Image Processing & Vision

Lecture 10: Motion Estimation

Hak Gu Kim

hakgukim@cau.ac.kr

Immersive Reality & Intelligent Systems Lab (IRIS LAB)

Graduate School of Advanced Imaging Science, Multimedia & Film (GSAIM)

Chung-Ang University (CAU)

22 May 2023

Topics

- Motion Estimation
  - Optical Flow
  - Lucas-Kanade method
  - Horn-Shunck method

***Note:** Many of these slides in this course were adapted from Computer Vision at CMU (16-385) and UBC (CPSC425)

Topics

- Motion Estimation
  - Optical Flow
  - Lucas-Kanade method
  - Horn-Shunck method

***Note:** Many of these slides in this course were adapted from Computer Vision at CMU (16-385) and UBC (CPSC425)

Motion Estimation

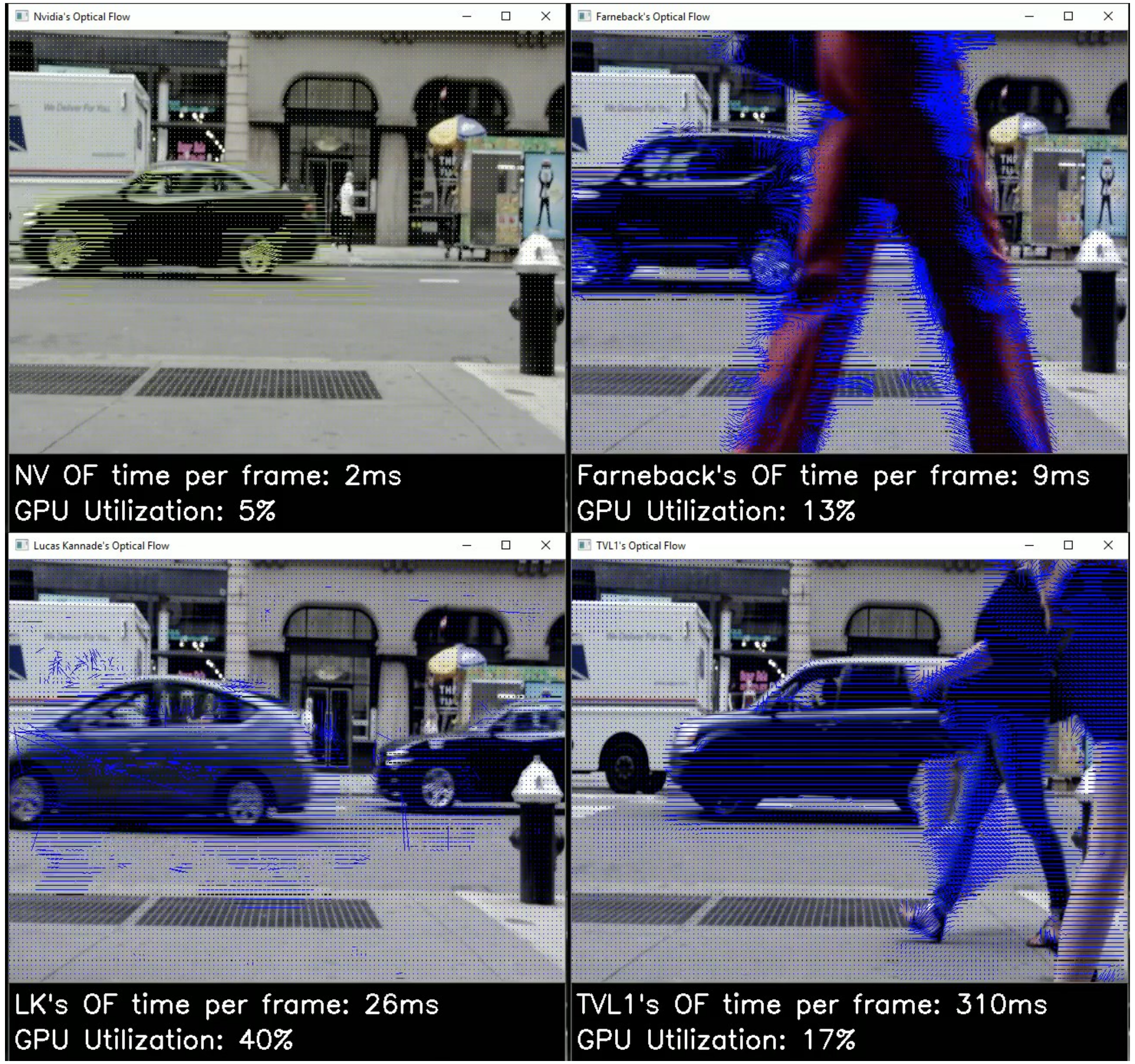
- **Problem:**

— Determine how objects (and/or the camera itself) move in the 3D world

- **Key Idea:**

— Images acquired as a function of time provide additional constraint

— Formulate motion analysis as finding point correspondences over time



https://developer.nvidia.com/blog/opencv-optical-flow-algorithms-with-nvidia-turing-gpus/

Optical Flow

- **Optical flow** is the apparent motion of brightness patterns in the image

- **Applications**

— Image and video stabilization in digital cameras, camcorders

— Motion-compensated video compression

— Motion segmentation

— Image registration

— Action recognition

Optical flow (motion vector)

