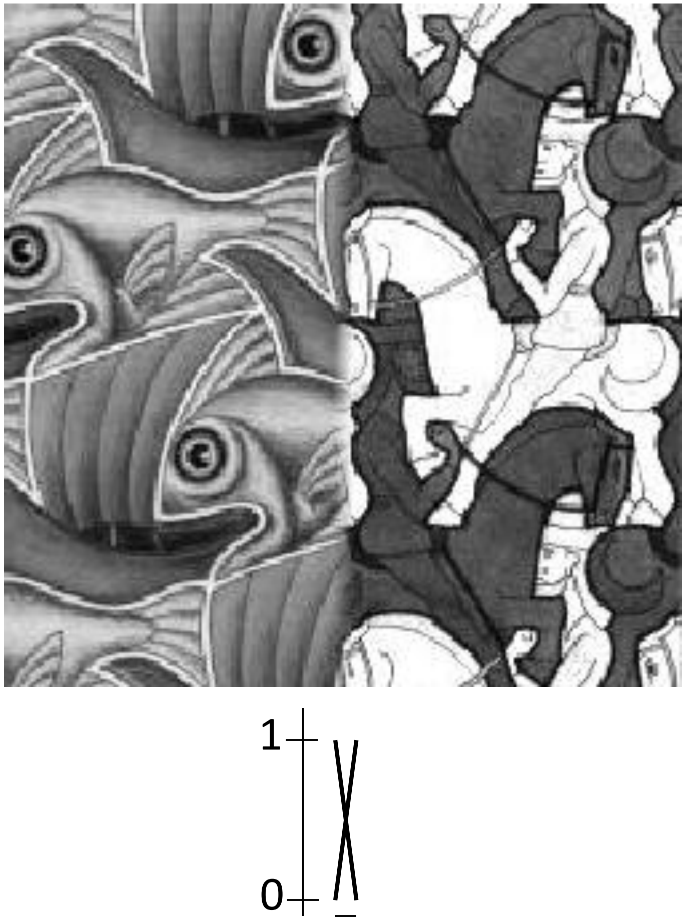
Effect of window size

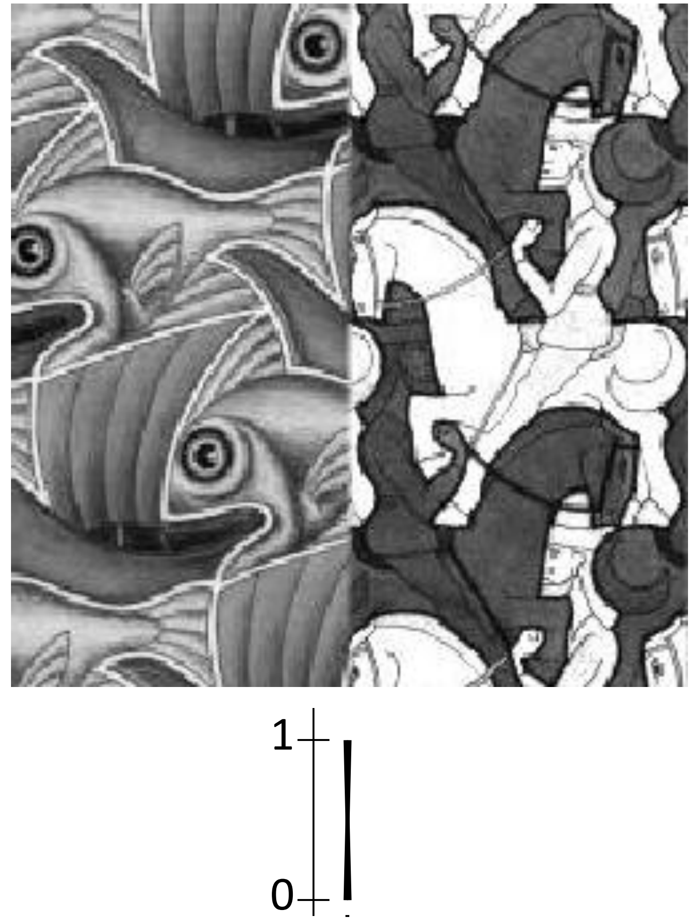




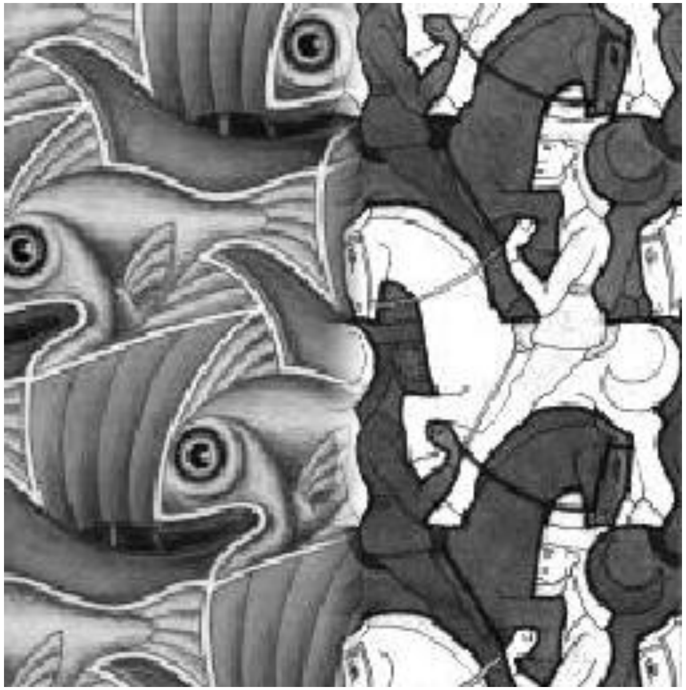


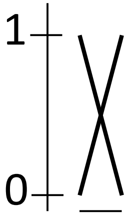
