Optical Flow

Asif Rajput*

*(Thanks to Shree Nayyar and Mubarak Shah)

(source)

Optical Flow

Method to estimate apparent motion of scene points from a sequence of images.

Topics:

(1) Motion Field and Optical Flow

(2) Optical Flow Constraint Equation

(3) Lucas-Kanade Method

(4) Coarse-to-Fine Flow Estimation

Motion Field

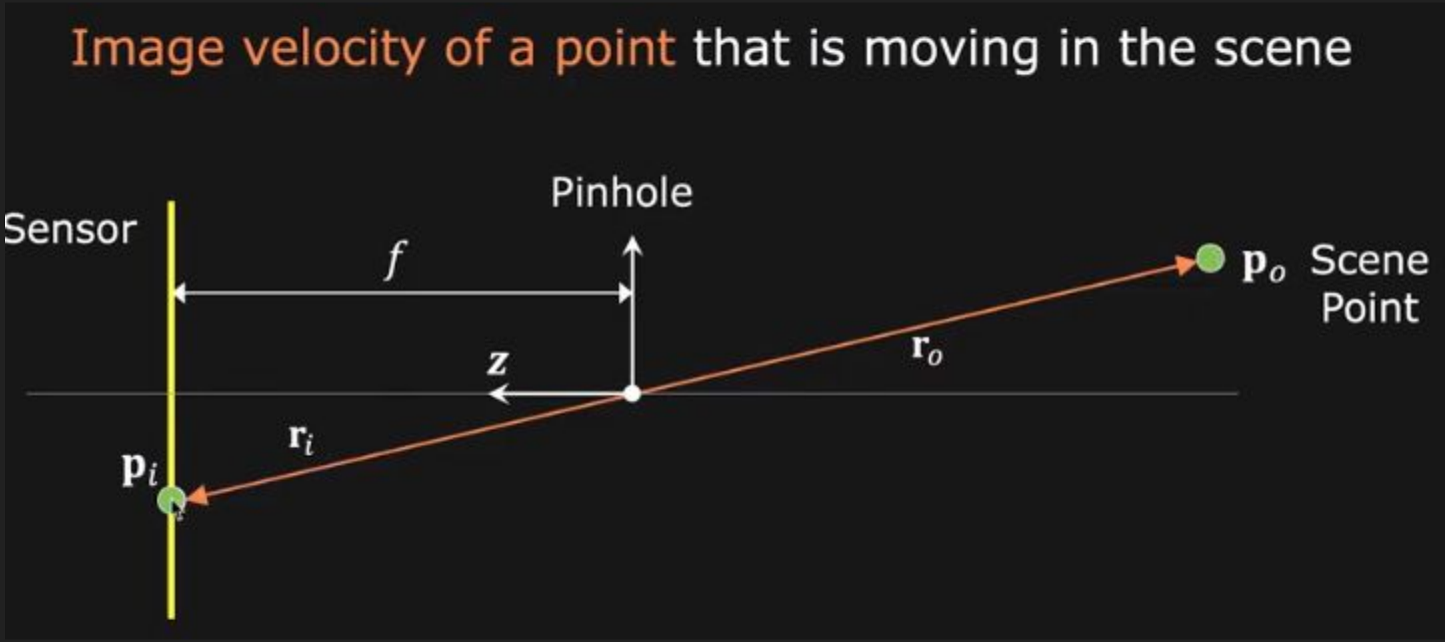


Image Point Velocity: $\mathbf{v}_i = \frac{d\mathbf{r}_i}{dt}$
(Motion Field)

