Calvin and Hobbes

Panoramas

AHHHH...

Uh-oh, something is seriously wrong here.

The laws of perspective have been repealed!

Objects no longer diminish in size with distance.

Lines do not converge toward any point on the horizon!

All spatial relationships are lost! It's impossible to judge where anything is! Oh no!

Calvin, quit running around and crashing into things, or I'll sell you to the monkey house!

...And now she's lost perspective.
6.088 Digital and Computational Photography
6.882 Advanced Computational Photography

Panoramas

Frédo Durand
MIT - EECS

Lots of slides stolen from Alyosha Efros, who stole them from Steve Seitz and Rick Szeliski
Why Mosaic?

• Are you getting the whole picture?
  – Compact Camera FOV = 50 x 35°
Why Mosaic?

• Are you getting the whole picture?
  – Compact Camera FOV = 50 x 35°
  – Human FOV = 200 x 135°
Why Mosaic?

• Are you getting the whole picture?
  – Compact Camera FOV = 50 x 35°
  – Human FOV = 200 x 135°
  – Panoramic Mosaic = 360 x 180°
Traditional panoramas
19th century panorama
Mosaics: stitching images together

virtual wide-angle camera
Today

- We assume we know feature correspondences between images
- We seek to align the images into a virtual wider-angle view
- Later lecture: automatic correspondence
Question?
How to do it?

• Basic Procedure
  – Take a sequence of images from the same position
    • Rotate the camera about its optical center (nodal point)
  – 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?
A pencil of rays contains all views. Can generate any synthetic camera view as long as it has the same center of projection!
Nodal point

• When camera is rotated around nodal point, there is no parallax
  – That is, if two 3D points are superimposed for one orientation, they remain superimposed after rotation

• Finding the nodal point is painful
Recap

• When we only rotate the camera (around nodal point) depth does not matter
• It only performs a 2D warp
  – one-to-one mapping of the 2D plane
  – plus of course reveals stuff that was outside the field of view
• Now we just need to figure out this mapping
Other interpretation

• Depth does not matter
• We can pretend that each pixel is at a convenient depth

• Three convenient depth distributions:
  – spherical
  – planar
  – cylindrical

• We focus on planar
  • it makes life more linear
Question?
Aligning images: translation

Translations are not enough to align the images
Which t-form is the right one for warping PP1 into PP2?
e.g. translation, Euclidean, affine, projective

- Translation: 2 unknowns
- Affine: 6 unknowns
- Perspective: 8 unknowns
• 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
Image reprojection

• **Basic question**
  – How to relate 2 images from 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
Recap

• We are looking for the 2D mapping that corresponds to rotation
• We need to understand perspective projection
Simple Perspective Projection

- Project all points to the $z = 1$ plane, viewpoint at the origin:
  - $x' = x/z$
  - $y' = y/z$
Simple Perspective Projection

• Project all points to the $z = 1$ plane, viewpoint at the origin:
  - $x' = x/z$
  - $y' = y/z$

• Can we represent this with a matrix?
  - not directly (division)
  - but we can cheat...
  - add a third coordinate to the result
  - interpret as: we always divide by 3rd coordinate
Homogeneous coordinate

• represent 2D points with 3 numbers
• (x, y, w) represents (x/w, y/w)
• Allows us to represent projective transforms
• Nice thing: projecting onto plane $z=1$ is just the strict interpretation of homogeneous coordinates

• Yes, you can view this as a notation trick
• But math is all about smart notations
3D rotation

• Now we rotate the 3D camera
• What is the new projection?
3D rotation

• Now we rotate the 3D camera
• What is the new projection?

• Rotating the camera is the same as rotating the world in the opposite direction
• To project a \((x,y,z)\) wrt rotated camera:
  – Apply rotation matrix to \((x,yz)\)
  – Apply projection division
Recap

• canonical projection = division by z
• Homogeneous coordinates are a notation trick to encapsulate this
• For other direction, just apply 3D rotation first

• But... this applies only when we know $z$
  – And for panorama stitching, we don’t
  – What are we going to do? Are we in big trouble? Should we give up? By a 3D scanner? Cancel assignment 4?
3D does not matter

- Recall: no parallax
- All points along a line going through the viewpoint will be reprojected the same.
- So just pick $z=1$
  - pretend the geometry is planar
Wrapping it up: Homography

• Projective – mapping between any two PPs with the same center of projection
  – rectangle should map to arbitrary quadrilateral
  – parallel lines aren’t parallel anymore
  – but must preserve straight lines
  – same as: project, rotate, reproject

• called Homography

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

To apply a homography \( H \)

• Compute \( p' = Hp \) (regular matrix multiply)
• Convert \( p' \) from homogeneous to image coordinates (divide by \( w \))
1D homogeneous coordinates

- Add one dimension to make life simpler
- \((x, w)\) represent point \(x/w\)
1D homography

• Reproject to different line
1D homography

- Reproject to different line
1D homography

• Reproject to different line
• Equivalent to rotating 2D points
  ➔ reprojection is linear in homogeneous coordinates
Recap

• Reprojection = homography
• 3x3 matrix in homogeneous coordinate
  – (the matrix can be constrained to be a rotation)

\[
\begin{bmatrix}
wx' \\
wy' \\
p', w
\end{bmatrix} =
\begin{bmatrix}
* & * & * \\
* & * & * \\
* & * & * \\
\end{bmatrix}
\begin{bmatrix}
x \\
y \\
l
\end{bmatrix}
\]
Image warping with homographies

- image plane in front
- black area where no pixel maps to
Questions?
Digression: perspective correction

Rise and fall move the front or back of the camera in a flat plane, like opening or closing an ordinary window. Rise moves the front or back up; fall moves the front or back down.

