國立高雄科技大學
National Kaohsiung University of Science and Technology

Determining Optical Flow
: *Horn and Schunck Method*

Berthold K. P. Horn and Brain G. Schunck

Artificial Intelligence, 1981

Speaker: Shih-Shinh Huang

April 8, 2021

Outline

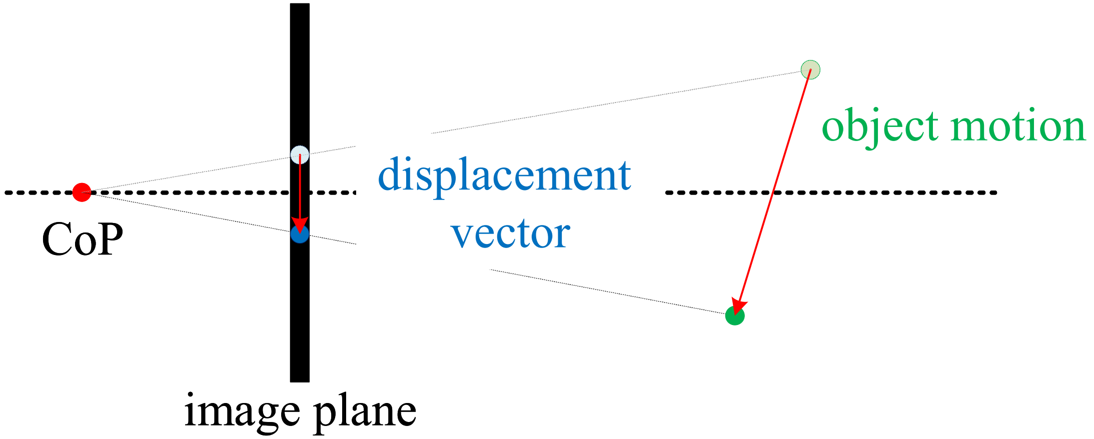
- Introduction
  - About Motion
  - About Optical Flow
- Brightness Constraint
  - Taylor Expansion
  - Formula Derivation
- Horn-Schunck Method
  - Smoothness Constraint
  - Optimization Formulation
  - Iterative Optimization

Introduction

- About Motion
  - Finding the motion of scene objects from time-varying images is important in many applications.
    - object tracking
    - video compression
    - camera stabilization (jitter correction)
    - etc.

Introduction

- About Motion
  - The motion of an object generally results in a displacement vector in the image plane
  - The collection of these displacement vectors is referred to as motion field



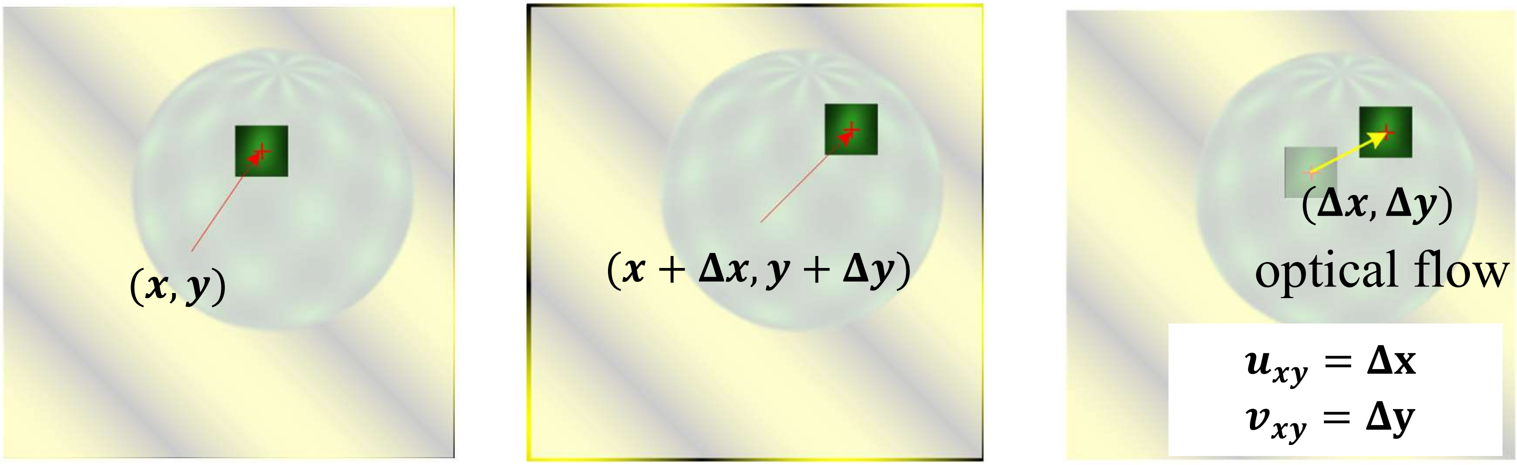
Introduction

- About Motion
  - The methods for computing motion field can be generally divided into two classes.
    - Feature Tracking (sparse): track the extracted visual features (corners, texture patch) over time
    - Optical Flow (dense): use brightness variations in time-varying images.

Horn and Schunck method is in the class of optical flow

Introduction

- About Optical Flow
  - Definition: the apparent motion of brightness patterns in the image.

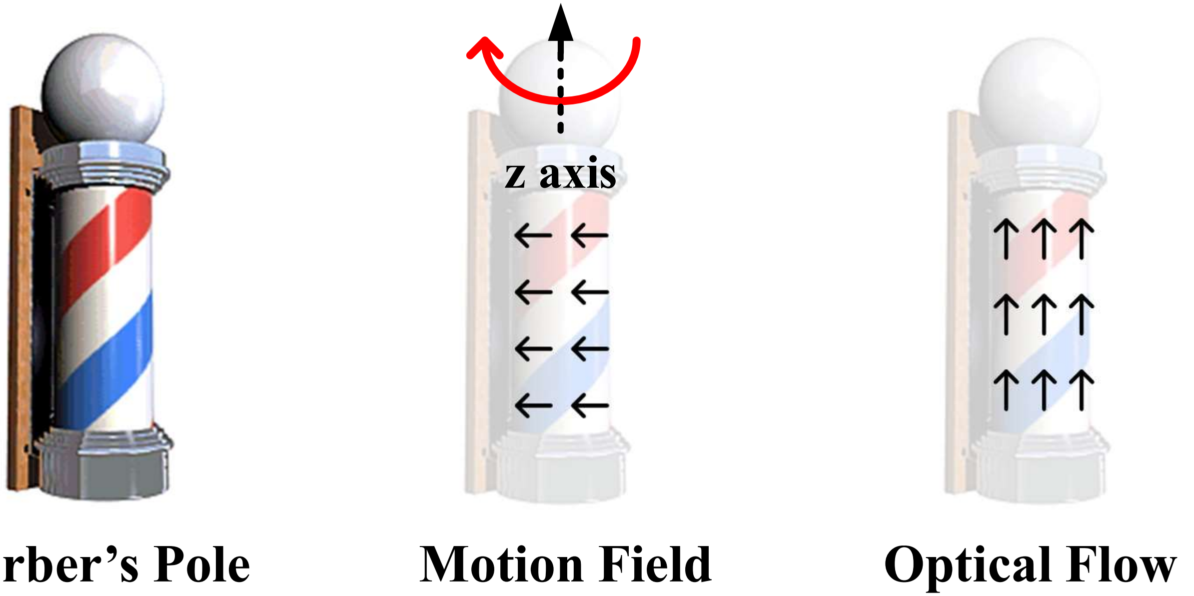


$I(.\,; t)$
image at time t

$I(.\,; t + 1)$
image at time $t + 1$

Introduction

- About Optical Flow
  - Optical flow generally corresponds to the object motion field but not always.

