Image Stitching
Panoramic Image Mosaics

Full screen panoramas (cubic):  http://www.panoramas.dk/
Image Mosaics

Goal: Stitch together several images into a seamless composite
Gigapixel images

GigaPan http://gigapan.com
Today’s lecture

Image alignment and stitching
• motion models
• image warping
• point-based alignment
• complete mosaics (global alignment)
• compositing and blending
• ghost and parallax removal
Today’s lecture

Image alignment and stitching
• **motion models**
• image warping
• point-based alignment
• complete mosaics (global alignment)
• compositing and blending
• ghost and parallax removal
Motion models

What happens when we take two images with a camera and try to align them?

• translation?
• rotation?
• scale?
• affine?
• perspective?
Image Warping

image filtering: change *range* of image

\[ g(x) = h(f(x)) \]

image warping: change *domain* of image

\[ g(x) = f(h(x)) \]
Parametric (global) warping

Examples of parametric warps:

- translation
- rotation
- aspect
- affine
- perspective
- cylindrical
2D coordinate transformations

translation: \[ x' = x + t \quad x = (x, y) \]
rotation: \[ x' = R x + t \]
similarity: \[ x' = s R x + t \]
affine: \[ x' = A x + t \]
perspective: \[ x' \cong H x \quad x = (x, y, 1) \]

(\(x\) is a \emph{homogeneous} coordinate)

These all form a nested \emph{group} (closed w/ inv.)
Image Warping

Given a coordinate transform $h(x)$ and a source image $f(x)$, how do we compute a transformed image $g(x) = f(h(x))$?
Forward Warping

Send each pixel $f(x)$ to its corresponding location $h(x)$ in $g(x)$

- What if pixel lands “between” two pixels?
Forward Warping

Send each pixel \( f(x) \) to its corresponding location \( h(x) \) in \( g(x) \)

- What if pixel lands “between” two pixels?
- Answer: add “contribution” to several pixels, normalize later (splatting)
Inverse Warping

Get each pixel $g(x)$ from its corresponding location $\text{Inverse}(h(x))$ in $f(x)$

- What if pixel comes from “between” two pixels?
Inverse Warping

Get each pixel $g(x)$ from its corresponding location $\text{Inverse}(h(x))$ in $f(x)$

• What if pixel comes from “between” two pixels?
• Answer: resample color value from interpolated (prefiltered) source image
Interpolation

Possible interpolation filters:
- nearest neighbor
- bilinear
- bicubic (interpolating)
- sinc
Motion models (reprise)
Motion models

- Translation: 2 unknowns
- Affine: 6 unknowns
- Perspective (Homography): 8 unknowns
- 3D rotation: 3 unknowns
Translation

\[ x' = x + cx \]
\[ y' = y + cy \]

2 unknowns
Affine

x' = ax + by + c
y' = dx + ey + f

\[
\begin{pmatrix}
  x' \\
  y' \\
  1
\end{pmatrix} =
\begin{pmatrix}
a & b & c \\
 d & e & f \\
 0 & 0 & 1
\end{pmatrix}
\begin{pmatrix}
x \\
y \\
1
\end{pmatrix}
\]

Linear is not affine?
Perspective - Homography

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

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

\[p' = Hp\]  \hspace{1cm} \text{(regular matrix multiply)}

To apply a homography \(H\)

• Compute \(p' = Hp\)  \hspace{1cm} \text{(regular matrix multiply)}
• Convert \(p'\) from homogeneous to image coordinates
  – divide by \(w\) (third) coordinate
Image warping with homographies

image plane in front

image plane below
3D Rotation

Important for Mosaics (no camera translation)
360 degrees panorama is an example