Lecture 19: Motion

Tuesday, Nov 20
• Review Problem set 3
  – Dense stereo matching
  – Sparse stereo matching
  – Indexing scenes
Effect of window size

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

Figures from Li Zhang
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
<table>
<thead>
<tr>
<th>Date</th>
<th>Assignment</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Slides fullpage (faces part 2, detection, boosting). Slides fullpage (SVMs, unsupervised model learning).</td>
</tr>
<tr>
<td>11/20</td>
<td>Motion, optical flow, tracking&lt;br&gt;Trucco &amp; Verri handout.</td>
</tr>
<tr>
<td>11/27</td>
<td></td>
</tr>
<tr>
<td>11/29</td>
<td></td>
</tr>
<tr>
<td>12/4</td>
<td></td>
</tr>
<tr>
<td>12/8</td>
<td>Wrap-up&lt;br&gt;Pset 4 due 12/4</td>
</tr>
<tr>
<td>12/13</td>
<td>Final exam&lt;br&gt;Graduate students' reviews and extensions due</td>
</tr>
</tbody>
</table>
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.

Figure by Martial Hebert, CMU
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

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

Figure by Martial Hebert, CMU
Motion field equations

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

Figure by Martial Hebert, CMU
Motion

Velocity of scene point described as

\[ \mathbf{V} = -\mathbf{T} - \mathbf{\omega} \times \mathbf{P} \]

- Translational motion
- Angular velocity

\[ V_x = -T_x - \omega_y Z + \omega_z Y \]
\[ V_y = -T_y - \omega_z X + \omega_x Z \]
\[ V_z = -T_z - \omega_x Y + \omega_y X \]

Using this and the motion field equation, can give expressions for the components of the image velocity \( \mathbf{v} \)...
Motion field equations

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

Take the time derivative of both sides:

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

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

\[ V_x = -T_x - \omega_y Z + \omega_z Y \]
\[ V_y = -T_y - \omega_z X + \omega_x Z \]
\[ V_z = -T_z - \omega_x Y + \omega_y X \]

**Translational components**

\[ v_x = \frac{T_x x - T_x f}{Z} \]
\[ v_y = \frac{T_y y - T_y f}{Z} \]

**Rotational components**

\[ \omega_y f + \omega_z y + \frac{\omega_x xy}{f} - \frac{\omega_y x^2}{f} \]
\[ \omega_x f + \omega_z x + \frac{\omega_y xy}{f} - \frac{\omega_x y^2}{f} \]

Trucco & Verri Section 8.2.1
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.

\[
\begin{align*}
  v_x &= \frac{T_z x - T_x f}{Z} \\
  v_y &= \frac{T_z y - T_y f}{Z} \\
  \omega_x f + \omega_y + \frac{\omega_{xy} y}{f} - \frac{\omega_{xy} x}{f} \\
  \omega_y f + \omega_x + \frac{\omega_{xy} x}{f} - \frac{\omega_{xy} y^2}{f}
\end{align*}
\]

Trucco & Verri Section 8.2.1
Motion parallax

- [http://psych.hanover.edu/KRANTZ/MotionParallax/MotionParallax.html](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

Figure 1.2: Two images taken from a helicopter flying through a canyon and the computed optical flow field.

Figure from Michael Black, Ph.D. Thesis
Translational motion

Radial motion field if $T_z$ nonzero.

Length of flow vectors inversely proportional to depth of 3d point

Figure from Michael Black, Ph.D. Thesis
Translational motion

Radial motion field if $T_z$ nonzero.

Length of flow vectors inversely proportional to depth of 3d point

Figure from Michael Black, Ph.D. Thesis
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., $u(x,y)$, $v(x,y)$
Optical flow problem

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 $I$

Adapted from Steve Seitz, UW
• What might make it difficult to estimate apparent motion?
Brightness constancy

Figure 1.5: Data conservation assumption. The highlighted region in the right image looks roughly the same as the region in the left image, despite the fact that it has moved.

Figure by Michael Black
Spatial coherence

Figure 1.7: Spatial coherence assumption. Neighboring points in the image are assumed to belong to the same surface in the scene.
Temporal smoothness

Figure 1.8: Temporal continuity assumption. A patch in the image is assumed to have the same motion (constant velocity, or acceleration) over time.

