Announcements

- Project 1 artifact voting (email announce later today)
- Project 2 out today (help session at end of class)
- **IMPORTANT**: choose Proj 2 partner and SIGNUP for panorama equip TODAY:

Today's Readings


Image Mosaics

Goal

- Stitch together several images into a seamless composite

Mosaics

Today's Readings


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
  - Lucas & Kanade registration
- Shift the second image to overlap with the first
- Blend the two together to create a mosaic
- If there are more images, repeat
Aligning images

How to account for warping?
- Translations are not enough to align the images
- Photoshop demo

Image reprojection

The mosaic has a natural interpretation in 3D
- The images are reprojected onto a common plane
  - Idea: replace camera with slide projector, project onto new PP
- The mosaic is formed on this plane

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

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

Perspective projection of a plane

- Lots of names for this:
  - homography, texture-map, colineation, planar projective map
- Modeled as a 2D warp using homogeneous coordinates

\[
\begin{bmatrix}
    w y' \\
    w y' \\
    w
\end{bmatrix} = \begin{bmatrix}
    H & p \\
    0 & 1
\end{bmatrix}
\]

To apply a homography $H$

- Compute $p' = Hp$ (regular matrix multiply)
- Convert $p'$ from homogeneous to image coordinates
  - divide by $w$ (third) coordinate

Image warping with homographies

Panoramas

What if you want a 360° field of view?

Mosaic Projection Cylinder

Cylindrical projection

- Map 3D point $(X,Y,Z)$ onto cylinder
  \[ (\hat{x}, \hat{y}, \hat{z}) = \frac{1}{\sqrt{x^2 + y^2}} (X, Y, Z) \]
- Convert to cylindrical coordinates
  \[ (\hat{\theta}, \hat{z}, \hat{r}) = (\hat{x}, \hat{y}, \hat{z}) \]
- Convert to cylindrical image coordinates
  \[ (\hat{x}, \hat{y}) = (fr, fh) + (\hat{\theta}, \hat{z}) \]
Cylindrical reprojection

How to map from a cylinder to a planar image?

- Apply camera projection matrix
  - for project 2, account for focal length and assume principle point is at center of image
    - $x' = \frac{w}{2}$ image width, $y' = \frac{h}{2}$ image height
  - $\begin{bmatrix} w' \\ y' \\ w \\ z \end{bmatrix} = \begin{bmatrix} -f & 0 & w/2 & 0 \\ 0 & -f & h/2 & 0 \\ 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} x \\ y \\ z \end{bmatrix}$

- Convert to image coordinates
  - divide by third coordinate ($w$)

- Map image to cylindrical coordinates
  - need to know the camera focal length

Cylindrical panoramas

Steps
- Reproject each image onto a cylinder
- Blend
- Output the resulting mosaic

Cylindrical image stitching

What if you don’t know the camera rotation?
- Solve for the camera rotations
  - Note that a rotation of the camera is a translation of the cylinder!
  - Use Lukas-Kanade to solve for translations of cylindrically-warped images
Project 2 (out today)

1. Take pictures on a tripod (or handheld)
2. Warp to cylindrical coordinates
3. Automatically compute pair-wise alignments
4. Correct for drift
5. Blend the images together
6. Crop the result and import into a viewer
Feathering

\[ I(x,y) = (\alpha R, \alpha G, \alpha B, \alpha) \]

\[ I_{\text{blend}} = I_{\text{left}} + I_{\text{right}} \]

See Blinn reading (CGA, 1994) for details

Effect of window size

"Optimal" window: smooth but not ghosted
- Doesn’t always work...
Pyramid blending

Create a Laplacian pyramid, blend each level


Image warping

Given a coordinate transform \( (x',y') = h(x,y) \) and a source image \( f(x,y) \), how do we compute a transformed image \( g(x',y') = f(h(x,y)) \)?

Forward warping

Send each pixel \( f(x,y) \) to its corresponding location \( (x',y') = h(x,y) \) in the second image

Q: what if pixel lands “between” two pixels?

A: distribute color among neighboring pixels \( (x',y') \) – Known as “splatting”
Inverse warping

Get each pixel \( g(x',y') \) from its corresponding location \((x,y) = h^{-1}(x',y')\) in the first image.

Q: what if pixel comes from “between” two pixels?

A: resample color value

- We discussed resampling techniques before
  - nearest neighbor, bilinear, Gaussian, bicubic

Forward vs. inverse warping

Q: which is better?

A: usually inverse—eliminates holes
  - however, it requires an invertible warp function—not always possible...

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/
Path Images

Cyclograph

Some path image references

Manifold mosaics:

Stereo Panoramas

Concentric Mosaics