Optical flow visualization

Optical Flow: **Problem Definition**

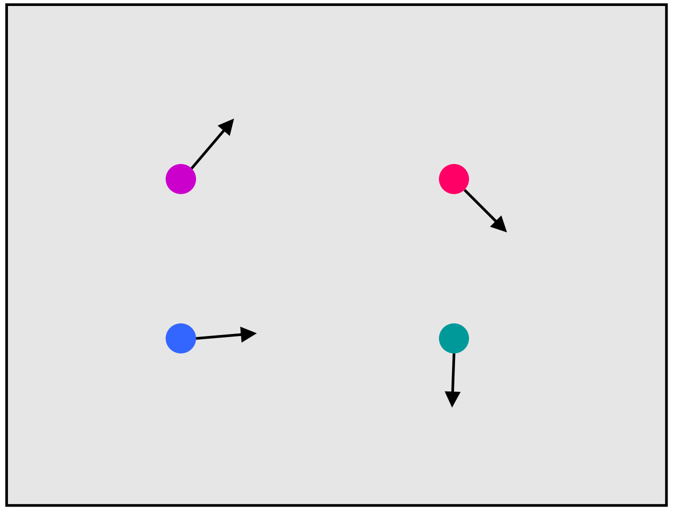
- **Problem Definition**

— Given two consecutive image frames, estimate the motion of each pixel

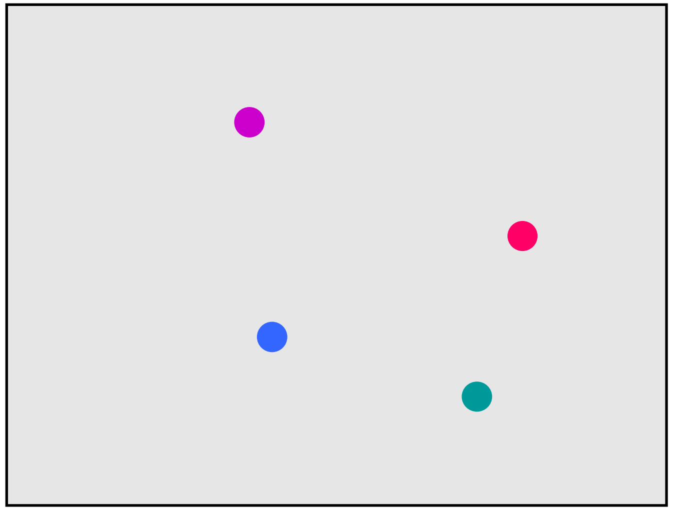
- **Assumptions:**

— Brightness constancy

— Small motion



$I(x, y, t)$



$I(x, y, t')$

Optical Flow: **Key Assumptions**

- **Brightness Constancy**

— Brightness constancy for intensity images

— It allows for pixel to pixel comparison (not image feature comparison)

- **Small Motion**

— Pixels only move a little bit

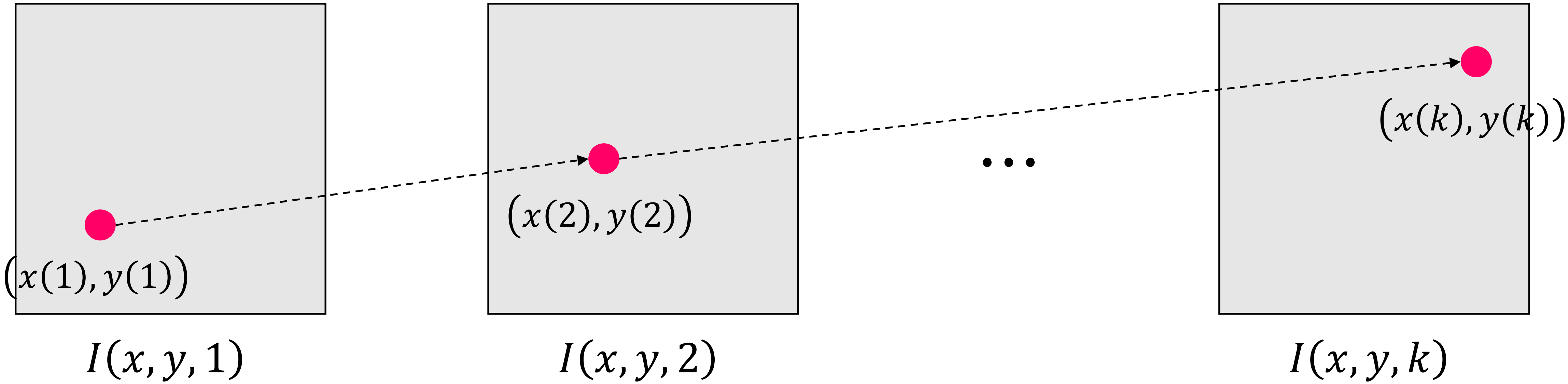
— It can be formulated as a linearization of the brightness constancy constraint

- **Approach:** Look for nearby pixels with the same color

Key Assumptions: ① Brightness Constancy

- Scene point moves through image sequence
- Brightness of the pixel point remains the same (constant, C)

