Panoramas

Calvin and Hobbes

Objects no longer diminish in size with distance.

The laws of perspective have been repealed!

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.
Panoramas and Homographies

Fredo Durand
MIT EECS 6.815/6.865
Piazza
Motivation
Panorama: virtual wide angle
All the way to 360

• http://people.csail.mit.edu/fredo/Panos/
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°
Similar to HDR

- Medium is limited (range, field of view)
- Take multiple images
- Merge
An old idea

This image by the firm of Maison Bonfils depicts the city of Beirut, Lebanon, sometime in the last third of the 19th century. Maison Bonfils was the extraordinarily prolific venture of the French photographer Félix Bonfils (1831-85), his wife Marie-Lydie Cabanis Bonfils (1837-1918), and their son, Adrien Bonfils (1861-1928). The Bonfils moved to Beirut in 1867 and, over the next five decades, their firm produced one of the world's most important bodies of photographic work about the Middle East. Maison Bonfils was known for landscape photographs, panoramas, biblical scenes, and posed “ethnographic” portraits. The family’s marketing acumen and commercial sense helped make their photographs known around the world. “Panoramic” photographs employ a variety of techniques to create a wide angle of view. This “panoramic view” is composed of four aerial photographs set together to give the viewer a broader image than would have been practical with a single photograph.

http://en.wikipedia.org/wiki/File:%D8%A8%D9%8A%D8%B1%D9%88%D8%AA-%D8%A7%D9%84%D9%82%D8%B1%D9%86_%D9%A1%D9%A9.jpg
An old idea

Old panos, modern viewer

- [http://www.eurofresh.se/history/](http://www.eurofresh.se/history/)

Figure 2. The principle of an infinite rotation presentation of a circular panorama (circular room or a computer screen and panorama viewer).
19th century panorama
Traditional panoramas
Overview
Today

- The user gives us 4 correspondences
- We reproject one image to match the other one
- Creates a wider angle view
Later

- Automagic correspondences
  - corner detection
  - patch descriptor

- Nice blending
  - smooth transition
  - 2 scale
Overview

• Today: Manual panorama
  - perspective and image reprojection
  - Homogenous coordinates and homographies
  - Inferring homography from correspondences

• Automatic panorama

• Nice blending
Reprojection
Virtual wide angle

• Take N images in different directions
• Deduce the image that would have been taken by a wider angle lens
• ...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!
Entrance pupil

• Often wrongly called 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 entrance pupil 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
  - Still useful for spherical panos

Tuesday, March 13, 12
Question?
Aligning images

- We have established that pairs of images from the same viewpoint can be aligned through a simple 2D spatial transformation (warp).
- What kind of transformation?
Aligning images: translation

Translations are not enough to align the images
Recap

- We are looking for the 2D mapping that corresponds to a 3D rotation of the camera
- We need to understand perspective projection
Projective geometry
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
    - see next slide...
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
- Homogenous coordinates are central in computer graphics and machine vision
3D rotation

• We observe point $x, y, z$ with camera PP1
• Now we rotate the 3D camera to PP2
• What is the new 2D 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 \(R\) to \((x,y,z)\)
  - 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? Buy a 3D scanner? Cancel assignment 6?
Camera rotation: unknown z

- Same thing but use \((x', y', 1)\) instead of \((x, y, z)\)
  - Rotate \((x', y', 1)\) by \(R\)
  - Perform division
3D rotation

• Same thing but use \((x', y', 1)\) instead of \((x, y, z)\)
  – Rotate \((x', y', 1)\) by \(R\)
  – Perform division

• Makes sense: depth does not matter, all points along a light ray project to the same point.
  – We arbitrarily (but conveniently) choose the point at depth 1
1D homogeneous coordinates

- Add one dimension to make life simpler
- \((x, w)\) represent point \(x/w\)
Other illustration: 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
Questions?
Homogenous coordinates
Homogenous coordinates

- Representation of 2D points using 3 coordinates

- \((x, y, w)\) represents \((x/w, y/w)\)
  - Homogenous points are interpreted as Euclidean coordinates by dividing by the last coordinate

- Motivations:
  - perspective transforms
  - Make translations linear
  - Duality with hyperplanes.
Scale invariant

- All points \((wx, wy, w)\) for any non-zero \(w\) represent the same Euclidean point
Extra profit