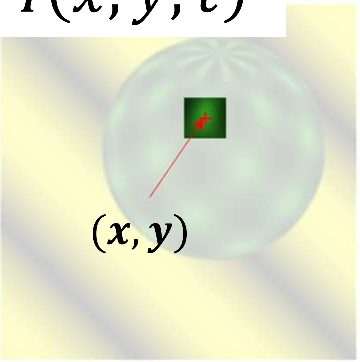


Barber's Pole **Motion Field** **Optical Flow**

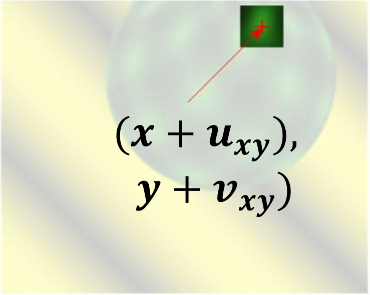
Introduction

- About Optical Flow: Common Assumption
  - Simple World: motion of the object surface is directly related to the one of brightness pattern
  - Brightness Constancy: brightness patterns before/after movement remains the same.

$$I(x, y; t) \quad = \quad I(x + u_{xy}, y + v_{xy}; t + 1)$$



$\boldsymbol{I}(.\,; \boldsymbol{t})$



$\boldsymbol{I}(.\,; \boldsymbol{t} + \mathbf{1})$

Brightness Constraint

- Taylor Expansion
  - Let $f(x)$ be a continuous and n-order differential function.
  - $f(x)$ can be expressed as a series of power terms at any point $x = a$

first differential

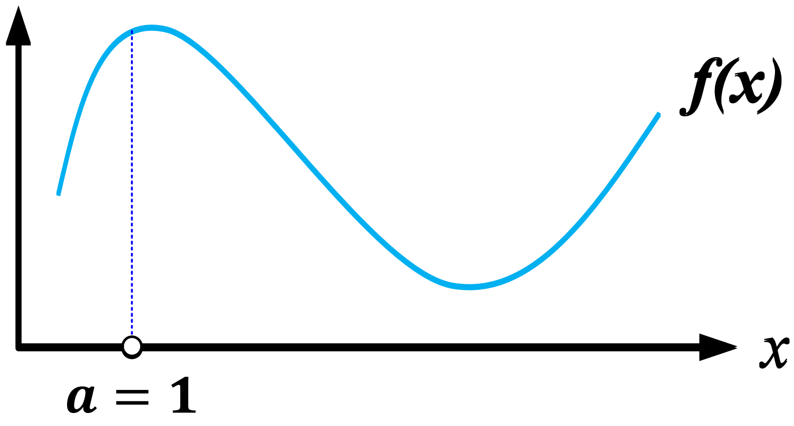
second differential

$$f(x) = f(a) + \frac{f'(a)}{1!}(x-a) + \frac{f''(a)}{2!}(x-a)^2 + \cdots$$

a chosen constant

Brightness Constraint

- Taylor Expansion



$$f(x) = \boxed{f(a)} + \boxed{\frac{f'(a)}{1!}(x-a)}$$

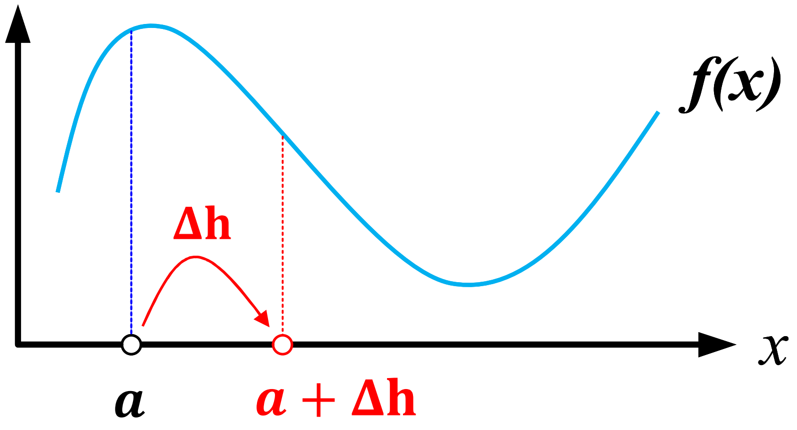
$$+ \frac{f''(a)}{2!}(x-a)^2 + \cdots$$

- Linear Approximation: constant and first differential terms

$$\boldsymbol{f}(\boldsymbol{x}) = f(\rlap{1}\boldsymbol{a}) + f'(\rlap{1}\boldsymbol{a}) \times (x - \rlap{1}\boldsymbol{a})$$

Brightness Constraint

- Taylor Expansion



$$f(x) = f(\boldsymbol{a}) + f'(\boldsymbol{a}) \times (x - \boldsymbol{a})$$

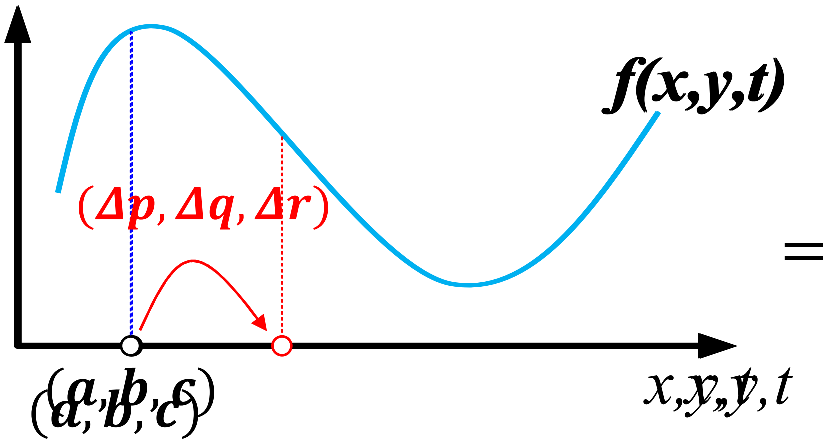
$x \leftarrow a + \Delta\mathrm{h}$

$$f(\boldsymbol{a} + \boldsymbol{\Delta h}) = f(\boldsymbol{a}) + f'(\boldsymbol{a}) \times (\boldsymbol{a} + \boldsymbol{\Delta h} - \boldsymbol{a})$$

$$= f(\boldsymbol{a}) + f'(\boldsymbol{a}) \times (\boldsymbol{\Delta h})$$

Brightness Constraint

- Taylor Expansion: Three Variables
  - extend the variable $x \Rightarrow (x, y, t)$



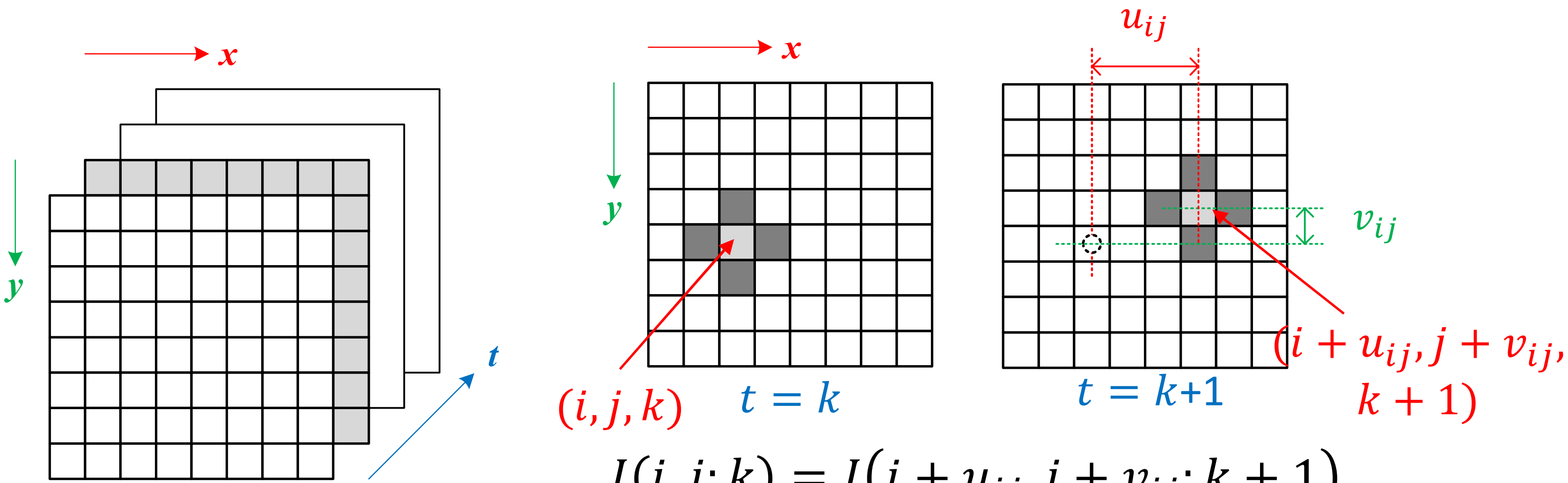
$$f(x,y,t) = f(a,b,c) + \frac{\partial f(a,b,c)}{\partial x} \times (x - \boldsymbol{a}) + \frac{\partial f(a,b,c)}{\partial y} \times (y - \boldsymbol{b}) + \frac{\partial f(a,b,c)}{}$$

$$f((a,b,c) + (\Delta p, \Delta q, \Delta r)) = f(a,b,c) + \frac{\partial f(a,b,c)}{\partial x} \times \Delta p + \frac{\partial f(a,b,c)}{\partial t} \times \Delta q + \frac{\partial f(a,b,c)}{\partial t} \times \Delta r$$

$$\underline{f(x)} = \underline{f(\boldsymbol{a})} + \underline{f'(\boldsymbol{a})} \times \underline{(x - \boldsymbol{a})}$$