Shift (like rise and fall) also moves the front or back of the camera in a flat plane, but from side to side in a motion like moving a sliding door.

Tilt tips the front or back of the camera forward or backward around a horizontal axis. Nodding your head yes is a tilt of your face.

Swing twists the front or back of the camera around a vertical axis to the left or right. Shaking your head no is a swing of your face.

From Photography, London et al.
Standing at street level and shooting straight at a building produces too much street and too little building. Sometimes it is possible to move back far enough to show the entire building while keeping the camera level, but this adds even more foreground and usually something gets in the way.
Standing at street level and shooting straight at a building produces too much street and too little building. Sometimes it is possible to move back far enough to show the entire building while keeping the camera level, but this adds even more foreground and usually something gets in the way.

Tilting the whole camera up shows the entire building but distorts its shape. Since the top is farther from the camera than the bottom, it appears smaller; the vertical lines of the building seem to be coming closer together, or converging, near the top. This is named the keystone effect, after the wedge-shaped stone at the top of an arch. This convergence gives the illusion that the building is falling backward—an effect particularly noticeable when only one side of the building is visible.

From Photography, London et al.
CONTROLLING CONVERGING LINES: THE KEYSTONE EFFECT

Standing at street level and shooting straight at a building produces too much street and too little building. Sometimes it is possible to move back far enough to show the entire building while keeping the camera level, but this adds even more foreground and usually something gets in the way.

Tilting the whole camera up shows the entire building but distorts its shape. Since the top is farther from the camera than the bottom, it appears smaller; the vertical lines of the building seem to be coming closer together, or converging, near the top. This is named the keystone effect, after the wedge-shaped stone at the top of an arch. This convergence gives the illusion that the building is falling backward—an effect particularly noticeable when only one side of the building is visible.

To straighten up the converging vertical lines, keep the camera back parallel to the face of the building. To keep the face of the building in focus, make sure the lens is parallel to the camera back. One way to do this is to level the camera and then use the rising front or falling back movements or both.

Another solution is to point the camera upward toward the top of the building, then use the tilting movements—first to tilt the back to a vertical position (which squares the shape of the building), then to tilt the lens so it is parallel to the camera back (which brings the face of the building into focus). The lens and film will end up in the same positions with both methods.

From Photography, London et al.
Tilt-shift lens

- 35mm SLR version
Photoshop version (perspective crop)

+ you control reflection and perspective independently
Question?

- The Ambassadors, Hans Holbein the Younger, 1533
Question?

- The Ambassadors, Hans Holbein the Younger, 1533
To unwarp (rectify) an image

- Find the homography $H$ given a set of $p$ and $p'$ pairs
- How many correspondences are needed?
- We can solve for $H$
  - Find such $H$ that transforms points $p$ into $p'$
Solving for homographies

\[ p' = Hp \]

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

- Can set scale factor \( i=1 \). So, there are 8 unknowns.
- Set up a system of linear equations:
  \[ Ah = b \]
- Where vector of unknowns \( h = [a,b,c,d,e,f,g,h]^T \)

- Note: we do not know \( w \) but we can compute it from \( x \& y \)
  \( w = gx + hy + 1 \)
- The equations are linear in the unknown
Careful: two equations

• Start from $p' = Hp$
• Say that the unknowns are the coefficients of $H$, put them into a vector $h$ of 8 coordinates
• New equation $Ah = b$
Solving for homographies

\[ p' = Hp \]

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

- Can set scale factor $i=1$. So, there are 8 unknowns.
- Set up a system of linear equations:
  - \[ Ah = b \]
  - where vector of unknowns $h = [a,b,c,d,e,f,g,h]^T$
- Need at least 8 eqs, but the more the better…
- Solve for $h$. If overconstrained, solve using least-squares:
  \[ \min\|Ah - b\|^2 \]
- Can be done in Matlab using \"\" command
  - see \"help lmdivide\"
Questions?

- Julian Beever
- e.g. http://users.skynet.be/J.Beever/
  http://www.crystalinks.com/julian_beever.html
Least Squares Example

• Say we have a set of data points \((X_1,X_1'), (X_2,X_2'), (X_3,X_3'),\) etc. (e.g. person’s height vs. weight)

• We want a nice compact formula (line) to predict X’s from Xs:
  
  \[
  Xa + b = X'
  \]

• We want to find a and b

• How many \((X,X')\) pairs do we need?