• Homogenous coordinates allow us to represent 2D translations as matrices

\[
\begin{pmatrix}
1 & 0 & t_y \\
0 & 1 & t_x \\
0 & 0 & 1
\end{pmatrix}
\]
Homographies
Recap

• To reproject a 3D view:
  - pretend image is a 3D plane at distance 1: points x,y,1
  - Apply a 3D rotation (3x3 matrix)
  - Reproject by dividing by the new z (often called w)
2D homogenous version

- 2D points represented as homogenous coordinates \((x, y, 1)\)
- Mapped into the other image by a 3x3 matrix \(H\), into homogenous coordinates \((x', y', w')\)
- get Euclidean coordinates by dividing by \(w'\)
3D vs. homogenous: the same

- 3 coordinates because viewed as plane in 3D
- 3x3 matrix because rotation
- Divide because perspective

- weird extra coordinate
- 3x3 homography matrix
- divide because it’s the rule
Wrapping it up: Homography

• Projective mapping between any two planes
• represented as 3x3 matrix in homogenous coordinates
  – corresponds to the 3D rotation

\[
\begin{bmatrix}
w x' \\
w y' \\
p', w
\end{bmatrix} = \begin{bmatrix}
* & * & * \\
* & * & * \\
* & * & *
\end{bmatrix}
\begin{bmatrix}
x \\
y \\
p
\end{bmatrix}
\]

To apply a homography \( H \)

• Compute \( p' = Hp \) (regular matrix multiply)
• Convert \( p' \) from homogeneous to image coordinates (divide by \( w \))
Wrapping it up: Homography

- rectangles map to arbitrary quadrilateral
- parallel lines aren’t parallel anymore
- but straight lines remain straight

- same as: project, rotate, reproject
Recap

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

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

– Homographies are defined up to a scale
– $H$ and $kH$ represent the same 2D transformation
– because $(w'x', w'y', w')$ and $(kw'x', kw'y', kw')$ represent the same point

\[
\begin{bmatrix}
wx' \\
w'x' \\
p'
\end{bmatrix} = \begin{bmatrix}
* & * & * \\
* & * & * \\
* & * & *
\end{bmatrix}
\begin{bmatrix}
x \\
y \\
l
\end{bmatrix}
\]
General pairs of plane

- Homographies can project from any plane to any plane
Image warping with homographies

- Homography so that image is parallel to floor

- Homography so that image is parallel to right wall

- Black area where no pixel maps to
Demo with projector
Questions?
Solving for homographies
Goal

• Given correspondences
• Find homography matrix $H$ that maps the $p_i$ to $p'_i$
Warning

• In what follows I use the order y, x
  - At least I will try

• This will make life easier for pset 6
Homography equation

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

• We are given pairs of corresponding points
  - \( x, y, x', y' \) are known

• Unknowns: matrix coefficients and \( w' \)
Homography equation

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

• We are given pairs of corresponding points
  - x, y, x', y' are known

• Unknowns: matrix coefficients and w'
  - but w' is easy to get:
    \[ w' = gy + hx + i \]
\[
\begin{pmatrix}
  a & b & c \\
  d & e & f \\
  g & h & i
\end{pmatrix}
\begin{pmatrix}
  y \\
  x \\
  1
\end{pmatrix}
=
\begin{pmatrix}
  y' & w' \\
  x' & w'
\end{pmatrix}
\]

\[w' = gy + hx + i\]

- For a pair of points \((x, y)\to(x', y')\) we have
  \[ay + bx + c = y'(gy + hx + i)\]
  \[dy + ex + f = x'(gy + hx + i)\]
For a pair of points \((x, y)\rightarrow(x', y')\) we have

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

\[w' = gy + hx + i\]

- For a pair of points \((x, y)\rightarrow(x', y')\) we have
  \[ay + bx + c = y' (gy + hx + i)\]
  \[dy + ex + f = x' (gy + hx + i)\]

- Unknowns: \(a, b, c, d, e, f, g, h, i\)
  - Linear!
How many pairs?

