Xu Zhao @ Shanghai Jiao Tong university

Lecture 4: Motion and Optical Flow

Contents

- Motion field
- Optical flow
- Lucas-Kanade algorithm

Motion and perceptual organization

- Sometimes, motion is the only and foremost cue.

Motion and Gestalt laws of grouping

slides from Grauman

Motion and perceptual organization

- Even “impoverished” motion data can evoke a strong percept

Motion and perceptual organization

What can the “motion" do?

- Estimating 3D structure
- Segmenting objects based on motion cues
- Learning dynamical models
- Recognizing events and activities
- Improving video quality (motion stabilization)

Case: motion segmentation

- Motion segmentation: segment the video into multiple ***coherently*** moving objects

Motion: video view

- A video is a sequence of frames captured over time
- Now our image data is a function of space (x, y) and time (t)

Methods for motion estimation

- **Sparse** feature based: point and line correspondences like we have seen before.
  - Extract visual features (corners, textured areas) and track them
  - Sparse motion fields, but possibly robust tracking
  - Suitable especially when image motion is large
- **Direct dense** motion estimation: pixel level
  - Directly recover image motion from spatio-temporal image brightness variations
  - Global motion parameters directly recovered without an intermediate feature motion calculation
  - Dense motion fields, but more sensitive to appearance variations
  - Suitable for video and when image motion is small (< 10 pixels)

Motion field

- The motion field is the projection of the 3D scene motion into the image.

Motion field and parallax

- **P**(t) is a moving 3D point
- Velocity of scene point: **V** = d**P**/dt
- **p**(t) = (x(t),y(t)) is the projection of **P** in the image
- Apparent velocity **v** in the image: given by components v_x = dx/dt and v_y = dy/dt
- These components are known as the *motion field* of the image

Motion field and parallax

$$\mathbf{V} = (V_x, V_y, V_Z) \quad \mathbf{p} = f\frac{\mathbf{P}}{Z}$$

To find image velocity **v**, differentiate **p** with respect to *t* (using quotient rule):

$$\mathbf{v} = f\frac{Z\mathbf{V} - V_z\mathbf{P}}{Z^2}$$

$$v_x = \frac{fV_x - V_z x}{Z} \qquad v_y = \frac{fV_y - V_z y}{Z}$$

Image motion is a function of both the 3D motion (V) and the depth of the 3D point (Z)

Quotient rule:
D(f/g) = (g f' – g' f)/g^2

Motion field and parallax

Pure translation: V is constant everywhere

$$v_x = \frac{fV_x - V_z x}{Z}$$

$$v_y = \frac{fV_y - V_z y}{Z}$$

$$\mathbf{v} = \frac{1}{Z}(\mathbf{v}_0 - V_z \mathbf{p}),$$

$$\mathbf{v}_0 = (fV_x, fV_y)$$

Vz is nonzero: Every motion vector points toward (or away from) v_0, the vanishing point of the translation direction

Vz is zero: Motion is parallel to the image plane, all the motion vectors are parallel

Motion parallax

- The length of the motion vectors is inversely proportional to the depth *Z*. Points closer to the camera move more quickly across the image plane.

$$v_x = \frac{fV_x - V_z x}{Z}$$

$$v_y = \frac{fV_y - V_z y}{Z}$$

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

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

Motion field + camera motion

Zoom out **Zoom in** **Pan right to left**

Optical flow

- **Definition**: optical flow is the *apparent* motion of brightness patterns in the image
- Ideally, optical flow would be the same as the motion field
- Have to be careful: apparent motion can be caused by lighting changes without any actual motion

Apparent motion ~= motion field

Figure from Horn book

Estimating optical flow

- Given two subsequent frames, estimate the apparent motion field between them
- Key assumptions
  - **Brightness constancy:** projection of the same point looks the same in every frame
  - **Small motion:** points do not move very far
  - **Spatial coherence:** points move like their neighbors.

$I(x,y,t-1)$

$I(x,y,t)$

Spatial coherence

Assumption

* Neighboring points in the scene typically belong to the same surface and hence typically have similar motions.
* Since they also project to nearby points in the image, we expect spatial coherence in image flow.

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.

The brightness constancy constraint

Brightness Constancy Equation:

$$I(x,y,t-1) = I(x+u(x,y), y+v(x,y), t)$$

Linearizing (assuming small *(u,v)*) using Taylor series expansion:

$$I(x,y,t-1) \approx I(x,y,t) + I_x \cdot u(x,y) + I_y \cdot v(x,y)$$

So

shorthand: $I_x = \frac{\partial I}{\partial x}$

$$I_x \cdot u + I_y \cdot v + I_t \approx 0$$

- How many equations and unknowns per pixel?
  One equation, two unknowns

$$I_x \cdot u + I_y \cdot v + I_t = 0$$

- Intuitively, what does this constraint mean?

$$\nabla I \cdot (u, v) + I_t = 0$$

- The component of the flow perpendicular to the gradient (i.e., parallel to the edge) is unknown

If (u, v) satisfies the equation,
so does ($u+u'$, $v+v'$) if $\nabla I \cdot (u', v') = 0$

$$I_x u + I_y v + I_t = 0$$

$$\nabla I^T \mathbf{u} = -I_t$$

$$\mathbf{u} = \begin{bmatrix} u \\ v \end{bmatrix} \quad \nabla I = \begin{bmatrix} I_x \\ I_y \end{bmatrix}$$

At a single image pixel, we get a line:

$$I_x u + I_y v = -I_t$$

We get at most "Normal Flow" – with one point we can only detect movement perpendicular to the brightness gradient. Solution is to take a patch of pixels around the pixel of interest.