Scene Point Velocity: $\mathbf{v}_o = \frac{d\mathbf{r}_o}{dt}$

Image velocity of a point that is moving in the scene

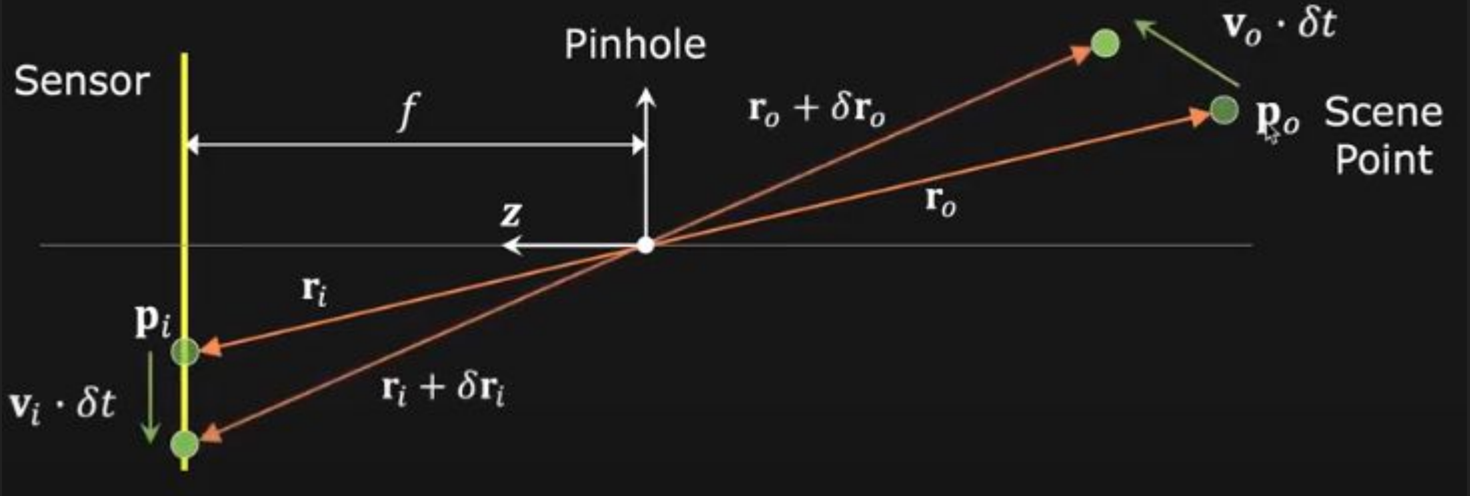
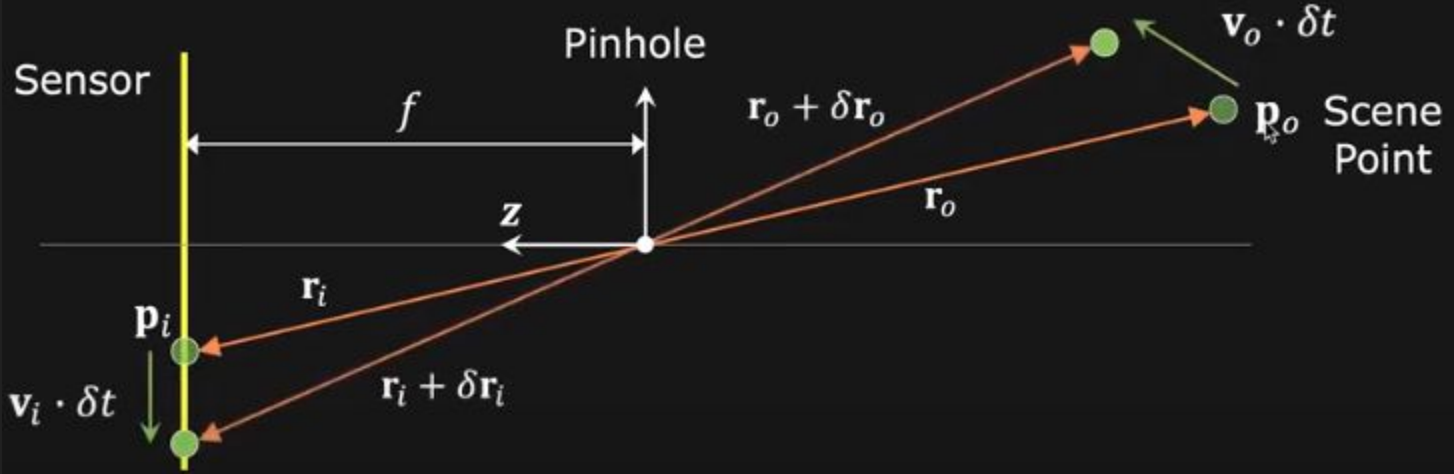


Image Point Velocity: $\mathbf{v}_i = \frac{d\mathbf{r}_i}{dt}$
(Motion Field)