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

• Each correspondence pair gives us two equations

\[
ay + bx + c = y'(gy + hx + i)
\]
\[
dy + ex + f = x'(gy + hx + i)
\]

• How many unknowns?
  - 9
  - but H is defined up to scale. Four pairs are enough!
Forming the linear system

• We have 4x2 linear equations in our 8 unknowns

• Represent as a matrix system $Ax = B$:

$$
\begin{pmatrix}
 a \\
 b \\
 c \\
 d \\
 f \\
 g \\
 h \\
 i
\end{pmatrix}
= 

B
$$

• Now we need to fill matrix $A$ and vector $B$
Forming the matrix

\[ ay + bx + c = y'(gy + hx + i) \]
\[ dy + ex + f = x'(gy + hx + i) \]

\[
\begin{pmatrix}
  a & b & c & d & f & g & h & i \\
\end{pmatrix}
\begin{pmatrix}
  a \\
  b \\
  c \\
  d \\
  f \\
  g \\
  h \\
  i \\
\end{pmatrix}
= \cdot 
\]
Forming the matrix

\[ \begin{align*}
ay + bx + c &= y'(gy + hx + i) \\
fy + ex + f &= x'(gy + hx + i)
\end{align*} \]

\[
\begin{pmatrix}
a & b & c & d & f & g & h & i \\
y & x & 1 & 0 & 0 & 0 & -yy' & -xy' - y'
\end{pmatrix}
= \begin{pmatrix}
a \\
b \\
c \\
d \\
f \\
g \\
h \\
i
\end{pmatrix}
= 0
\]

I’ll let you do the x case for pset 6
Recap

- We have four pairs of points

- Looking for homography H

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

- Formed a big 8x9 linear system \( Ax = 0 \)
  - where \( x \) is the 9 homography coefficients
Solve for scale invariance

• We know that there exists a full family of solutions $kH$, for any non-zero $k$

• i.e., if $(a, b, c, d, e, f, g, h, i)$ is a solution, so is $(ka, kb, kc, kd, ke, kf, kg, kh, ki)$

• Adding more correspondences won’t help
Dirty solution

- Hope that $i$ is not 0
- Set it arbitrarily to 1
  - either create an 8x8 matrix
  - or add a last row that says $i$ should be 1

- In practice $i$ is rarely zero
  - But it’s still dirty

- You can use this solution for pset 6
Cleaner solution

• Use SVD
• The singular vector with singular value 0 is a solution

• See your favorite linear algebra textbook
Questions?

- Julian Beever
- e.g. http://users.skynet.be/J.Beever/
  http://www.crystalinks.com/julian_beever.html
Questions?
Recap
Manual linear panoramas

- Create a virtual wide angle view from 2 images
- Choose one image as reference
- User gives 4 correspondences
- Deduce homography matrix
- Reproject (warp) second image into first one

\[ \mathbf{H} \]
Under the hood

• Homogenous coordinates
  - encode 2D points with 3 coordinates \((x, y, w)\)
  - represents Euclidean points \((x/w, y/w)\)
  - make it easy to express perspective and to go between 3D and 2D

• Homography
  - 3x3 matrix on homogenous coordinates
  - represent any perspective mapping of a plane
  - to apply a homography to a 2D point \(x, y\):
    compute \(H(x, 1, y)^T\) and divide by third coordinate
Multiple images

1. Pick one image (red)
2. Warp the other images towards it (usually, one by one)
3. blend
Homography warp

- For each output pixel
- compute input location with homography matrix
  - $p' = Hp$, followed by division
- copy pixel color (with appropriate antialiasing)
Later

- Automagic correspondences
  - corner detection
  - patch descriptor

- Nice blending
  - smooth transition
  - 2 scale

- Other projections
  - spherical, cylindrical, miniplanets

http://designedbynatalie.com/tag/panoramas/
Questions?
changing camera center

• Does it still work?
Only for Planar mosaic

Image Compositing for Tele-Reality

1. Introduction
2. Previous work
3. Basic imaging equations
4. Planar image compositing
5. Panoramic
6. Piecewise-planar scenes
7. Scenes with arbitrary depth
8. 3-D model recovery
9. Applications
10. Discussion & conclusions
Questions?
Perspective correction
Digression: perspective correction

• Perspective makes parallel lines converge
• Sometimes objectionable
• In particular, architecture photography
  – photo looks more formal if verticals stay vertical

• But if lines are parallel to the image plane, parallel lines don’t converge
Perspective correction

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.

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.