• \[
  X_1a + b = X_1'
  \]
  
  \[
  X_2a + b = X_2'
  \]

• What if the data is noisy?

\[
\begin{bmatrix}
X_1 & 1 \\
X_2 & 1 \\
X_3 & 1 \\
\vdots & \vdots
\end{bmatrix}
\begin{bmatrix}
a \\
b
\end{bmatrix}
=
\begin{bmatrix}
X_1' \\
X_2' \\
X_3' \\
\vdots
\end{bmatrix}
\]

\[
Ax = B
\]

\[
\text{overconstrained}
\]

\[
\min\|Ax - B\|^2
\]
least square

- $Ax=b$ where $A$ is rectangular (overconstrained)

- Solve $A^TAx = A^Tb$
Questions?
Panoramas

1. Pick one image (red)
2. Warp the other images towards it (usually, one by one)
3. Blend
Recap

• Panorama = reprojection
• 3D rotation ➔ homography
  – Homogeneous coordinates are kewl
• Use feature correspondence
• Solve least square problem
  – Se of linear equations
• Warp all images to a reference one
• Use your favorite blending
Questions?
changing camera center

- Does it still work?
Planar mosaic
Cool applications of homographies

• Oh, Durand & Dorsey
Limitations of 2D Clone Brushing

- Distortions due to foreshortening and surface orientation
Clone brush (Photoshop)

- Click on a reference pixel (blue)
- Then start painting somewhere else
- Copy pixel color with a translation
Perspective clone brush

Oh, Durand, Dorsey, unpublished

- Correct for perspective
- And other tricks
Figure 15: The cars and the street furniture have been removed. This example took less than 10 minutes.
Questions?
Do we have to project onto a plane?
Full Panoramas

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

mosaic Projection Cylinder
Cylindrical projection

- Map 3D point \((X,Y,Z)\) onto cylinder
  
  \[
  (\hat{x}, \hat{y}, \hat{z}) = \frac{1}{\sqrt{X^2+Z^2}} (X, Y, Z)
  \]

- Convert to cylindrical coordinates
  
  \[
  (\sin\theta, h, \cos\theta) = (\hat{x}, \hat{y}, \hat{z})
  \]

- Convert to cylindrical image coordinates
  
  \[
  (\tilde{x}, \tilde{y}) = (f\theta, fh) + (\tilde{x}_c, \tilde{y}_c)
  \]
Cylindrical Projection
Full-view (360°) panoramas
Spherical projection

- 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

\((\tilde{x}, \tilde{y}) = (f \theta, f h) + (\tilde{x}_c, \tilde{y}_c)\)
Spherical Projection
Full-view Panorama
• Radial distortion of the image
  – Caused by imperfect lenses
  – Deviations are most noticeable for rays that pass through the edge of the lens
Radial distortion

- Correct for “bending” in wide field of view lenses

\[
\begin{align*}
\hat{r}^2 &= \hat{x}^2 + \hat{y}^2 \\
\hat{x}' &= \hat{x}/(1 + \kappa_1 \hat{r}^2 + \kappa_2 \hat{r}^4) \\
\hat{y}' &= \hat{y}/(1 + \kappa_1 \hat{r}^2 + \kappa_2 \hat{r}^4) \\
x &= f\hat{x}'/\hat{z} + x_c \\
y &= f\hat{y}'/\hat{z} + y_c
\end{align*}
\]

Use this instead of normal projection
Blending the mosaic

An example of image compositing: the art (and sometime science) of combining images together…
Questions?
M. Uyttendaele, A. Eden, and R. Szeliski.
Eliminating ghosting and exposure artifacts in image mosaics.
M. Uyttendaele, A. Eden, and R. Szeliski.
Eliminating ghosting and exposure artifacts in image mosaics.
Magic: automatic panos

Extensions

- Video
- Additional objects
- Mok’s panomorph
- http://citeseer.ist.psu.edu/cache/papers/cs/20590/
  http:zSzzSzwww.sarnoff.comzSzcareer_movezSztech_paperszSzpdfzSzvisrep95.pdf/kumar95representation.pdf
Software

- http://photocreations.ca/collage/circle.jpg
- http://webuser.fh-furtwangen.de/%7Edersch/
- http://www.ptgui.com/
- http://hugin.sourceforge.net/
- http://epaperpress.com/ptlens/

- http://www.fdrtools.com/front_e.php
• http://www.cs.washington.edu/education/courses/csep576/05wi/readings/szeliskiShum97.pdf
• http://portal.acm.org/citation.cfm?id=218395&dl=ACM&coll=portal
• http://research.microsoft.com/~brown/papers/cvpr05.pdf
• http://citeseer.ist.psu.edu/mann94virtual.html
• http://grail.cs.washington.edu/projects/panovidtex/
• http://research.microsoft.com/vision/visionbasedmodeling/publications/Baudisch-OZCHI05.pdf
• http://www.vision.caltech.edu/lihi/Demos/SquarePanorama.html
• http://graphics.stanford.edu/papers/multi-cross-slits/
View morphing

subject

common view plane

viewpoint 1

viewpoint 2