Lecture 19: Motion

Tuesday, Nov 20

• Review Problem set 3
  – Dense stereo matching
  – Sparse stereo matching
  – Indexing scenes

Effect of window size

W = 3  W = 20

Want window large enough to have sufficient intensity variation, yet small enough to contain only pixels with about the same disparity.

Sources of error in correspondences

• Low-contrast / textureless image regions
• Occlusions
• Camera calibration errors
• Poor image resolution
• Violations of brightness constancy (specular reflections)
• Large motions

Sparse matching

Indexing scenes
So far:
- Features and filters
- Grouping, segmentation, fitting
- Multiple views, stereo, matching
- Recognition and learning

So far: Features and filters
Transforming and describing images; textures and colors

So far: Grouping
Clustering, segmentation, fitting; what parts belong together?

So far: Multiple views
Multi-view geometry and matching, stereo

So far: Recognition and learning
Shape matching, recognizing objects and categories, learning techniques
Motion and tracking

Tracking objects, video analysis, low level motion

Outline

- Motion field and parallax
- Optical flow, brightness constancy
- Aperture problem
- Constraints on image motion

Uses of motion

- Analyzing motion can be useful for
  - Estimating 3D structure
  - Segmentation of moving objects
  - Tracking objects, features over time

Image sequences

A digital video is a sequence of images (frames) captured over time.

Now we consider image as a function of both position and time.

Types of motion in video

- Considering rigid objects – they can rotate and translate in the scene.
- Motion may be due to
  - Movement in scene
  - Movement of camera (ego motion)
- Geometrically equivalent, however illumination effects can make one appear different than the other.
Motion field and apparent motion

Point in the scene

Velocity vector

Projection of scene point

Goal: estimate apparent motion, the u and v values at each pixel x,y, i.e., u(x,y), v(x,y)

Motion field equations

\[ \mathbf{p} = f \frac{\mathbf{P}}{Z} \]

Take the time derivative of both sides:

\[ \mathbf{v} = f \frac{Z \mathbf{V} - \mathbf{V} \mathbf{P}}{Z^2} \]

Motion field equations

- Translational part of image motion depends on (unknown) depth of the point
- Motion parallax: image motion is a function of both motion in space and depth of each point.
Motion parallax

- http://psych.hanover.edu/KRANTZ/MotionParallax/MotionParallax.html

Translational motion

Radial motion field if $T_z$ nonzero.
Length of flow vectors inversely proportional to depth of 3d point

Motion vs. Stereo: Similarities

- Both involve solving
  - Correspondence: disparities, motion vectors
  - Reconstruction

Motion vs. Stereo: Differences

- Motion:
  - Uses velocity: consecutive frames must be close to get good approximate time derivative
  - 3d movement between camera and scene not necessarily single 3d rigid transformation
- Whereas with stereo:
  - Could have any disparity value
  - View pair separated by a single 3d transformation
Optical flow problem

Goal: estimate apparent motion, the u and v values at each pixel x, y, i.e., u(x, y), v(x, y)

How to estimate pixel motion from image H to image I?
- Solve pixel correspondence problem
  - given a pixel in H, look for nearby pixels of the same color in I

What might make it difficult to estimate apparent motion?

- Brightness constancy
- Spatial coherence
- Temporal smoothness
Motion constraints

- To recover optical flow, we need some constraints (assumptions)
  - Brightness constancy: in spite of motion, image measurement in small region will remain the same
  - Spatial coherence: assume nearby points belong to the same surface, thus have similar motions, so estimated motion should vary smoothly.
  - Temporal smoothness: motion of a surface patch changes gradually over time.

Brightness constancy equation

\[ \frac{dl}{dt} = 0 \]

Total derivative: \( x \) and \( y \) are also functions of time \( t \)

\[ \begin{align*}
\frac{\partial I}{\partial x} + \frac{\partial I}{\partial y} + \frac{\partial I}{\partial t} & = 0 \\
\end{align*} \]

Rewritten as:

\[ \nabla I^T \mathbf{u} + I_t = 0. \]

This is exactly true in the limit as \( u \) and \( v \) go to 0, for infinitesimal motions.

