EE 417 Introduction to Computer Vision

/ EE 569 3D vision

Motion and Optical Flow

Instructor: Associate Prof. Mehmet Keskinöz

Communication Theory &Technologies (CTT) Group,
Electronics Engineering Program,
Computer Science and Engineering,
Cyber-Security Program
Faculty of Engineering and Natural Sciences

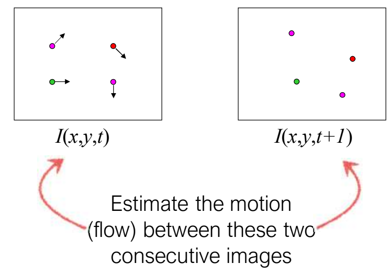
Email: keskinoz@sabanciuniv.edu

Sabancı Üniversitesi

1

Optical Flow

(Problem definition)



2

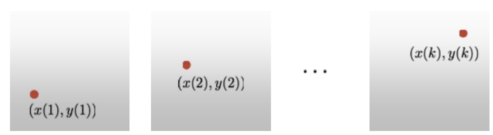
- Key assumptions of Lucas-Kanade Tracker
  - **Brightness constancy:** projection of the same point looks the same in every frame Lambertian assumption
  - **Small motion:** points do not move very far
  - **Spatial coherence:** points move like their neighbors Rigid body motion

3

Assumption 1

Brightness constancy

Scene point moving through image sequence

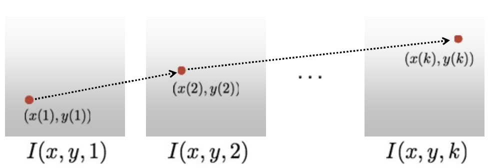


4

Assumption 1

Brightness constancy

Scene point moving through image sequence

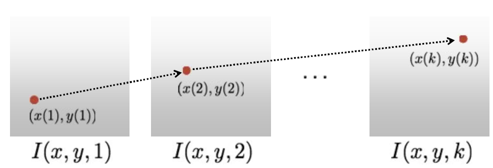


5

Assumption 1 Lambertian assumption

Brightness constancy

Scene point moving through image sequence



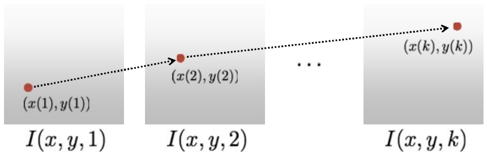
Assumption:Brightness of the point will remain the same

6

Assumption 1

Brightness constancy

Scene point moving through image sequence



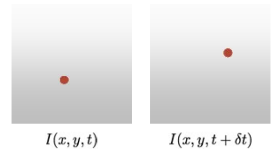
Assumption:Brightness of the point will remain the same

$$I(x(t),y(t),t) = \underbrace{C}_{\text{constant}}$$

7

Assumption 2

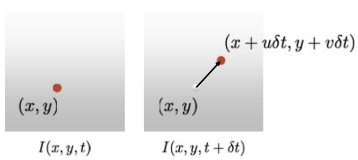
Small motion



8

Assumption 2

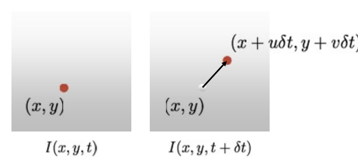
Small motion



9

Assumption 2

Small motion

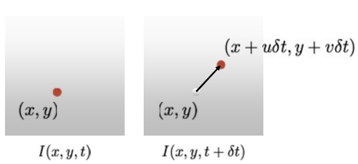


Optical flow (velocities): (u,v) Displacement: $(\delta x, \delta y) = (u\delta t, v\delta t)$

10

Assumption 2

Small motion



Optical flow (velocities): (u,v) Displacement: $(\delta x, \delta y) = (u\delta t, v\delta t)$

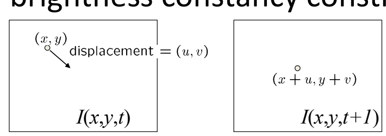
For a ***really small space-time step***…

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

… the brightness between two consecutive image frames is the same

11

The brightness constancy constraint



- Brightness Constancy Equation:

$$\boxed{I(x,y,t) = I(x+u,y+v,t+1)}$$

Take Taylor expansion of $I(x+u, y+v, t+1)$ at (x,y,t) to linearize the right side:

Image derivative along x Difference over frames

$$\boxed{I(x+u,y+v,t+1) \approx I(x,y,t) + \boxed{I_x} \cdot u + I_y \cdot v + \boxed{I_t}}$$

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

So: $I_x \cdot u + I_y \cdot v + I_t \approx 0$ $\quad \to \nabla^T I \cdot \begin{bmatrix} u \\ v \end{bmatrix} + I_t = 0$

12

(putting the math aside for a second…)

What do the term of the
brightness constancy equation represent?

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

13

(putting the math aside for a second…)

What do the term of the
brightness constancy equation represent?

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

Image gradients
(at a point p)

14

(putting the math aside for a second…)

What do the term of the
brightness constancy equation represent?

flow velocities

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

Image gradients
(at a point p)

15

(putting the math aside for a second…)

What do the term of the
brightness constancy equation represent?

flow velocities

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

Image gradients
(at a point p)

temporal gradient

How do you compute these terms?