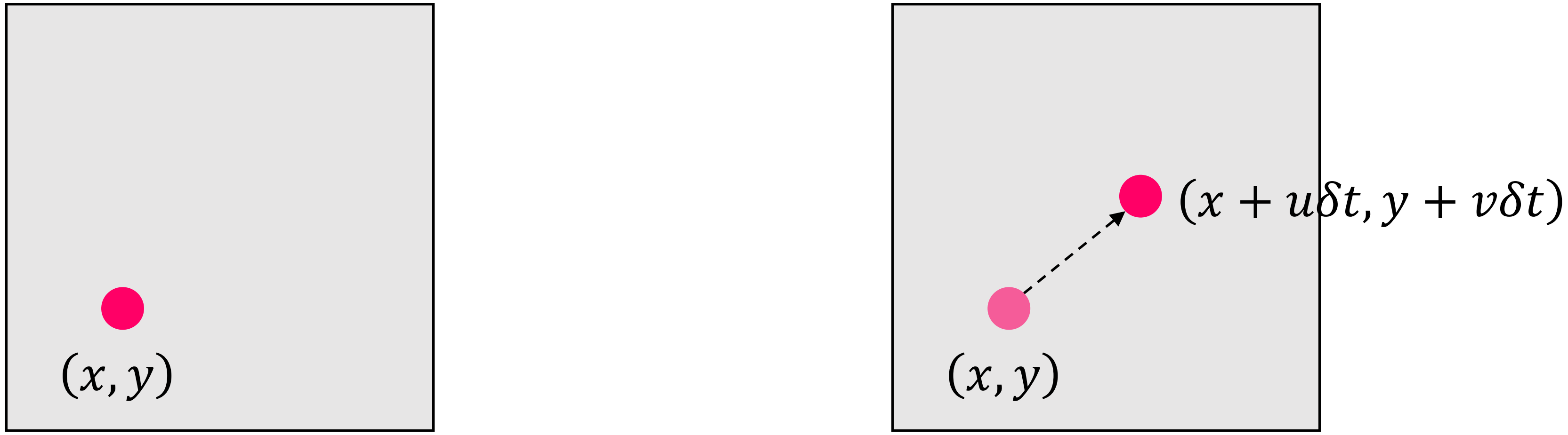
$$I(x(t), y(t), t) = C \qquad where\ \ t = 1, \cdots, k$$



Key Assumptions: ② Small Motion

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

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



$I(x, y, t)$ $\qquad$ $I(x, y, t + \delta t)$

- For small space-time step, brightness of a point is the same

Optical Flow: **Constraint Equation**

- **Optical Flow Constraint Equation:**

$$\frac{dI(x,y,t)}{dt} = \frac{\partial I}{\partial x}\frac{dx}{dt} + \frac{\partial I}{\partial y}\frac{dy}{dt} + \frac{\partial I}{\partial t} = 0$$

- If the time step is really small, we can linearize the intensity function (and motion is really-small … think less than a pixel)

Optical Flow: **Constraint Equation**

- Multivariable Taylor Series Expansion (1st order approximation)

$$f(x,y) \approx f(a,b) + f_x(a,b)(x-a) - f_y(a,b)(y-b)$$

- **Derivation of Optical Flow Constraint:**

$$I(x,y,t) + \frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t = I(x,y,t)$$: Assuming small motion

$$\frac{\partial I}{\partial x}\delta x + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t = 0$$: Divide by δt
Take limit $\delta t \to 0$

$$\frac{\partial I}{\partial x}\frac{dx}{dt} + \frac{\partial I}{\partial y}\frac{dy}{dt} + \frac{\partial I}{\partial t} = 0$$: Brightness Constancy Equation

Optical Flow: **Constraint Equation**

- Brightness Constancy Equation

$$\frac{\partial I}{\partial x}\frac{dx}{dt} + \frac{\partial I}{\partial y}\frac{dy}{dt} + \frac{\partial I}{\partial t} = 0 \xrightarrow{\text{Shorthand notation}} I_x u + I_y v + I_t = 0$$

— Image gradients: I_x and I_y

— Flow velocities: u and v

— Temporal gradient: I_t

Computing Optical Flow Equation

- Brightness Constancy Equation

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

— Image gradients: I_x and I_y $\longrightarrow$ Sobel filter, Canny edge detection, *etc.*

— Flow velocities: u and v

— Temporal gradient: I_t $\longrightarrow$ Frame difference

Example: Image Gradients & Temporal Gradient

$I(x, y, t)$

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

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

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	-

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

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

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

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

Computing Optical Flow Equation

- Brightness Constancy Equation

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

— Image gradients: I_x and I_y $\longrightarrow$ Sobel filter, Canny edge detection, *etc.*

— Flow velocities: u and v $\longrightarrow$ *Unknown*

— Temporal gradient: I_t $\longrightarrow$ Frame difference

Topics

- Motion Estimation

— Optical Flow

— Lucas-Kanade method

— Horn-Shunck method

Lucas-Kanade Method

- **Observations:**

— The 2D motion (u, v) at a given point (x, y) has two degrees of freedom

— The partial derivatives, I_x, I_y, and I_t, provide one constraint

— The 2D motion (u, v) cannot be determined locally from I_x, I_y, and I_t alone

- **Idea:**

— Obtain additional local constraint by computing the partial derivatives I_x, I_y, and I_t in a window centered at the given (x, y)

→ **Constant Flow:** Nearby pixels are likely have the same optical flow

B. D. Lucas and T. Kanade, An Iterative Image Registration Technique with An Application to Stereo Vision, 1981

Lucas-Kanade Method

- Considering all n points in the window

$$I_{x_1}u + I_{y_1}v = -I_{t_1}$$
$$I_{x_2}u + I_{y_2}v = -I_{t_2}$$
$$\vdots$$
$$I_{x_n}u + I_{y_n}v = -I_{t_n}$$

- It can be written as the matrix equation form, $\mathbf{Ax} = \mathbf{b}$:

$$\mathbf{A} = \begin{bmatrix} I_{x_1} & I_{y_1} \\ I_{x_2} & I_{y_2} \\ \vdots & \vdots \\ I_{x_n} & I_{y_n} \end{bmatrix} \qquad \mathbf{x} = \begin{bmatrix} u \\ v \end{bmatrix} \qquad \mathbf{b} = -\begin{bmatrix} I_{t_1} \\ I_{t_2} \\ \vdots \\ I_{t_n} \end{bmatrix}$$