Effect of window size





Good window size

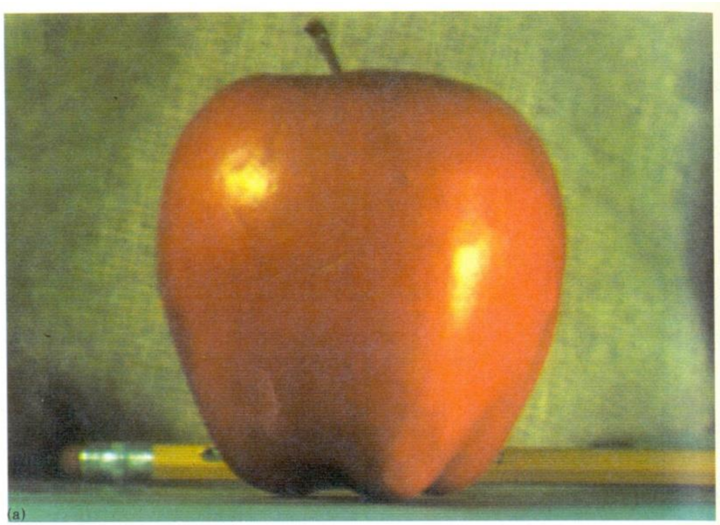


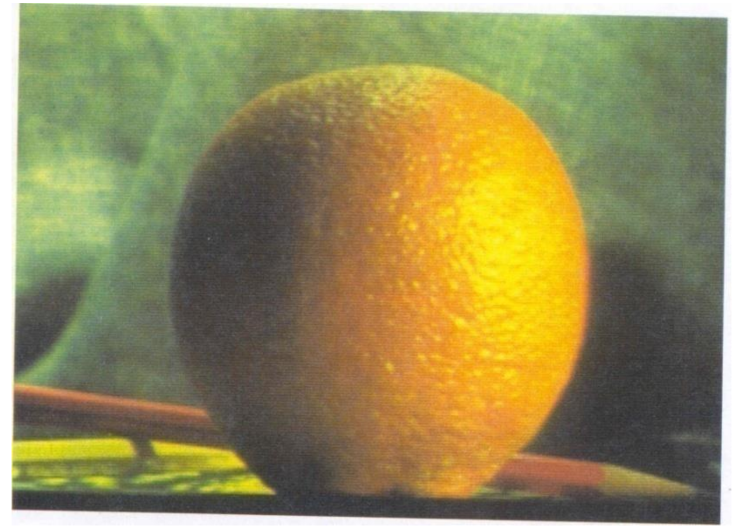


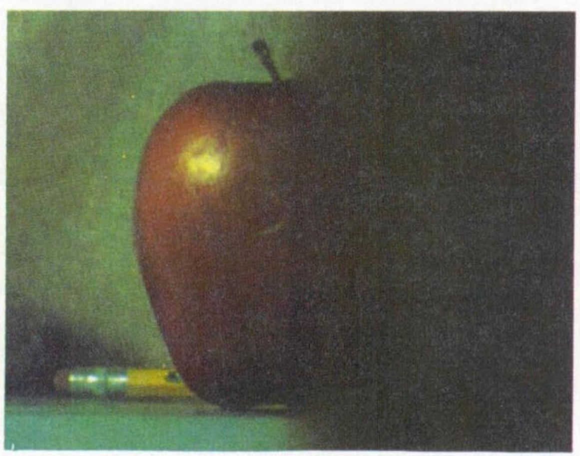
"Optimal" window: smooth but not ghosted