16

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

How do you compute …

$$I_x = \frac{\partial I}{\partial x} \quad I_y = \frac{\partial I}{\partial y}$$

spatial derivative

17

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

How do you compute …

$$I_x = \frac{\partial I}{\partial x} \quad I_y = \frac{\partial I}{\partial y}$$

spatial derivative

Forward difference
Sobel filter
Scharr filter
…

18

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

How do you compute …

$$I_x = \frac{\partial I}{\partial x} \quad I_y = \frac{\partial I}{\partial y}$$

spatial derivative

$$I_t = \frac{\partial I}{\partial t}$$

temporal derivative

Forward difference
Sobel filter
Prewitt filter
…

19

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

How do you compute …

$$I_x = \frac{\partial I}{\partial x} \quad I_y = \frac{\partial I}{\partial y}$$

spatial derivative

$$I_t = \frac{\partial I}{\partial t}$$

temporal derivative

Forward difference
Sobel filter
Scharr filter
…

frame differencing

20

Frame differencing

t

1	1	1	1	1
1	1	1	1	1
1	10	10	10	10
1	10	10	10	10
1	10	10	10	10
1	10	10	10	10

-

$t+1$

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	10	10	10
1	1	10	10	10
1	1	10	10	10

=

$$I_t = \frac{\partial I}{\partial t}$$

0	0	0	0	0
0	0	0	0	0
0	9	9	9	9
0	9	0	0	0
0	9	0	0	0
0	9	0	0	0

(example of a forward difference)

21

Example:

t

3 x 3 patch

1	1	1	1	1
1	1	1	1	1
1	10	10	10	10
1	10	10	10	10
1	10	10	10	10
1	10	10	10	10

$t+1$

1	1	1	1	1
1	1	1	1	1
1	1	1	1	1
1	1	10	10	10
1	1	10	10	10
1	1	10	10	10

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

-	0	0	0	-
-	0	0	0	-
-	9	0	0	-
-	9	0	0	-
-	9	0	0	-
-	9	0	0	-

-1 0 1

$$I_y = \frac{\partial I}{\partial y}$$

-	-	-	-	-
0	0	0	0	0
0	9	9	9	9
-	0	0	0	0
0	0	0	0	0
-	-	-	-	-

-1
0
1

$$I_t = \frac{\partial I}{\partial t}$$

0	0	0	0	0
0	0	0	0	0
0	9	9	9	9
0	9	0	0	0
0	9	0	0	0
0	9	0	0	0

22

The brightness constancy constraint

Can we use this equation to recover image motion (u,v) at each pixel?

$$\nabla^T I \cdot \begin{bmatrix} u \\ v \end{bmatrix} + I_t = 0$$

- How many equations and unknowns per pixel?
  - One equation (this is a scalar equation!), two unknowns (u,v)

The component of the motion perpendicular to the gradient (i.e., parallel to the edge) cannot be measured

If (u, v) satisfies the equation,
so does ($u+u'$, $v+v'$) if

$$\nabla^T I \cdot \begin{bmatrix} u' \\ v' \end{bmatrix} = 0$$

23

Optical Flow

- Integrate around over image patch

$$E_b(\mathbf{u}) = \sum_{W(x,y)} [\nabla I^T(x,y,t)\mathbf{u}(x,y) + I_t(x,y,t)]^2$$

- Solve

$$\nabla E_b(\mathbf{u}) = 2 \sum_{W(x,y)} \nabla I(\nabla I^T \mathbf{u} + I_t)$$

$$= 2 \sum_{W(x,y)} \left(\begin{bmatrix} I_x^2 & I_x I_y \\ I_x I_y & I_y^2 \end{bmatrix} \mathbf{u} + \begin{bmatrix} I_x I_t \\ I_y I_t \end{bmatrix} \right)$$

$$\begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix} \mathbf{u} + \begin{bmatrix} \sum I_x I_t \\ \sum I_y I_t \end{bmatrix} = 0$$

$$G\mathbf{u} + \mathbf{b} = 0$$

24

Optical Flow

$$\boxed{\mathbf{u} = -G^{-1}\mathbf{b}}$$

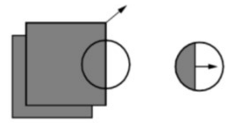
$$G = \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix}$$

Conceptually:

rank(G) = 0 blank wall problem
rank(G) = 1 aperture problem
rank(G) = 2 enough texture – good feature candidates

25

Aperture Problem



26

The barber pole illusion

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

27

Point Feature Extraction

$$G = \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix}$$

- Compute eigenvalues of G
- If smallest eigenvalue σ of G is bigger than τ - mark pixel as candidate feature point

- Alternatively feature quality function (Harris Corner Detector)

28

Wide Baseline Matching

29

Region based Similarity Metric

- Sum of squared differences

$$SSD(h) = \sum_{\tilde{\mathbf{x}} \in W(\mathbf{x})} \|I_1(\tilde{\mathbf{x}}) - I_2(h(\tilde{\mathbf{x}})\|^2$$

- Normalize cross-correlation

$$NCC(h) = \frac{\sum_{W(\mathbf{x})}(I_1(\tilde{\mathbf{x}}) - \bar{I}_1)(I_2(h(\tilde{\mathbf{x}})) - \bar{I}_2))}{\sqrt{\sum_{W(\mathbf{x})}(I_1(\tilde{\mathbf{x}}) - \bar{I}_1)^2 \sum_{W(\mathbf{x})}(I_2(h(\tilde{\mathbf{x}})) - \bar{I}_2)^2)}}$$

- Sum of absolute differences

$$SAD(h) = \sum_{\tilde{\mathbf{x}} \in W(\mathbf{x})} |I_1(\tilde{\mathbf{x}}) - I_2(h(\tilde{\mathbf{x}}))|$$

30