Lucas-Kanade Method

- Standard least squares solution:

$$\hat{\mathbf{x}} = (\mathbf{A}^T\mathbf{A})^{-1}\mathbf{A}^T\mathbf{b}$$

- Again we provided that u and v are the same in all equations and provided that the rank of $\mathbf{A}^T\mathbf{A}$ is 2 (so that the required inverse exists)

$$\mathbf{A}^T\mathbf{A} = \begin{bmatrix} I_{x_1} & I_{x_2} & \cdots & I_{x_n} \\ I_{y_1} & I_{y_2} & \cdots & I_{y_n} \end{bmatrix} \begin{bmatrix} I_{x_1} & I_{y_1} \\ I_{x_2} & I_{y_2} \\ \vdots & \vdots \\ I_{x_n} & I_{y_n} \end{bmatrix} = \begin{bmatrix} \sum_{\mathbf{p}\in W} I_x^2 & \sum_{\mathbf{p}\in W} I_x I_y \\ \sum_{\mathbf{p}\in W} I_x I_y & \sum_{\mathbf{p}\in W} I_y^2 \end{bmatrix}$$

— which is identical to the covariance matrix $\mathbf{C}$ in Harris corner detection

Summary: **Lucas-Kanade**

- A dense method to compute motion (u, v) at every location in an image

- **Key Assumptions:**

— Motion is slow enough and smooth enough that differential methods apply (i.e., that the partial derivatives I_x, I_y, and I_t are well-defined)

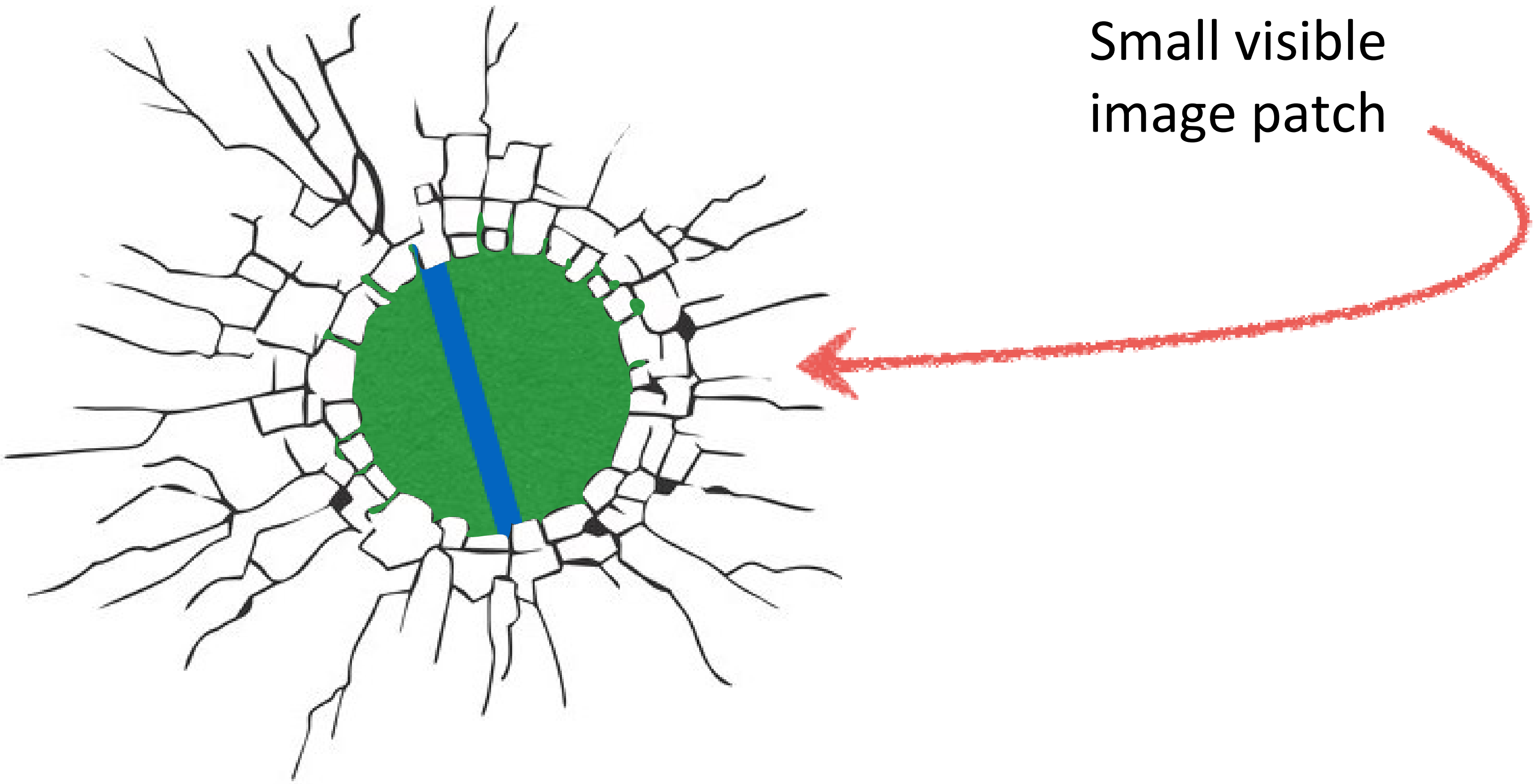
— The optical flow constraint equation holds

— A window size W is chosen so that motion (u, v) is constant in the window

— A window size W is chosen so that the rank of $\mathbf{A}^T\mathbf{A}$ is 2 for the window