Brightness Constraint

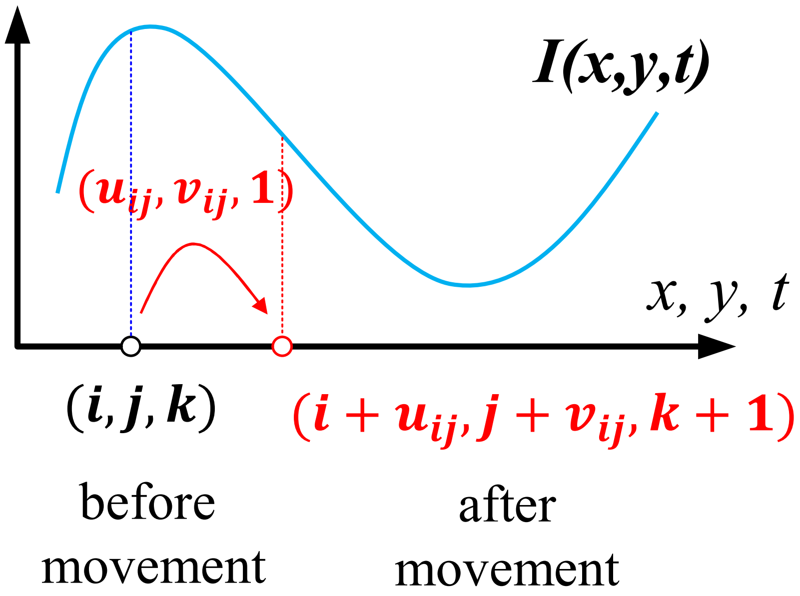
- Formula Derivation
  - consider a sequence of time-varying images $I(x, y; t)$ as a function with three variables
  - let the optical flow of a point (i, j, k) be (u_{ij}, v_{ij}).



$$I(i, j; k) = I\left(i + u_{ij}, j + v_{ij}; k + 1\right)$$

Brightness Constraint

- Formula Derivation
  - approximate point value $I(i + u_{ij}, j + v_{ij},\ k + 1)$ by Taylor expansion at (i, j, k)



$$I\left(i + u_{ij}, j + v_{ij}, k + 1\right)$$

$$= I(\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k}) + \underbrace{\frac{\partial I(\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k})}{\partial x}}_{\equiv \boldsymbol{I_x}} \times \left(\boldsymbol{u_{ij}}\right)$$

$$+ \underbrace{\frac{\partial I(\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k})}{\partial y}}_{\equiv \boldsymbol{I_t}} \times \left(\boldsymbol{v_{ij}}\right)$$

$$+ \underbrace{\frac{\partial I(\boldsymbol{i}, \boldsymbol{j}, \boldsymbol{k})}{\partial t}}_{\equiv \boldsymbol{I_y}} \times (1)$$

Brightness Constraint

- Formula Derivation
  - substitute $I\left(i+u_{ij},j+v_{ij};k+1\right)$ from Taylor expansion in brightness constancy

$$I(i,j;k)=\underline{I\left(i+u_{ij}\,,j+v_{ij};k+1\right)}$$

$$\Rightarrow\ \underline{I(i,j;k)}=\underline{I(i,j;k)}+I_x(i,j;k)\times\left(\boldsymbol{u_{ij}}\right)$$

$$+I_y(i,j;k)\times\left(\boldsymbol{v_{ij}}\right)+I_t(i,j;k)$$

$$\Rightarrow\ 0=\ I_x(i,j;k)\times\left(\boldsymbol{u_{ij}}\right)+I_y(i,j;k)\times\left(\boldsymbol{v_{ij}}\right)+I_t(i,j;k)$$

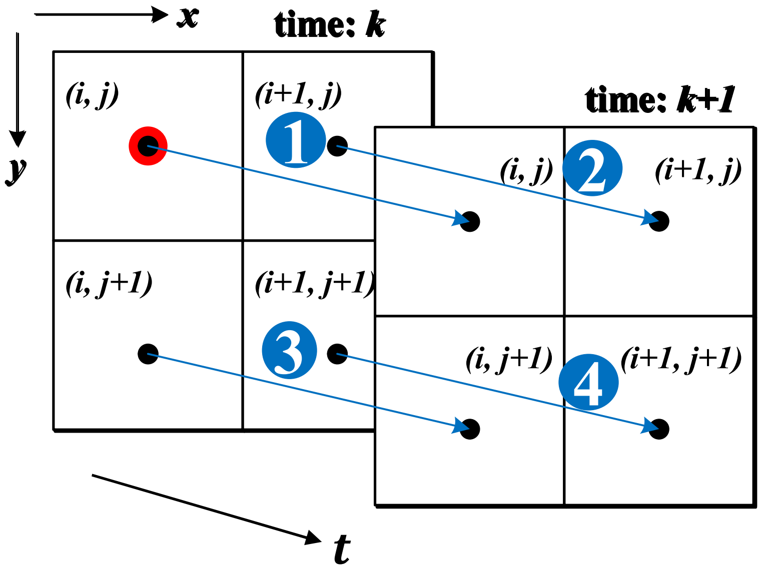
$$I\left(i+\boldsymbol{u_{ij}},j+\boldsymbol{v_{ij}};k+1\right)=I(i,j;k)+I_x(i,j;k)\times\left(\boldsymbol{u_{ij}}\right)$$

$$+I_y(i,j;k)\times\left(\boldsymbol{v_{ij}}\right)+I_t(i,j;k)$$

Brightness Constraint

- Formula Derivation

$$I_x(i,j;k)\times(u_{ij})+I_y(i,j;k)\times(v_{ij})+I_t(i,j;k)=0$$



$$I_x(i,j,k)=\frac{1}{4}\Big(\underbrace{(I(i+1,j,k)-I(i,j,k))}_{\text{①}}+\underbrace{(I(i+1,j+1,k)-I(i,j+1,k))}_{\text{②}}+\underbrace{(I(i+1,j,k+1)-I(i,j,k+1))}_{\text{③}}+\underbrace{(I(i+1,j+1,k+1)-I(i,j+1,k+1))}_{\text{④}}\Big)$$

$$I_y(i,j,k)=\frac{1}{4}\Big(\underbrace{(I(i,j+1,k)-I(i,j,k))}_{\text{①}}+\underbrace{(I(i+1,j+1,k)-I(i+1,j,k))}_{\text{②}}+\underbrace{(I(i,j+1,k+1)-I(i,j,k+1))}_{\text{③}}+\underbrace{(I(i+1,j+1,k+1)-I(i+1,j,k+1))}_{\text{④}}\Big)$$

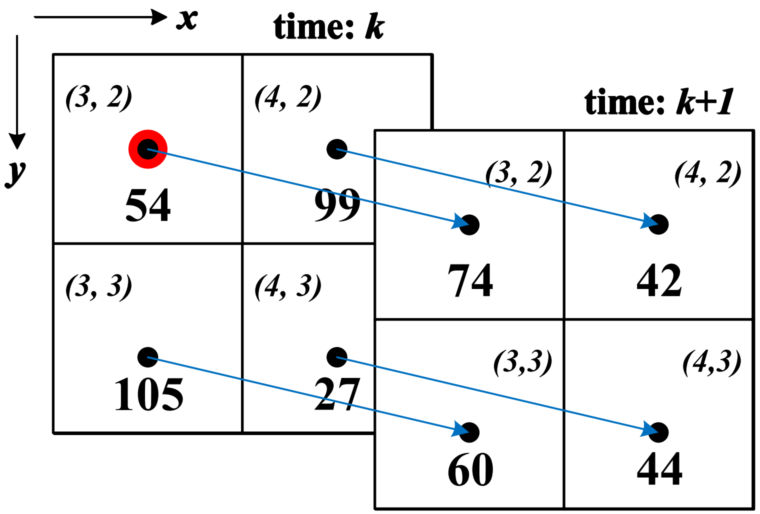
$$I_t(i,j,k)=\frac{1}{4}\Big(\underbrace{(I(i,j,k+1)-I(i,j,k))}_{\text{①}}+\underbrace{(I(i+1,j,k+1)-I(i+1,j,k))}_{\text{②}}+\underbrace{(I(i,j+1,k+1)-I(i,j+1,k))}_{\text{③}}+\underbrace{(I(i+1,j+1,k+1)-I(i+1,j+1,k))}_{\text{④}}\Big)$$