Aperture problem

\[ \nabla I^T \mathbf{u} + I_t = 0. \]

According to brightness constancy constraint, motions that satisfy the optical flow equation are only constrained to lie along a line in \( u, v \) space.

Aperture problem

\[ \nabla I^T \mathbf{u} + I_t = 0. \]

- Brightness constancy equation: single equation, two unknowns; infinitely many solutions.

- Can only compute projection of actual flow vector \([u, v]\) in the direction of the image gradient, that is, in the direction normal to the image edge.
  - Flow component in gradient direction determined
  - Flow component parallel to edge unknown.
Aperture problem


Solving the aperture problem

How to get more equations for a pixel?
- Basic idea: impose additional constraints
  - most common is to assume that the flow field is smooth locally
  - one method: pretend the pixel’s neighbors have the same \((u,v)\)
    - if we use a 5x5 window, that gives us 25 equations per pixel

\[
0 = I_x(p_0) + \nabla f(p_0) \cdot [u \ v]
\]

\[
\begin{bmatrix}
I_x(p_1) & \ldots & I_x(p_25) \\
I_y(p_1) & \ldots & I_y(p_25)
\end{bmatrix}
\begin{bmatrix}
u \\
v
\end{bmatrix}
= \begin{bmatrix}
I_x(p_{25}) \\
I_y(p_{25})
\end{bmatrix}
\]

RGB version

How to get more equations for a pixel?
- Basic idea: impose additional constraints
  - most common is to assume that the flow field is smooth locally
  - one method: pretend the pixel’s neighbors have the same \((u,v)\)
    - if we use a 5x5 window, that gives us 25**3 equations per pixel

\[
0 = I_x(p_{0}[0,1,2]) + \nabla f(p_{0}[0,1,2]) \cdot [u \ v]
\]

Lucas-Kanade flow

Prob: we have more equations than unknowns
- minimum least squares solution given by solution \((d)\) of:

\[
\begin{bmatrix}
A^T A & \ast \\
\ast & \ast
\end{bmatrix}
\begin{bmatrix}
d \\
b
\end{bmatrix}
= \begin{bmatrix}
A^T b
\end{bmatrix}
\]

- The summations are over all pixels in the \(K \times K\) window
- This technique was first proposed by Lucas & Kanade (1981)
Windows and apparent motion

Conditions for solvability

- Optimal \((u, v)\) satisfies Lucas-Kanade equation

\[
\begin{bmatrix}
\sum I_x I_x & \sum I_x I_y \\
\sum I_y I_x & \sum I_y I_y
\end{bmatrix}
\begin{bmatrix}
u \\
v
\end{bmatrix} =
\begin{bmatrix}
\sum I_x I_1 \\
\sum I_y I_1
\end{bmatrix},
\]

When is this solvable?

- \(A^T A\) should be invertible
- \(A^T A\) should not be too small
  - eigenvalues \(\lambda_1\) and \(\lambda_2\) of \(A^T A\) should not be too small
- \(A^T A\) should be well-conditioned
  - \(\lambda_1 / \lambda_2\) should not be too large \((\lambda_1 = \text{larger eigenvalue})\)

Edge

- gradient strong in one direction
- large \(\lambda_1\), small \(\lambda_2\)

Low texture region

- gradients have small magnitude
- small \(\lambda_1\), small \(\lambda_2\)

High textured region

- gradients are different, large magnitudes
- large \(\lambda_1\), large \(\lambda_2\)

Good conditions for solving flow

- Recall Harris corner detection
- Good feature windows to track in time can be detected independently in a single frame.
Revisiting the small motion assumption

Is this motion small enough?
- Probably not—it's much larger than one pixel (2nd order terms dominate)
- How might we solve this problem?

Coarse-to-fine optical flow estimation

Example use of optical flow: Motion Paint
Use optical flow to track brush strokes, in order to animate them to follow underlying scene motion.

Reduce the resolution!

Slide by Steve Seitz, UW

http://www.fxguide.com/article333.html

Coming up
- Problem set 4 due 12/4
More on motion
- Multiple motions and segmentation
- Tracking
- SfM