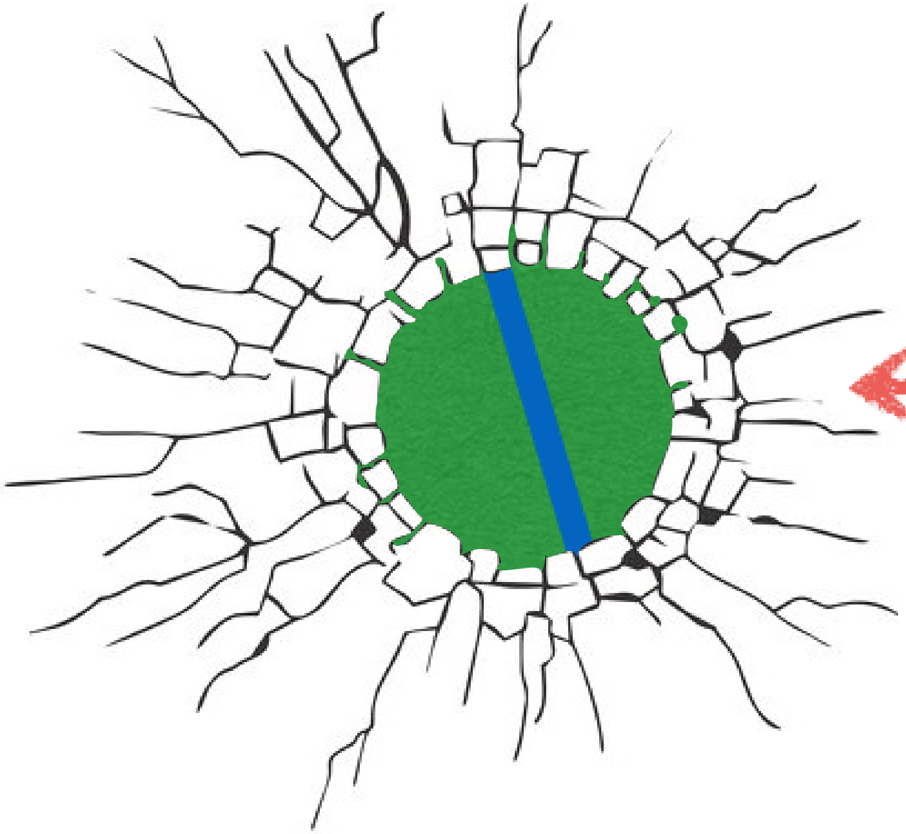
Aperture Problem

- In which direction is the line moving?



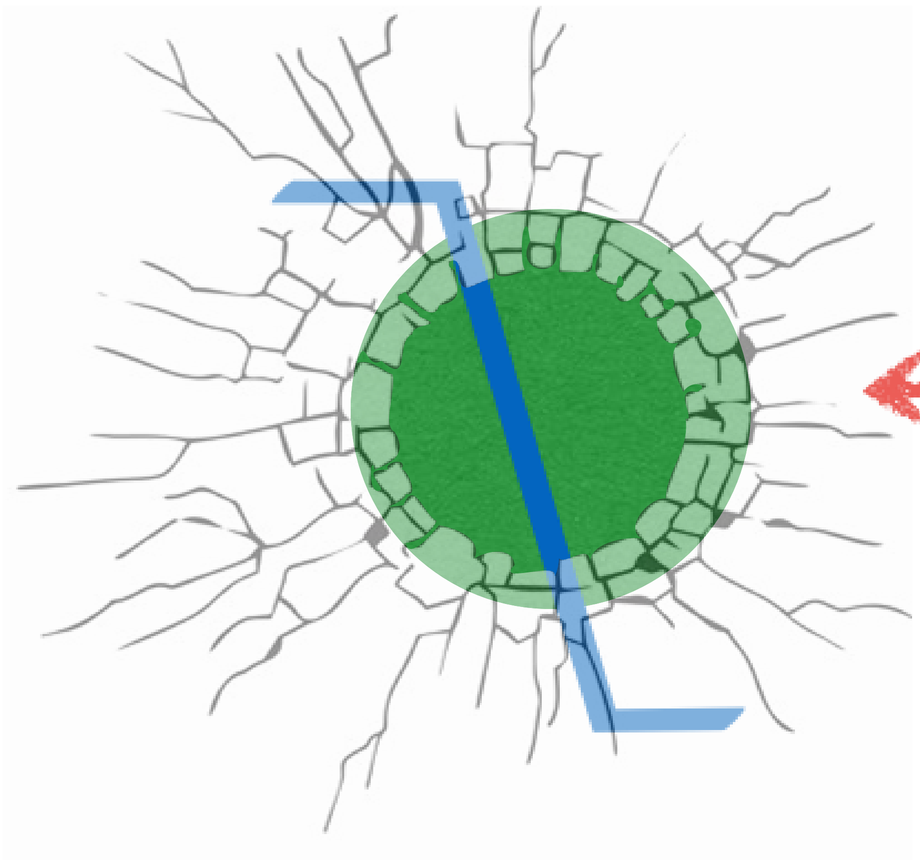
Aperture Problem

- In which direction is the line moving?