Brightness Constraint

- Formula Derivation

$$I_x(3,2,k) \times (u_{3,2}) + I_y(3,2,k) \times (v_{3,2}) + I_t(3,2,k) = 0$$

$$(-20.25) \times (u_{3,2}) + (-8.25) \times (v_{3,2}) + (-16.25) = 0$$



$$I_x(3,2,k) = \frac{1}{4} \times \begin{pmatrix} (99-54)+(27-105)+ \\ (42-74)+(44-60) \end{pmatrix}$$

$$I_x(3,2,k) = -20.25$$

$$I_y(3,2,k) = \frac{1}{4} \times \begin{pmatrix} (105-54)+(27-99)+ \\ (60-74)+(44-42) \end{pmatrix}$$

$$I_y(3,2,k) = -8.25$$

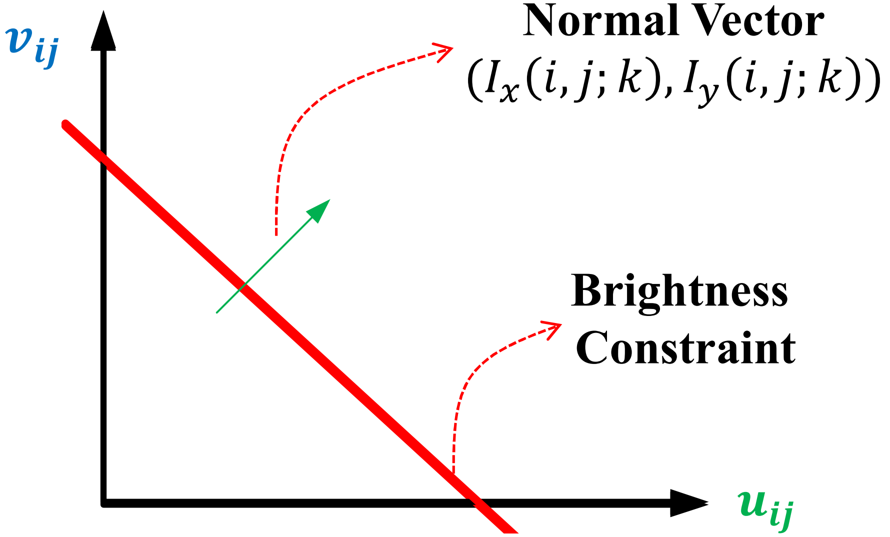
$$I_t(3,2,k) = \frac{1}{4} \times \begin{pmatrix} (74-54)+(42-99)+ \\ (60-105)+(44-27) \end{pmatrix}$$

$$I_t(3,2,k) = -16.25$$

Brightness Constraint

- Formula Derivation

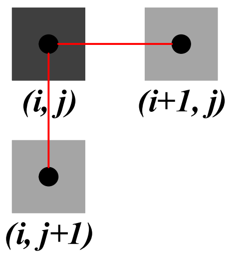
$$I_x(i,j;k) \times (\boldsymbol{u_{ij}}) + I_y(i,j;k) \times (\boldsymbol{v_{ij}}) + I_t(i,j;k) = 0$$



additional constraint is necessary
for determining optical flow of each point

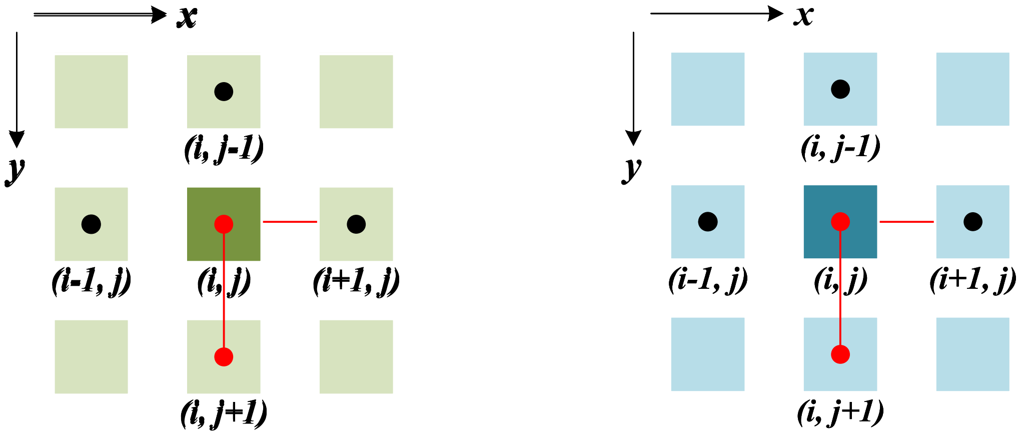
Horn-Schunck Method

- Smoothness Constraint
  - Assumption: objects in the scene commonly undergo rigid motion or deformation
    - Property: neighboring points have similar optical flows (smoothness)
    - Definition: optical flow of the point (i, j) is the same as ones at the points $(i + 1, j)$ and $(i, j + 1)$



Horn-Schunck Method

- Smoothness Constraint



u component $\qquad$ v component

$$\frac{1}{4}\left(\left(u_{ij}-u_{i+1,j}\right)^2 + \left(v_{ij}-v_{i+1,j}\right)^2 + \left(u_{ij}-u_{i,j+1}\right)^2 + \left(v_{ij}-v_{i,j+1}\right)^2\right) = 0$$

Horn-Schunck Method

- Optimization Formulation
  - Given: two images $I(.,t)$ and $I(.,t+1)$
  - Goal: estimate dense optical flow (u_{ij}, v_{ij}) of all points (i,j)
  - Conditions: meet the constraints for all (i,j)

Brightness: $I_x(i,j;k) \times \boldsymbol{u_{ij}} + I_y(i,j;k) \times \boldsymbol{v_{ij}} + I_t(i,j;k) = 0$

Smoothness: $\frac{1}{4}\begin{pmatrix} \left(\boldsymbol{u_{ij}} - \boldsymbol{u_{i+1,j}}\right)^2 + \left(\boldsymbol{v_{ij}} - \boldsymbol{v_{i+1,j}}\right)^2 \\ \left(\boldsymbol{u_{ij}} - \boldsymbol{u_{i,j+1}}\right)^2 + \left(\boldsymbol{v_{ij}} - \boldsymbol{v_{i,j+1}}\right)^2 \end{pmatrix} = 0$

Horn-Schunck Method

- Optimization Formulation
  - Objective Function: sum of brightness error $E_b(.)$ and smoothness error $E_s(.)$.