Figure by Michael Black
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

\[ I(x, y, t) = I(x + \delta x, y + \delta y, t + \delta t) = I(x + u\delta t, y + v\delta t, t + \delta t) \]

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

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

\[ \frac{\partial I}{\partial x} \frac{dx}{dt} + \frac{\partial I}{\partial y} \frac{dy}{dt} + \frac{\partial I}{\partial t} \]

**Spatial image gradients**

**Temporal derivatives, \( u \) and \( v \)**
Brightness constancy equation

shorthand: \[ I_x = \frac{\partial I}{\partial x} \]

Rewritten as:
\[ I_x u + I_y v + I_t = 0. \]

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

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

shorthand: \( I_x = \frac{\partial I}{\partial x} \)

Rewritten as:

\[
I_x u + I_y v + I_t = 0.
\]

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

Which terms are measurable from images?
How many unknowns in this equation?
Aperture problem

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

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

Figure from Michael Black's Ph.D. Thesis
Aperture problem

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

• 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

Slide by Steve Seitz, UW
Aperture problem
Aperture problem

Barber Pole

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!

\[
0 = I_t(p_i) + \nabla I(p_i) \cdot [u \ v]
\]

\[
\begin{bmatrix}
I_x(p_1) & I_y(p_1) \\
I_x(p_2) & I_y(p_2) \\
\vdots & \vdots \\
I_x(p_{25}) & I_y(p_{25})
\end{bmatrix}
\begin{bmatrix}
  u \\
  v
\end{bmatrix}
= 
\begin{bmatrix}
  I_t(p_1) \\
  I_t(p_2) \\
  \vdots \\
  I_t(p_{25})
\end{bmatrix}
\]

\[
A^{25\times2} \quad d^{2\times1} \quad b^{25\times1}
\]

Slide by Steve Seitz, UW
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!

\[
0 = I_t(p_i)[0, 1, 2] + \nabla I(p_i)[0, 1, 2] \cdot [u \ v]
\]

\[
\begin{bmatrix}
I_x(p_1)[0] & I_y(p_1)[0] \\
I_x(p_1)[1] & I_y(p_1)[1] \\
I_x(p_1)[2] & I_y(p_1)[2] \\
\vdots & \vdots \\
I_x(p_{25})[0] & I_y(p_{25})[0] \\
I_x(p_{25})[1] & I_y(p_{25})[1] \\
I_x(p_{25})[2] & I_y(p_{25})[2]
\end{bmatrix}
\begin{bmatrix}
[u] \\
[v]
\end{bmatrix}
= 
\begin{bmatrix}
I_t(p_1)[0] \\
I_t(p_1)[1] \\
I_t(p_1)[2] \\
\vdots \\
I_t(p_{25})[0] \\
I_t(p_{25})[1] \\
I_t(p_{25})[2]
\end{bmatrix}
\]

\[
A \quad 75 \times 2 \\
d \quad 2 \times 1 \\
b \quad 75 \times 1
\]
Lucas-Kanade flow

Prob: we have more equations than unknowns

\[ A \begin{bmatrix} d \\ 25x2 \begin{bmatrix} 2x1 & 25x1 \end{bmatrix} b \end{bmatrix} \rightarrow \text{minimize } ||Ad - b||^2 \]

Solution: solve least squares problem

- minimum least squares solution given by solution (in d) of:

\[ (A^T A) \begin{bmatrix} d \\ 2x1 \end{bmatrix} = A^T b \begin{bmatrix} 2x1 \end{bmatrix} \]

\[
\begin{bmatrix}
\sum I_x I_x & \sum I_x I_y \\
\sum I_x I_y & \sum I_y I_y
\end{bmatrix}
\begin{bmatrix}
u \\
v
\end{bmatrix} = -
\begin{bmatrix}
\sum I_x I_t \\
\sum I_y I_t
\end{bmatrix}
\]

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

Slide from Trevor Darrell, MIT
Conditions for solvability

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

\[
\begin{bmatrix}
\sum I_x I_x & \sum I_x I_y \\
\sum I_x I_y & \sum I_y I_y
\end{bmatrix}
\begin{bmatrix}
u \\
v
\end{bmatrix}
\quad = 
\begin{bmatrix}
\sum I_x I_t \\
\sum I_y I_t
\end{bmatrix}
\begin{bmatrix}
A^T \\
A^T
\end{bmatrix}
\]

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 (\(\lambda_1\) = larger eigenvalue)

Slide by Steve Seitz, UW
Edge

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

Adapted from Steve Seitz, UW
Low texture region

- gradients have small magnitude
- small $\lambda_1$, small $\lambda_2$

Slide by Steve Seitz, UW
High textured region

- gradients are different, large magnitudes
- large $\lambda_1$, large $\lambda_2$

Slide by Steve Seitz, UW
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 (2\textsuperscript{nd} order terms dominate)
- How might we solve this problem?

Slide by Steve Seitz, UW
Reduce the resolution!

Slide by Steve Seitz, UW
Coarse-to-fine optical flow estimation

Gaussian pyramid of image H

Gaussian pyramid of image I

u=1.25 pixels

u=2.5 pixels

u=5 pixels

u=10 pixels

image H

image I
Coarse-to-fine optical flow estimation

Image I

Gaussian pyramid of image I

Image H

Gaussian pyramid of image H

run iterative L-K

warp & upsample

run iterative L-K
Example use of optical flow: Motion Paint

Use optical flow to track brush strokes, in order to animate them to follow underlying scene motion.

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

• Problem set 4 due 12/4

More on motion
• Multiple motions and segmentation
• Tracking
• SfM