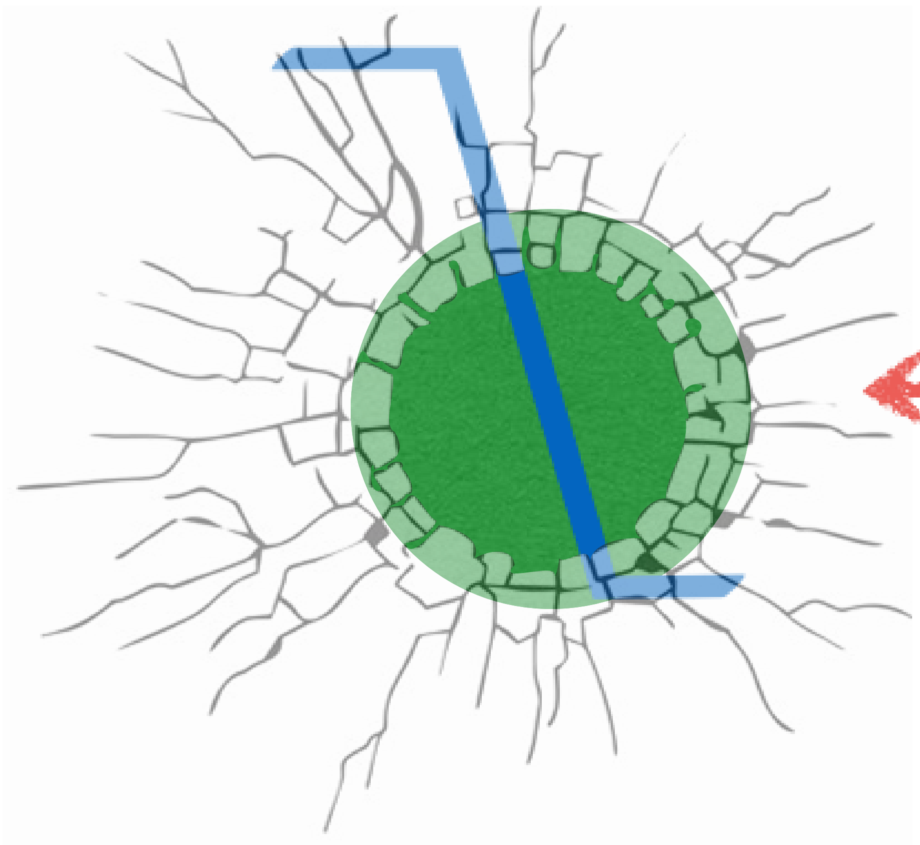
Small visible
image patch

Aperture Problem



Small visible
image patch

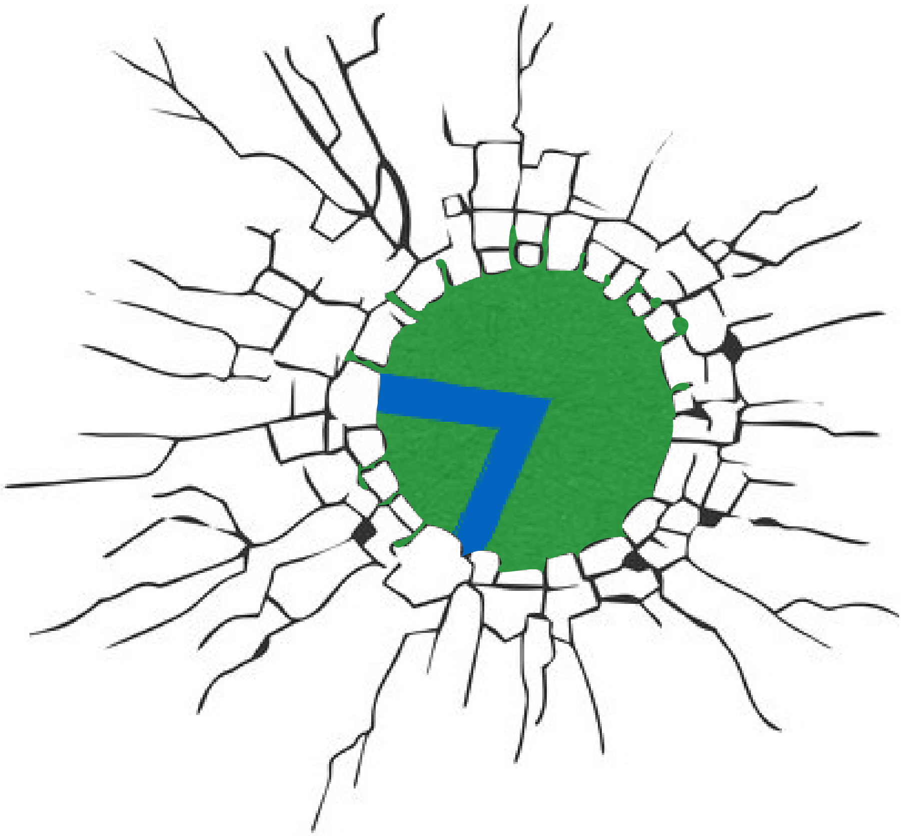
Aperture Problem



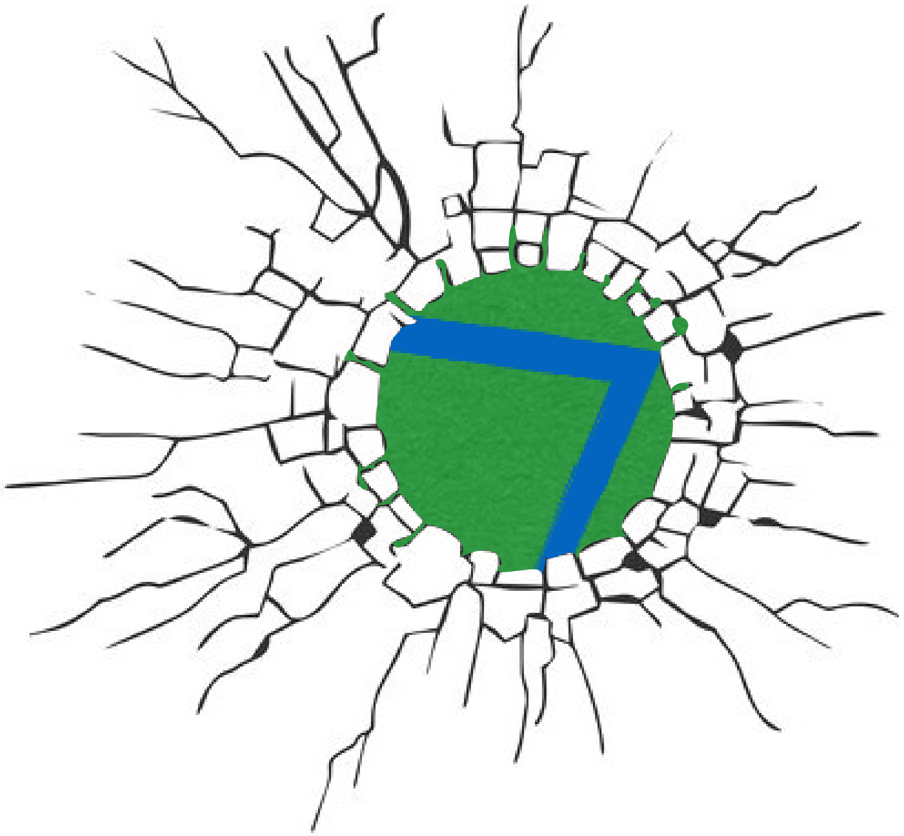
Small visible image patch

Solution: Aperture Problem

- Patches with different gradients to the avoid aperture problem



t



$t + 1$

Topics

- Motion Estimation
  - Optical Flow
  - Lucas-Kanade method
  - Horn-Shunck method