- Doesn't always work...

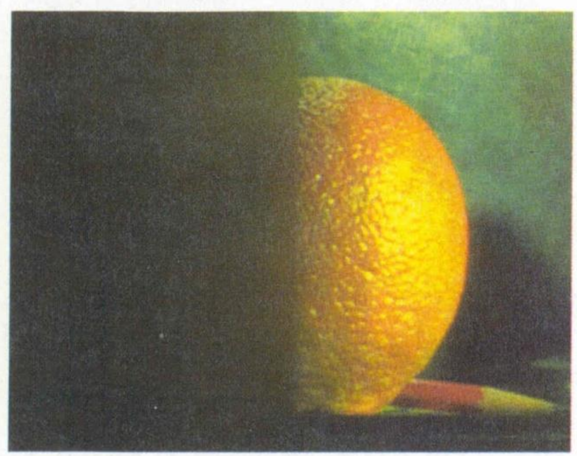
Pyramid blending



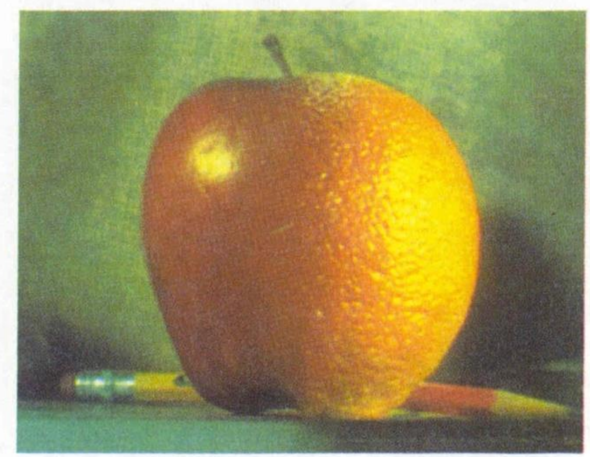




(d)



(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.