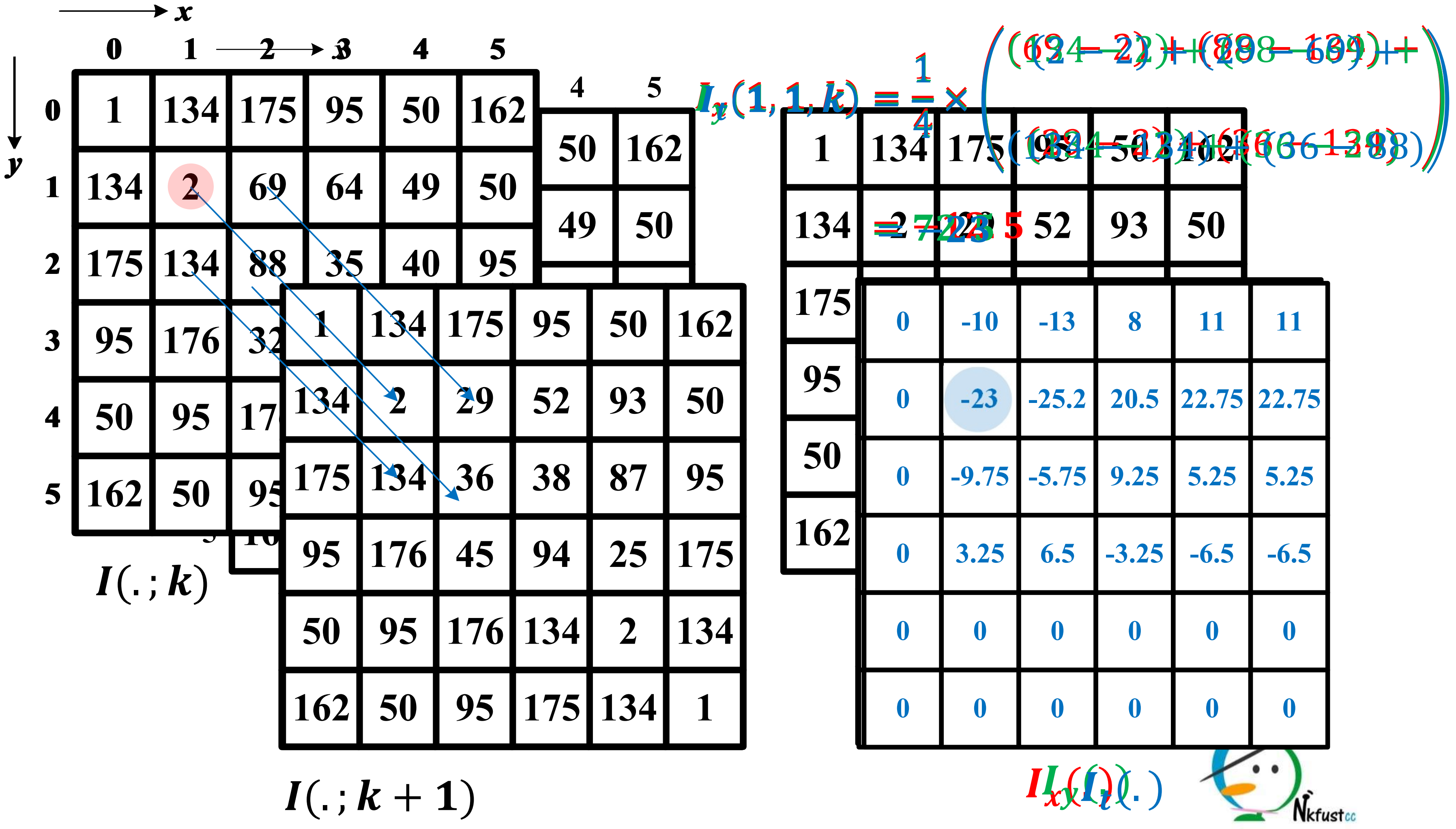
weight parameter

$$E(\boldsymbol{u},\boldsymbol{v}) = \lambda E_b(\boldsymbol{u},\boldsymbol{v}) + E_s(\boldsymbol{u},\boldsymbol{v})$$

$$E_b(\boldsymbol{u},\boldsymbol{v}) = \sum_{(i,j)} \overbrace{\left(I_x(i,j;k)\boldsymbol{u}_{ij} + I_y(i,j;k)\boldsymbol{v}_{ij} + I_t(i,j;k)\right)}^{\text{departure from \textbf{\textit{brightness}} constraint}}{}^{2}$$

$$E_s(\boldsymbol{u},\boldsymbol{v}) = \sum_{(i,j)} \overbrace{\frac{1}{4}\left(\begin{array}{l} \left(\boldsymbol{u}_{ij} - \boldsymbol{u}_{i+1,j}\right)^2 + \left(\boldsymbol{v}_{ij} - \boldsymbol{v}_{i+1,j}\right)^2 \\ \left(\boldsymbol{u}_{ij} - \boldsymbol{u}_{i,j+1}\right)^2 + \left(\boldsymbol{v}_{ij} - \boldsymbol{v}_{i,j+1}\right)^2 \end{array}\right)}^{\text{departure from \textbf{\textit{smoothness}} constraint}}$$

Horn-Schunck Method



y \ x	0	1	2	3	4	5
0	1	134	175	95	50	162
1	134	2	69	64	49	50
2	175	134	88	35	40	95
3	95	176	32			
4	50	95	17			
5	162	50	95			

$I(.;k)$

1	134	175	95	50	162
134	2	29	52	93	50
175	134	36	38	87	95
95	176	45	94	25	175
50	95	176	134	2	134
162	50	95	175	134	1

$I(.;k+1)$

0	-10	-13	8	11	11
0	-23	-25.2	20.5	22.75	22.75
0	-9.75	-5.75	9.25	5.25	5.25
0	3.25	6.5	-3.25	-6.5	-6.5
0	0	0	0	0	0
0	0	0	0	0	0

I_x

0.5	44.0	-35.5	-16	45.5	0
-86.5	-12.5	-8.25	20	5.25	-5.25
20	-104	11.75	-11.2	84.25	-84.2
63	-28.2	3.5	-90.7	134.5	-134
-33.5	63	19	-86.5	-0.5	0.5
-33.5	63	19	-86.5	-0.5	0.5

I_y

0.5	-129	-81.5	-8	-45.5	-45.5
86.5	72.5	-4.25	-14.5	18.25	18.25
-19	9.25	13.73	12.75	27.25	27.25
-63	28.25	92	5.25	-38.5	-38.5
33.5	-63	-20	86.5	-0.5	-0.5
-33.5	63	20	-86.5	0.5	0.5

I_t

0	-10	-13	8	11	11
0	-23	-25.2	20.5	22.75	22.75
0	-9.75	-5.75	9.25	5.25	5.25
0	3.25	6.5	-3.25	-6.5	-6.5
0	0	0	0	0	0
0	0	0	0	0	0

x

y

	0	1	2	3	4	5
0	u_{00}	u_{10}	u_{20}	u_{30}	u_{40}	u_{50}
1	u_{01}	u_{11}	u_{21}	u_{31}	u_{41}	u_{51}
2	u_{02}	u_{12}	u_{22}	u_{32}	u_{42}	u_{52}
3	u_{03}	u_{13}	u_{23}	u_{33}	u_{43}	u_{53}
4	u_{04}	u_{14}	u_{24}	u_{34}	u_{44}	u_{54}
5	u_{05}	u_{15}	u_{25}	u_{35}	u_{45}	u_{55}

0	v_{00}	v_{10}	v_{20}	v_{30}	v_{40}	v_{50}
1	v_{01}	v_{11}	v_{21}	v_{31}	v_{41}	v_{51}
2	v_{02}	v_{12}	v_{22}	v_{32}	v_{42}	v_{52}
3	v_{03}	v_{13}	v_{23}	v_{33}	v_{43}	v_{53}
4	v_{04}	v_{14}	v_{24}	v_{34}	v_{44}	v_{54}
5	v_{05}	v_{15}	v_{25}	v_{35}	v_{45}	v_{55}

$$E(\boldsymbol{u},\boldsymbol{v}) = \lambda E_b(\boldsymbol{u},\boldsymbol{v}) + E_s(\boldsymbol{u},\boldsymbol{v})$$

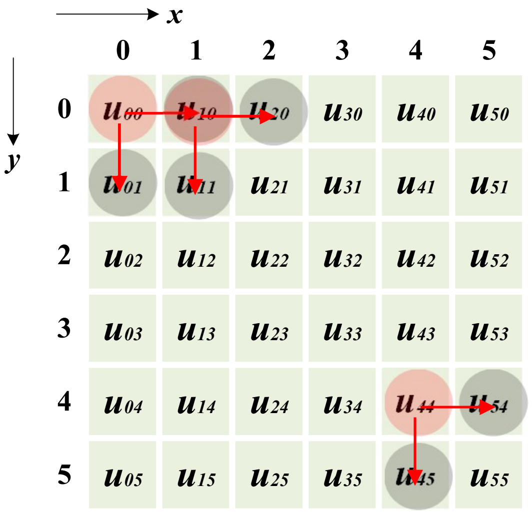
$$E_b(\boldsymbol{u},\boldsymbol{v}) = \sum_{(i,j)} \Big(I_x(i,j;t)\boldsymbol{u_{ij}} + I_y(i,j;t)\boldsymbol{v_{ij}} + I_t(i,j;t)\Big)^2$$

$$= (0.5u_{00} + 0.5v_{00} + 0)^2$$

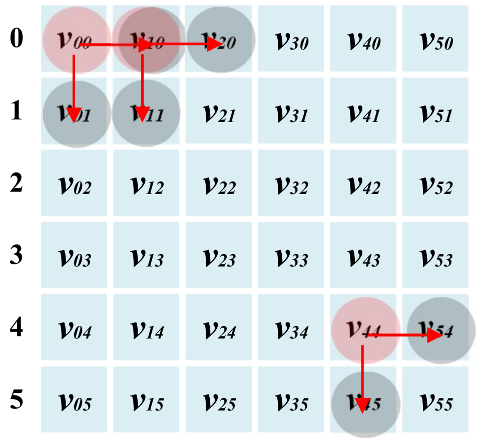
$$+ (44u_{10} - 129v_{10} - 10)^2$$

$$+ \cdots$$

$$+ (0.5u_{55} + 0.5v_{55} + 0)^2$$



	0	1	2	3	4	5
0	u_{00}	u_{10}	u_{20}	u_{30}	u_{40}	u_{50}
1	u_{01}	u_{11}	u_{21}	u_{31}	u_{41}	u_{51}
2	u_{02}	u_{12}	u_{22}	u_{32}	u_{42}	u_{52}
3	u_{03}	u_{13}	u_{23}	u_{33}	u_{43}	u_{53}
4	u_{04}	u_{14}	u_{24}	u_{34}	u_{44}	u_{54}
5	u_{05}	u_{15}	u_{25}	u_{35}	u_{45}	u_{55}