Horn-Schunck

- Most object in the world are rigid or deform elastically moving together coherently

  → **Smoothness Flow:** optical flow fields to be smooth

- **Horn-Schunck Idea:**

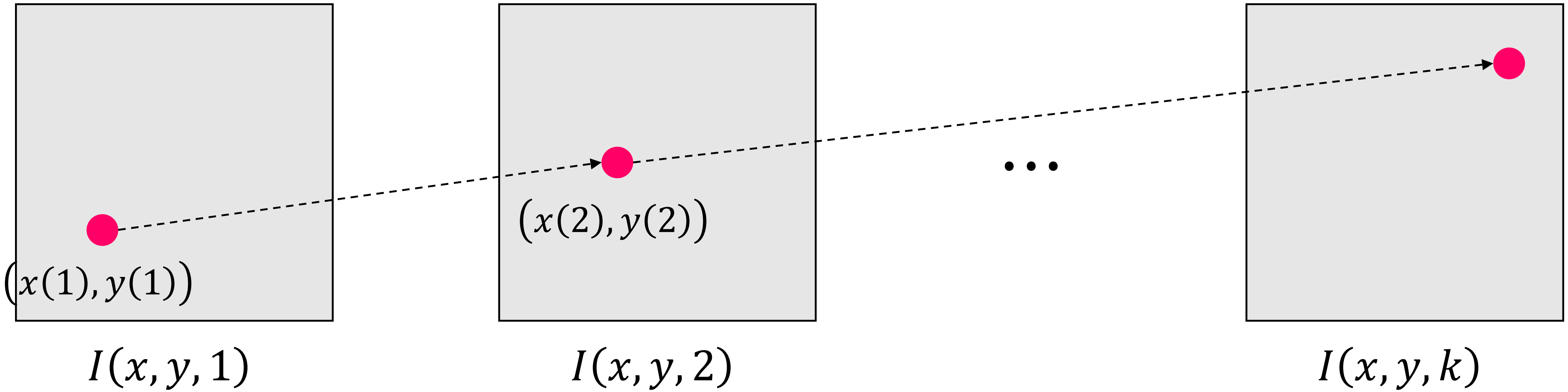
— Enforce both *brightness constancy* and *smooth flow field* to compute optical flow

B.K.P. Horn and B.G. Schunck, Determining Optical Flow, **Artificial Intelligence, 1981**

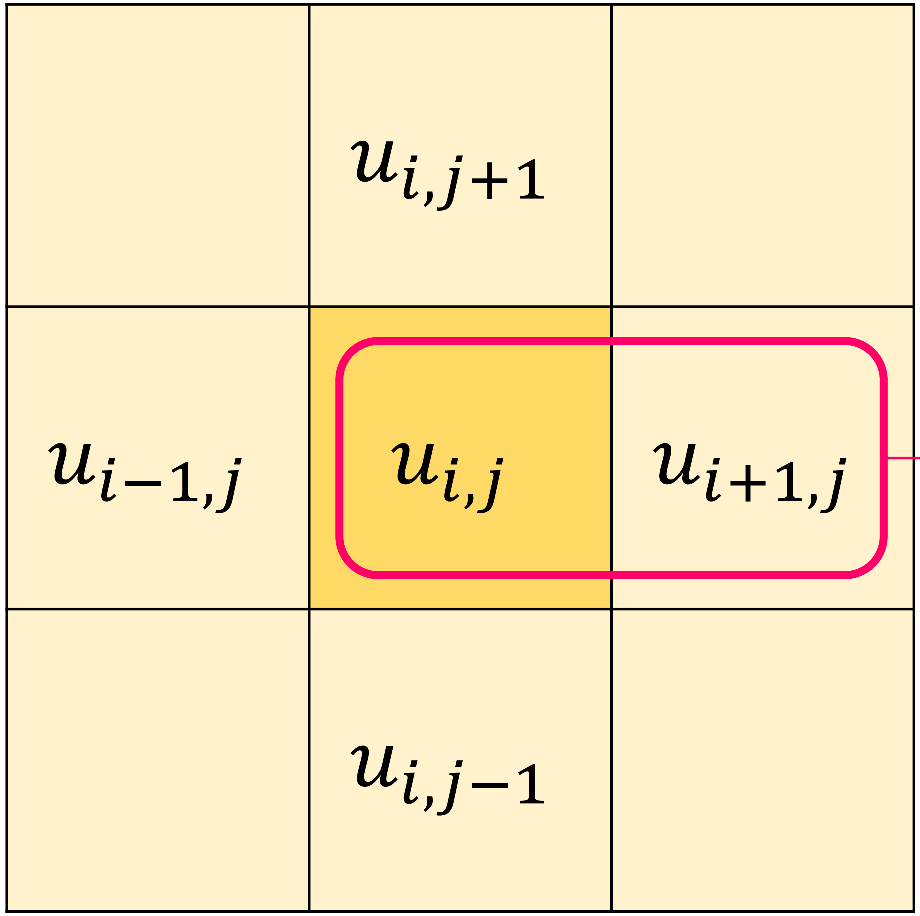
Horn-Schunck: **Brightness Constancy**

- Brightness of the pixel point remains the same: $I_x u + I_y v + I_t = 0$
- For every pixel (i, j),