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

• Equivalent to using a wider-angle view and cropping

From Photography, London et al.
Perspective correction

- The lens creates an image bigger than the sensor
- Tilting and shifting aligns the sensor with the part we want

From Photography, London et al.
• The two images are related by a homography

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
Cool applications of homographies

• With Mok Oh.
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?
Other application: View morphing

• We want to morph between two views of the same object
• Standard morphing won’t give a realistic result
  – Because it uses linear interpolation of image locations
  – With perspective & 3D, points don’t move linearly
View morphing

• Seitz & Dyer
  

• Interpolation consistent with 3D view interpolation

Figure 1: View morphing between two images of an object taken from two different viewpoints produces the illusion of physically moving a virtual camera.
View morphing

- Given view 1 and view 2 (and no depth!)
- Compute intermediate views consistent with 3D

viewpoint 1

subject

viewpoint 2
View morphing

- Reproject to a common view plane parallel to the 2 viewpoints: homographies
View morphing

• Similar triangles: everything becomes linear
• i.e. interpolate location of features linearly

\[
\text{subject}
\]

\[
\text{common view plane}
\]

P1

P2

viewpoint 1

viewpoint 2

Tuesday, March 13, 12
View morphing

- Similar triangles: everything becomes linear
- Independent of depth

subject

Q1

Q2

common view plane

viewpoint 1

viewpoint 2
View morphing recap

- When views are reprojected onto a common plane parallel to the line between the viewpoints: Interpolating the viewpoint results in linear interpolation of feature locations

- i.e.
  - for viewpoint \((1-t)V_1+tV_2\)
  - a 3D point that was at \(P_1\) from \(V_1\) and \(P_2\) from \(V_2\) is now at \((1-t)P_1+tP_2\)

- That is, a simplistic warp works
View morphing

• Prewarp with a homography to "pre-align" images
• So that the two views are parallel
  – Because linear interpolation works when views are parallel

Figure 4: View Morphing in Three Steps. (1) Original images $I_0$ and $I_1$ are prewarped to form parallel views $\hat{I}_0$ and $\hat{I}_1$. (2) $\hat{I}_s$ is produced by morphing (interpolating) the prewarped images. (3) $\hat{I}_s$ is postwarped to form $I_s$. 
Figure 6: View Morphing Procedure: A set of features (yellow lines) is selected in original images $I_0$ and $I_1$. Using these features, the images are automatically prewarped to produce $\hat{I}_0$ and $\hat{I}_1$. The prewarped images are morphed to create a sequence of in-between images, the middle of which, $\hat{I}_{0.5}$, is shown at top-center. $\hat{I}_{0.5}$ is interactively postwarped by selecting a quadrilateral region (marked red) and specifying its desired configuration, $Q_{0.5}$, in $\hat{I}_{0.5}$. The postwarps for other in-between images are determined by interpolating the quadrilaterals (bottom).
Figure 10: Image Morphing Versus View Morphing. Top: image morph between two views of a helicopter toy causes the in-between images to contract and bend. Bottom: view morph between the same two views results in a physically consistent morph. In this example the image morph also results in an extraneous hole between the blade and the stick. Holes can appear in view morphs as well.
Figure 7: Facial View Morphs. Top: morph between two views of the same person. Bottom: morph between views of two different people. In each case, view morphing captures the change in facial pose between original images $I_0$ and $I_1$, conveying a natural 3D rotation.
Figure 9: Mona Lisa View Morph. Morphed view (center) is halfway between original image (left) and it’s reflection (right).
Extensions

• Video
• Additional objects
• Mok’s panomorph
• http://www.sarnoff.com/products_services/vision/tech_papers/kumarvb.pdf
• http://www.cs.huji.ac.il/~peleg/papers/pami00-manifold.pdf
• http://www.cs.huji.ac.il/~peleg/papers/cvpr00-rectified.pdf
• http://www.cs.huji.ac.il/~peleg/papers/cvpr05-dynmos.pdf
• http://www.robots.ox.ac.uk/~vgg/publications/papers/schaffalitzky02.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
Refs

- http://portal.acm.org/citation.cfm?id=218395&dl=ACM&coll=portal
- http://citeseer.ist.psu.edu/mann94virtual.html
- http://www.vision.caltech.edu/lihi/Demos/SquarePanorama.html