0	v_{00}	v_{10}	v_{20}	v_{30}	v_{40}	v_{50}
1	v_{01}	v_{11}	v_{21}	v_{31}	v_{41}	v_{51}
2	v_{02}	v_{12}	v_{22}	v_{32}	v_{42}	v_{52}
3	v_{03}	v_{13}	v_{23}	v_{33}	v_{43}	v_{53}
4	v_{04}	v_{14}	v_{24}	v_{34}	v_{44}	v_{54}
5	v_{05}	v_{15}	v_{25}	v_{35}	v_{45}	v_{55}

$$E(u,v) = \lambda E_b(u,v) + E_s(u,v)$$

$$E_s(u,v) = \sum_{(i,j)} \frac{1}{4} \begin{pmatrix} (u_{ij} - u_{i+1,j})^2 + (v_{ij} - v_{i+1,j})^2 \\ (u_{ij} - u_{i,j+1})^2 + (v_{ij} - v_{i,j+1})^2 \end{pmatrix}$$

$$= \frac{1}{4} \begin{pmatrix} (u_{00} - u_{10})^2 + (v_{00} - v_{10})^2 \\ (u_{00} - u_{01})^2 + (v_{00} - v_{01})^2 \end{pmatrix}$$

$$+ \frac{1}{4} \begin{pmatrix} (u_{10} - u_{20})^2 + (v_{10} - v_{20})^2 \\ (u_{10} - u_{11})^2 + (v_{10} - v_{11})^2 \end{pmatrix}$$

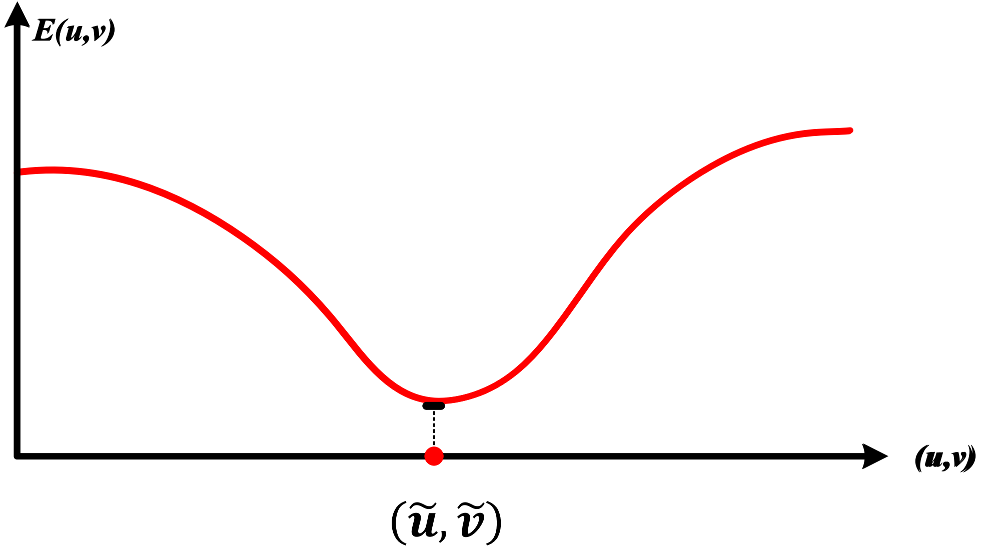
$$+ \cdots$$

$$+ \frac{1}{4} \begin{pmatrix} (u_{44} - u_{54})^2 + (v_{44} - v_{54})^2 \\ (u_{44} - u_{45})^2 + (v_{44} - v_{45})^2 \end{pmatrix}$$

Horn-Schunck Method

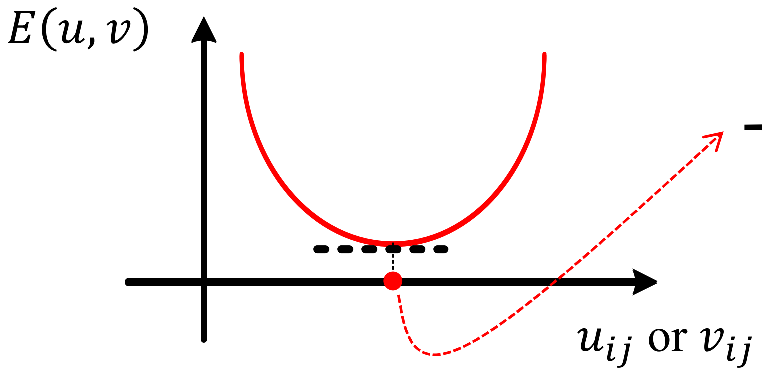
- Optimization Formulation
  - Goal: find the optimal $(\tilde{u}, \tilde{v})$ in the (u, v) parameter space that minimizes $E(u, v)$.

$$(\tilde{u}, \tilde{v}) = \arg\min_{(u,v)} E(u, v)$$



Horn-Schunck Method

- Iterative Optimization
  - $E(u,v)$ is a quadratic function **w.r.t.** any u_{ij} or v_{ij}
  - The extrema of $E(u,v)$ is at the derivatives equal to zero **w.r.t**. any u_{ij} or v_{ij}



$$\begin{cases} \dfrac{\partial E(u,v)}{\partial u_{ij}} = 0 \quad \forall (i,j) \\ \dfrac{\partial E(u,v)}{\partial v_{ij}} = 0 \quad \forall (i,j) \end{cases}$$

Horn-Schunck Method

- Iterative Optimization

$$E(u,v) = \lambda \times \sum_{(i,j)} \left(I_x(i,j;t)\,u_{ij} + I_y(i,j;t)\,v_{ij} + I_t(i,j;t) \right)^2$$

$$+ \sum_{(i,j)} \frac{1}{4} \left(\begin{array}{c} (u_{ij} - u_{i+1,j})^2 + (v_{ij} - v_{i+1,j})^2 \\ (u_{ij} - u_{i,j+1})^2 + (v_{ij} - v_{i,j+1})^2 \end{array} \right)$$

$$\frac{\partial E(u,v)}{\partial u_{kl}} = \lambda \times \sum_{(i,j)} \frac{\partial}{\partial u_{rs}} \left(I_x(k,l;t)\,u_{rs} + I_y(k,l;t)\,v_{rs} + I_t(k,l;t) \right)^2$$

$$+ \sum_{(i,j)} \frac{\partial}{\partial u_{rs}} \frac{1}{4} \left(\begin{array}{c} (u_{ij} - u_{i+1,j})^2 + (v_{ij} - v_{i+1,j})^2 \\ (u_{ij} - u_{i,j+1})^2 + (v_{ij} - v_{i,j+1})^2 \end{array} \right)$$

$$= \frac{\partial}{\partial u_{rs}} \left(I_x(0,0;t)\,u_{00} + I_y(0,0;t)\,v_{00} + I_t(0,0;t) \right)^2 + \cdots$$

= 0 = 0

$$+ \frac{\partial}{\partial u_{rs}} \left(I_x(r,s;t)\,u_{rs} + I_y(r,s;t)\,v_{rs} + I_t(r,s;t) \right)^2 + \cdots$$

= 0

$$\frac{\partial E(u,v)}{\partial u_{rs}} = \lambda \times \frac{\partial}{\partial u_{rs}} \left(I_x(k,l;t)\, u_{kl} + I_y(k,l;t)\, v_{kl} + I_t(k,l;t) \right)^2$$

$$+ \frac{\partial}{\partial u_{rs}} \sum_{(i,j)} \left(\frac{1}{4} \left(\begin{matrix} (u_{ij}-u_{i+1,j})^2 + (v_{ij}-v_{i+1,j})^2 \\ (u_{ij}-u_{i,j+1})^2 + (v_{ij}-v_{i,j+1})^2 \end{matrix} \right) \right)$$