$$\min_{u,v}\left[I_x(i,j)u_{i,j} + I_y(i,j)v_{i,j} + I_t(i,j)\right]^2$$

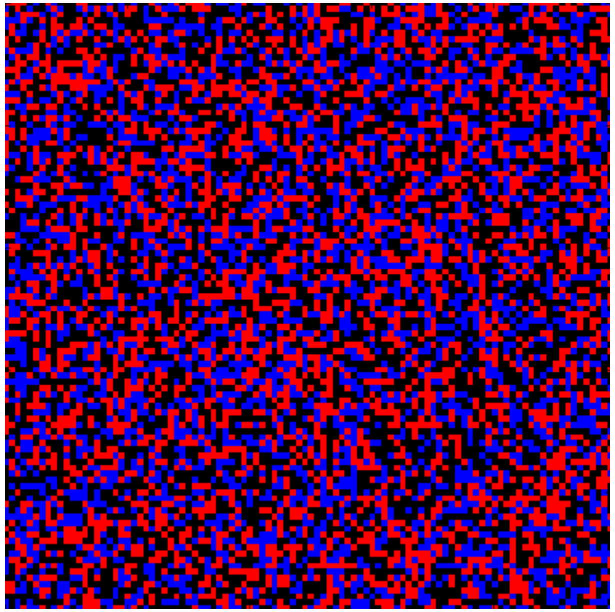


Horn-Schunck: **Smooth Flow Field**



Optical flow in x-direction (u-component)

$$\min_{u}\left[u_{i,j} - u_{i+1,j}\right]^2$$



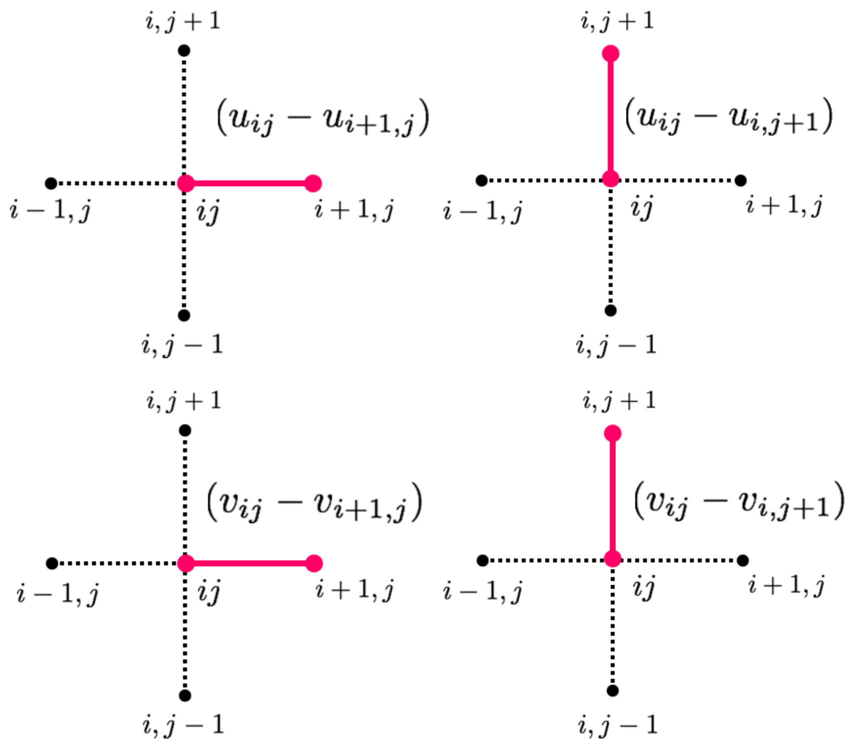
Large differences
between neighboring flows

Small differences
between neighboring flows

Horn-Schunck

- **Horn-Schunck Optical Flow:**

$$\min_{u,v} \sum_{i,j} E_s(i,j) + \lambda E_b(i,j)$$



Smooth flow field

— Brightness Constancy:

$$E_b(i,j) = \left[I_x(i,j)u_{i,j} + I_y(i,j)v_{i,j} + I_t(i,j)\right]^2$$

— Smooth Flow Field:

$$E_s(i,j) = \frac{1}{4}\left[\left(u_{i,j} - u_{i+1,j}\right)^2 + \left(u_{i,j} - u_{i,j+1}\right)^2 + \left(v_{i,j} - v_{i+1,j}\right)^2 + \left(v_{i,j} - v_{i,j+1}\right)^2\right]$$

Horn-Schunck

- **Algorithm:**

① Precompute image gradients, I_x and I_y

② Precompute temporal gradients, I_t

③ Initialize optical flow field $\mathbf{u} = 0$ and $\mathbf{v} = 0$

④ Compute the optical flow field updates for each pixel until the loss function will be converged

$$\hat{u}_{k,l} = \bar{u}_{k,l} - \frac{I_x \bar{u}_{k,l} + I_y \bar{v}_{k,l} + I_t}{\lambda^{-1} + I_x^2 + I_y^2} I_x \qquad \hat{v}_{k,l} = \bar{v}_{k,l} - \frac{I_x \bar{u}_{k,l} + I_y \bar{v}_{k,l} + I_t}{\lambda^{-1} + I_x^2 + I_y^2} I_y$$

Summary: **Optical Flow**

Lucas-Kanade	Horn-Schunk
Brightness Constancy Small Motion	
Constant flow	Smooth Flow Field
Local method (Sparse)	Global method (Dense)