The aperture problem

The aperture problem

The barber pole illusion

http://en.wikipedia.org/wiki/Barberpole_illusion

Barber pole illusion

Solving the aperture problem (grayscale image)

- How to get more equations for a pixel?
- **Spatial coherence constraint:** pretend the pixel's neighbors have the same (u,v)
  - If we use a 5x5 window, that gives us 25 equations per pixel

$$0 = I_t(\mathbf{p_i}) + \nabla I(\mathbf{p_i}) \cdot [u \;\; v]$$

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

$$\underset{25\text{x}2}{A} \;\; \underset{2\text{x}1}{d} = \underset{25\text{x}1}{b}$$

RGB version

$$0 = I_t(\mathbf{p_i})[0,1,2] + \nabla I(\mathbf{p_i})[0,1,2] \cdot [u \;\; v]$$

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

75x2 75x1

Note that RGB is not enough to disambiguate because R, G & B are correlated just provides better gradient

Solving the aperture problem

- **Problem**: we have more equations than unknowns

$$\underset{25\text{x}2}{A} \quad \underset{2\text{x}1}{d} = \underset{25\text{x}1}{b} \longrightarrow \text{minimize } \|Ad - b\|^2$$

Solution: solve least squares problem

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

$$\underset{2\text{x}2}{(A^T A)} \; \underset{2\text{x}1}{d} = \underset{2\text{x}1}{A^T b}$$

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

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

Conditions for solvability

- When is this solvable?
  - $\mathbf{A}^T\mathbf{A}$ should be invertible
  - $\mathbf{A}^T\mathbf{A}$ should not be too small
    - eigenvalues λ_1 and λ_2 of $\mathbf{A}^T\mathbf{A}$ should not be too small
  - $\mathbf{A}^T\mathbf{A}$ should be well-conditioned
    - $\dfrac{\lambda_1}{\lambda_2}$ should not be too large (λ_1= larger eigenvalue)

Edge

- Gradients very large or very small
- large λ_1, small λ_2 $\sum \nabla I (\nabla I)^T$

Low-texture region

- Gradients very large or very small
- Small λ_1, small λ_2 $\sum \nabla I (\nabla I)^T$

High-texture region

- Gradients very large or very small
- large λ_1, large λ_2 $\sum \nabla I (\nabla I)^T$

The aperture problem resolved

The aperture problem resolved

Observation

- This is a two image problem BUT
  - Can measure sensitivity by just looking at one of the images!
  - This tells us which pixels are easy to track, which are hard
  - very useful when we do feature tracking...

Errors in Lukas-Kanade

- What are the potential causes of errors in this procedure?
  - Suppose A^TA is easily invertible
  - Suppose there is not much noise in the image
- When our assumptions are violated
  - Brightness constancy is **not** satisfied
  - The motion is **not** small
  - A point does **not** move like its neighbors
  - window size is too large
  - what is the ideal window size?

- Is this motion small enough?
  - Probably not—it's much larger than one pixel
  - How might we solve this problem?

Small motion assumption

- When the motion is not small enough, how might we solve this problem? Reduce the resolution!

Coarse-to-fine optical flow estimation

Results

Lucas-Kanade without pyramids

Fails in areas of large motion

Results

From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Optical flow

- Motion or appearance?
  - small motion assumption
  - brightness constancy

Methods for motion estimation

- **Direct dense** motion estimation: pixel level
  - Directly recover image motion from spatio-temporal image brightness variations
  - Global motion parameters directly recovered without an intermediate feature motion calculation
  - Dense motion fields, but more sensitive to appearance variations
  - Suitable for video and when image motion is small (< 10 pixels)
- **Sparse** feature based: point and line correspondences like we have seen before.
  - Extract visual features (corners, textured areas) and track them
  - Sparse motion fields, but possibly robust tracking
  - Suitable especially when image motion is large

Motion magnification

- http://people.csail.mit.edu/mrub/vidmag/
- Hao-Yu Wu, Michael Rubinstein, Eugene Shih, John Guttag, Frédo Durand, William T. Freeman **Eulerian Video Magnification for Revealing Subtle Changes in the World** ACM Transactions on Graphics, Volume 31, Number 4 (Proc. SIGGRAPH), 2012
- Ce Liu, Antonio Torralba, William T. Freeman, Frédo Durand, Edward H. Adelson. **Motion Magnification.** ACM Transactions on Graphics, Volume 24, Number 3 (Proc. SIGGRAPH), 2005