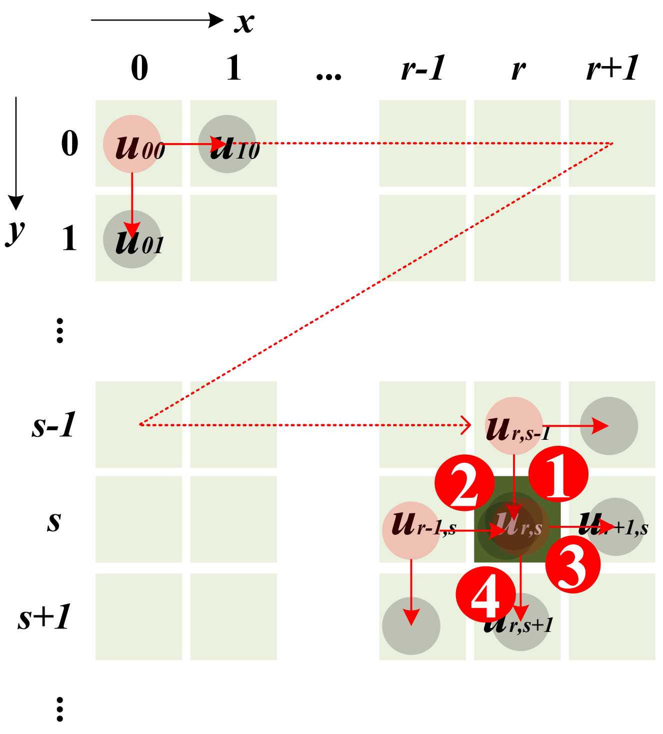
$$= \cancelto{0}{\frac{\partial}{\partial u_{rs}} \frac{1}{4} \left(\begin{matrix} (u_{00}-u_{10})^2 + (v_{00}-v_{10})^2 \\ (u_{00}-u_{01})^2 + (v_{00}-v_{01})^2 \end{matrix} \right)} \quad \cancelto{0}{+\cdots}$$

$$+ \frac{\partial}{\partial u_{rs}} \frac{1}{4} \left(\begin{matrix} \cancelto{0}{(u_{r,s-1}-u_{r+1,s-1})^2} + \cancelto{0}{(v_{r,s-1}-v_{r+1,s-1})^2} \\ \textcircled{1}\ (u_{r,s-1}-u_{r,s})^2 + \cancel{(v_{r,s-1}-v_{r,s})^2} \end{matrix} \right)$$

$$+\cdots$$

$$+ \frac{\partial}{\partial u_{rs}} \frac{1}{4} \left(\begin{matrix} \textcircled{2}\ (u_{r-1,s}-u_{r,s})^2 + \cancelto{0}{(v_{r-1,s}-v_{r,s})^2} \\ \cancelto{0}{(u_{r-s,l}-u_{r-1,s+1})^2} + \cancel{(v_{r-1,s}-v_{r-1,s+1})^2} \end{matrix} \right)$$

$$+ \frac{\partial}{\partial u_{rs}} \frac{1}{4} \left(\begin{matrix} (u_{r,s}-u_{r+1,s})^2\ \textcircled{3} + \cancelto{0}{(v_{r,s}-v_{r+1,s})^2} \\ (u_{r,s}-u_{r,s+1})^2\ \textcircled{4} + \cancel{(v_{r,s}-v_{r,s+1})^2} \end{matrix} \right) + \cdots$$



$$\frac{E(\boldsymbol{u},\boldsymbol{v})}{\partial \boldsymbol{u}_{rs}} = \lambda \times \frac{\partial}{\partial \boldsymbol{u}_{rs}}\left(\underbrace{I_x(r,s;t)}_{\equiv I_x}\boldsymbol{u}_{rs} + \underbrace{I_y(r,s;t)}_{\equiv I_y}\boldsymbol{v}_{rs} + \underbrace{I_t(r,s;t)}_{\equiv I_t}\right)^2$$

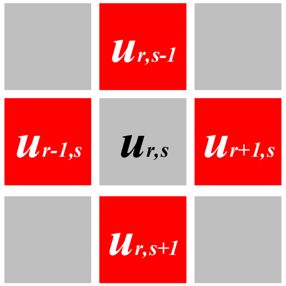
$$+\frac{\partial}{\partial \boldsymbol{u}_{rs}}\frac{1}{4}\begin{pmatrix}(\boldsymbol{u}_{r,s-1}-\boldsymbol{u}_{rs})^2 + (\boldsymbol{u}_{r-1,s}-\boldsymbol{u}_{rs})^2\\ (\boldsymbol{u}_{rs}-\boldsymbol{u}_{r+1,s})^2 + (\boldsymbol{u}_{rs}-\boldsymbol{u}_{r,s+1})^2\end{pmatrix}$$

$$= \lambda \times 2 \times (I_x u_{rs} + I_y v_{rs} + I_t) \times I_x$$

$$+\frac{1}{4}\begin{pmatrix}2\times(\boldsymbol{u}_{r,s-1}-\underline{\boldsymbol{u}_{rs}})\times(-\mathbf{1}) + 2\times(\boldsymbol{u}_{r-1,s}-\underline{\boldsymbol{u}_{rs}})\times(-\mathbf{1})\\ 2\times(\underline{\boldsymbol{u}_{rs}}-\boldsymbol{u}_{r+1,s})\times(\mathbf{1}) + 2\times(\underline{\boldsymbol{u}_{rs}}-\boldsymbol{u}_{r+1,s})\times(\mathbf{1})\end{pmatrix}$$

$$= \lambda \times 2 \times (I_x^2 u_{rs} + I_x I_y v_{rs} + I_x I_t)$$

$$+2\times\left(\boldsymbol{u}_{rs} - \underbrace{\frac{1}{4}\times(\boldsymbol{u}_{r,s-1}+\boldsymbol{u}_{r-1,s}+\boldsymbol{u}_{r+1,s}+\boldsymbol{u}_{r,s+1})}_{\equiv \overline{\boldsymbol{u}_{rs}}}\right)$$



	$u_{r,s-1}$	
$u_{r-1,s}$	$u_{r,s}$	$u_{r+1,s}$
	$u_{r,s+1}$	

General Power Law: $\frac{\partial}{\partial x}[f(x)]^n \equiv n \times [f(x)]^{n-1} \times \frac{\partial f(x)}{\partial x}$

Horn-Schunck Method

- Iterative Optimization

$$\frac{E(\boldsymbol{u},\boldsymbol{v})}{\partial \boldsymbol{u_{rs}}} = \boldsymbol{\lambda} \times \cancel{2} \times \left(I_x^2 \boldsymbol{u_{rs}} + I_x I_y \boldsymbol{v_{rs}} + I_x I_t\right) + \cancel{2} \times \left(\boldsymbol{u_{r,s}} - \overline{\boldsymbol{u_{r,s}}}\right) \mathbf{= 0}$$

$$\frac{E(\boldsymbol{u},\boldsymbol{v})}{\partial \boldsymbol{v_{rs}}} = \boldsymbol{\lambda} \times \cancel{2} \times \left(I_x I_y \boldsymbol{u_{rs}} + I_y^2 \boldsymbol{v_{rs}} + I_y I_t\right) + \cancel{2} \times \left(\boldsymbol{v_{r,s}} - \overline{\boldsymbol{v_{r,s}}}\right) \mathbf{= 0}$$

$$\begin{aligned} (\mathbf{1} + \boldsymbol{\lambda} I_x^2)\boldsymbol{u_{rs}} + (\boldsymbol{\lambda} I_x I_y)\boldsymbol{v_{rs}} &= \overline{\boldsymbol{u_{r,s}}} - \lambda I_x I_t \\ (\boldsymbol{\lambda} I_x I_y)\boldsymbol{u_{rs}} + (\mathbf{1} + \boldsymbol{\lambda} I_y^2)\boldsymbol{v_{rs}} &= \overline{\boldsymbol{v_{r,s}}} - \lambda I_y I_t \end{aligned}$$

system of two linear equations in two unknows u_{rs} and v_{rs}

Horn-Schunck Method

- Iterative Optimization

$$\begin{cases} \overset{=\,a_1}{\left(1+\lambda I_x^2\right)} u_{rs} + \overset{=\,b_1}{\left(\lambda I_x I_y\right)} v_{rs} = \overset{=\,c_1}{\overline{u_{r,s}} - \lambda I_x I_t} \\ \overset{=\,a_2}{\left(\lambda I_x I_y\right)} u_{rs} + \overset{=\,b_2}{\left(1+\lambda I_y^2\right)} v_{rs} = \overset{=\,c_2}{\overline{v_{r,s}} - \lambda I_y I_t} \end{cases}$$

….

long calculation

$$u_{rs} = \overline{u_{rs}} - \frac{\overline{u_{r,s}} \times I_x + \overline{v_{r,s}} \times I_y + I_t}{\lambda^{-1} + I_x^2 + I_y^2} \times I_x$$

$$v_{rs} = \overline{v_{rs}} - \frac{\overline{u_{r,s}} \times I_x + \overline{v_{r,s}} \times I_y + I_t}{\lambda^{-1} + I_x^2 + I_y^2} \times I_y$$

$$\begin{cases} a_1 x + b_1 y = c_1 \\ a_2 x + b_2 y = c_2 \end{cases} \qquad x = \frac{b_2 c_1 - b_1 c_2}{a_1 b_2 - a_2 b_1} \qquad y = \frac{a_1 c_2 - a_2 c_1}{a_1 b_2 - a_2 b_1}$$

Horn-Schunck Method

- Iterative Optimization
  - Estimating optical flow by solving all equations simultaneously is very costly
    - Each point has a pair of equations with two unknowns.
    - This results in $w \times h \times 2$ equations in total
  - An iterative algorithm is used for optical flow estimation in Horn-Schunck Method.