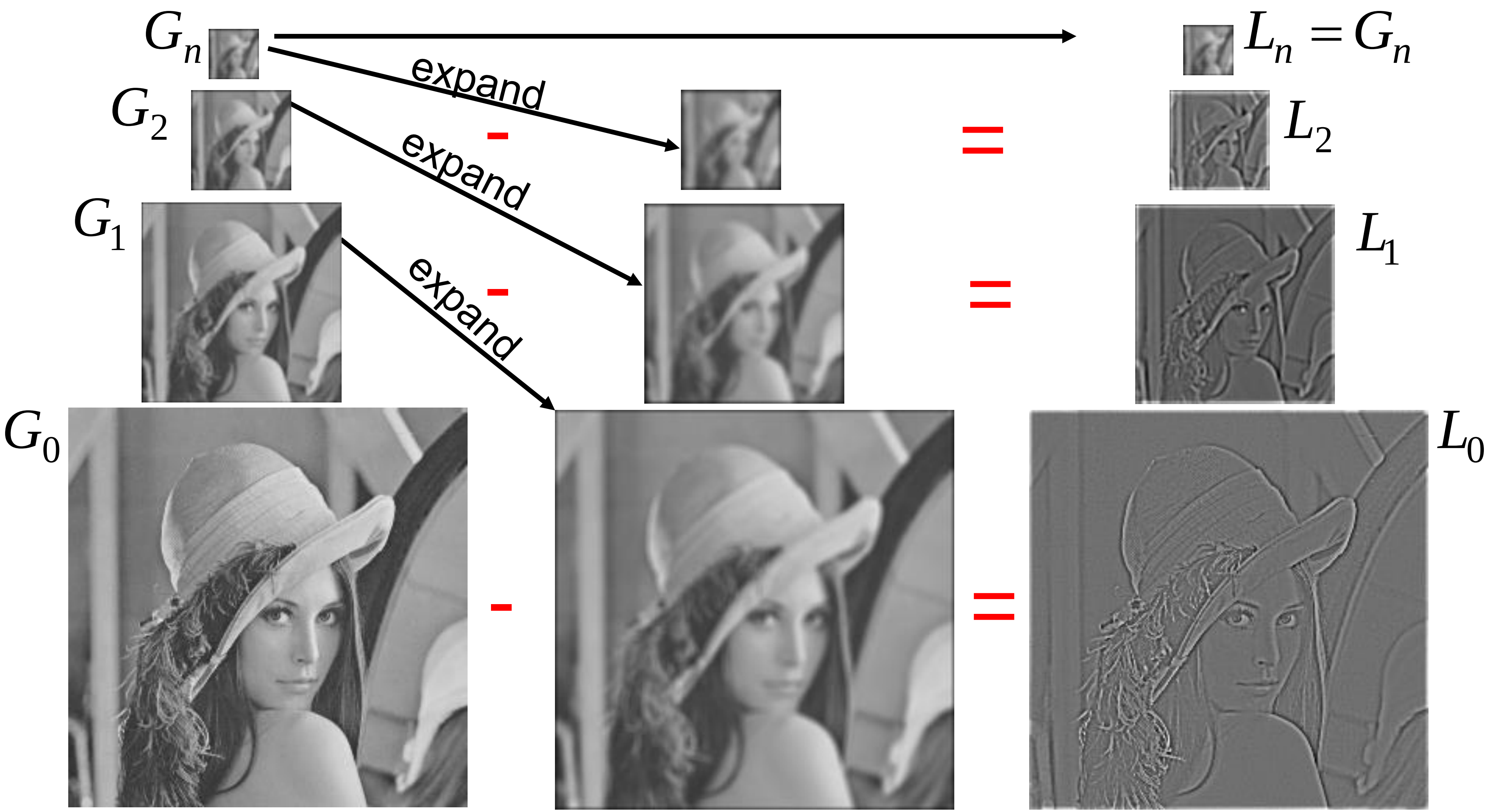
The Laplacian Pyramid

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

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

Gaussian Pyramid

Laplacian Pyramid



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](http://earthobservatory.nasa.gov/Newsroom/BlueMarble/) for some stunning earth mosaics
    - http://earthobservatory.nasa.gov/Newsroom/BlueMarble/
    - Click for images…

Questions?