Horn-Schunck Method

- Iterative Optimization

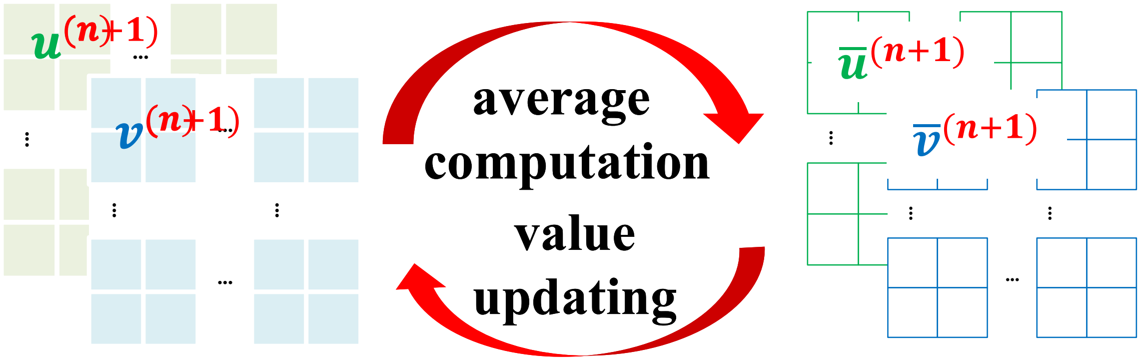
$$u_{rs}^{(n+1)} = \overline{u}_{rs}^{(n)} - \frac{\overline{u}_{rs}^{(n)} \times I_x + \overline{v}_{rs}^{(n)} \times I_y + I_t}{\lambda^{-1} + I_x^2 + I_y^2} \times I_x$$

$$v_{rs}^{(n+1)} = \overline{v}_{rs}^{(n)} - \frac{\overline{u}_{rs}^{(n)} \times I_x + \overline{v}_{rs}^{(n)} \times I_y + I_t}{\lambda^{-1} + I_x^2 + I_y^2} \times I_y$$

$$u_{rs} = \overline{u}_{rs} - \frac{\overline{u}_{rs} \times I_x + \overline{v}_{rs} \times I_y + I_t}{\lambda^{-1} + \lambda I_x^2 + I_y^2} \times I_x$$

$$v_{rs} = \overline{v}_{rs} - \frac{\overline{u}_{rs} \times I_x + \overline{v}_{rs} \times I_y + I_t}{\lambda^{-1} + I_x^2 + I_y^2} \times I_y$$

new value **old average**

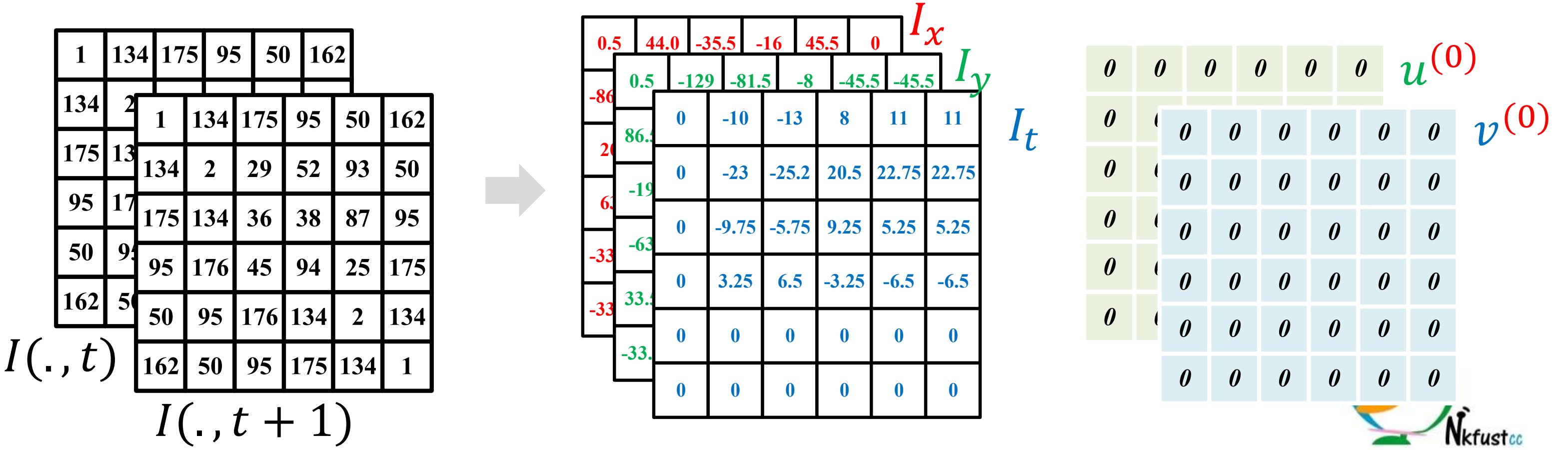


average computation

value updating

Horn-Schunck Method

- Iterative Optimization: Algorithm
  - Input: (1) two images: $I(.,t)$ and $I(.,t+1)$; (2) paramters: λ and N (number of iterations)
  - Initialization
    - pre-compute three derivatives: I_x, I_y and I_t
    - initialize (u_{ij}, v_{ij}) of all points (i,j) to $(0.0, 0.0)$



$I(.,t+1)$

1	134	175	95	50	162
134	2	29	52	93	50
175	134	36	38	87	95
95	176	45	94	25	175
50	95	176	134	2	134
162	50	95	175	134	1

I_t

0	-10	-13	8	11	11
0	-23	-25.2	20.5	22.75	22.75
0	-9.75	-5.75	9.25	5.25	5.25
0	3.25	6.5	-3.25	-6.5	-6.5
0	0	0	0	0	0
0	0	0	0	0	0

$v^{(0)}$

0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0
0	0	0	0	0	0

Horn-Schunck Method

- Iterative Optimization: Algorithm
  - Iteration (`for n=0 to` $N-1$)
    - Average Computation: compute $(\overline{u_{ij}}^{(n)}, \overline{v_{ij}}^{(n)})\ \forall(i,j)$
    - Value Updating: update $(u_{ij}^{(n+1)}, v_{ij}^{(n+1)})\ \forall(i,j)$

$$\overline{u_{ij}}^{(n)} = \frac{1}{4}\left(u_{i,j-1}^{(n)} + u_{i-1,j}^{(n)} + u_{i+1,j}^{(n)} + u_{i,j+1}^{(n)}\right)$$

$$\overline{v_{ij}}^{(n)} = \frac{1}{4}\left(v_{i,j-1}^{(n)} + v_{i-1,j}^{(n)} + v_{i+1,j}^{(n)} + v_{i,j+1}^{(n)}\right)$$

$$u_{ij}^{(n+1)} = \overline{u_{ij}}^{(n)} - \frac{\overline{u_{ij}}^{(n)} \times I_x + \overline{v_{ij}}^{(n)} \times I_y + I_t}{\lambda + I_x^2 + I_y^2} \times I_x$$

$$v_{ij}^{(n+1)} = \overline{v_{ij}}^{(n)} - \frac{\overline{u_{ij}}^{(n)} \times I_x + \overline{v_{ij}}^{(n)} \times I_y + I_t}{\lambda + I_x^2 + I_y^2} \times I_y$$


thank you