Scene Point Velocity: $\mathbf{v}_o = \frac{d\mathbf{r}_o}{dt}$

Image velocity of a point that is moving in the scene



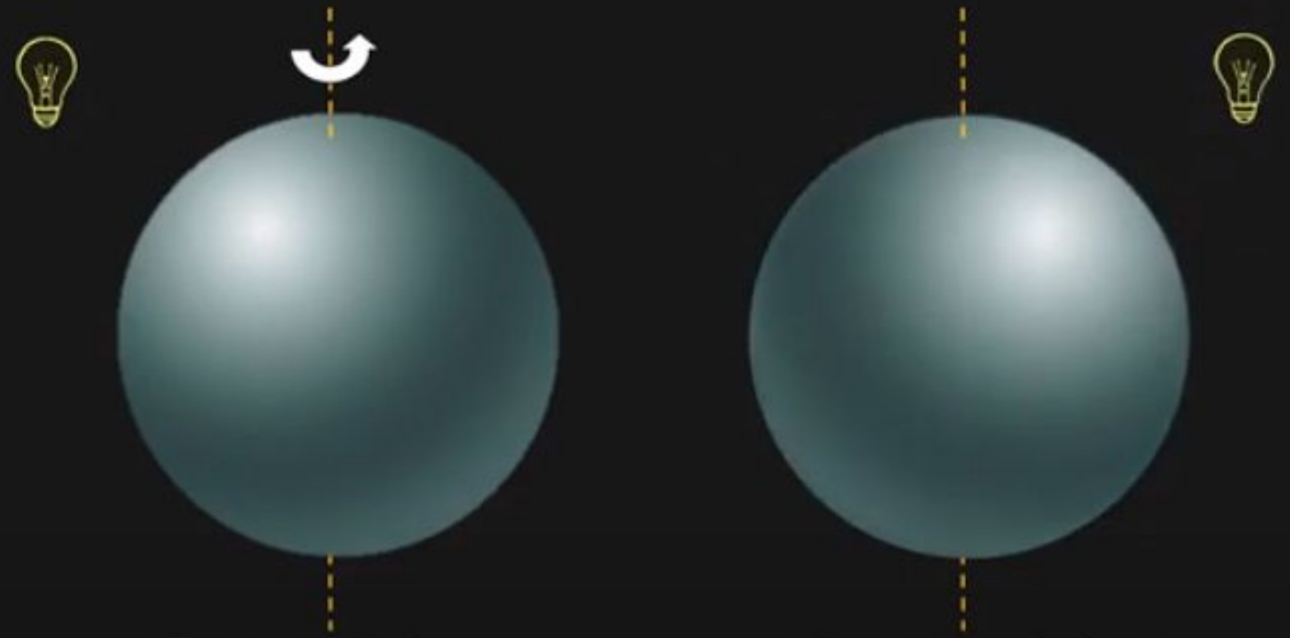
Perspective projection: $\frac{\mathbf{r}_i}{f} = \frac{\mathbf{r}_o}{\mathbf{r}_o \cdot \mathbf{z}}$


1.4


14

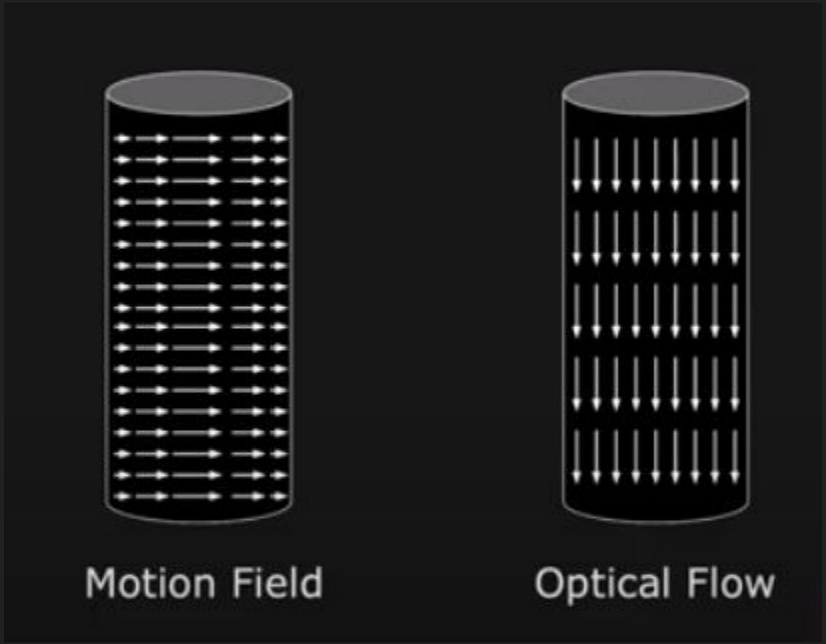
Optical Flow (Ideally, Optical Flow = Motion Field)

When is Optical Flow $\neq$ Motion Field?



Spinning Sphere
Stationary Light Source

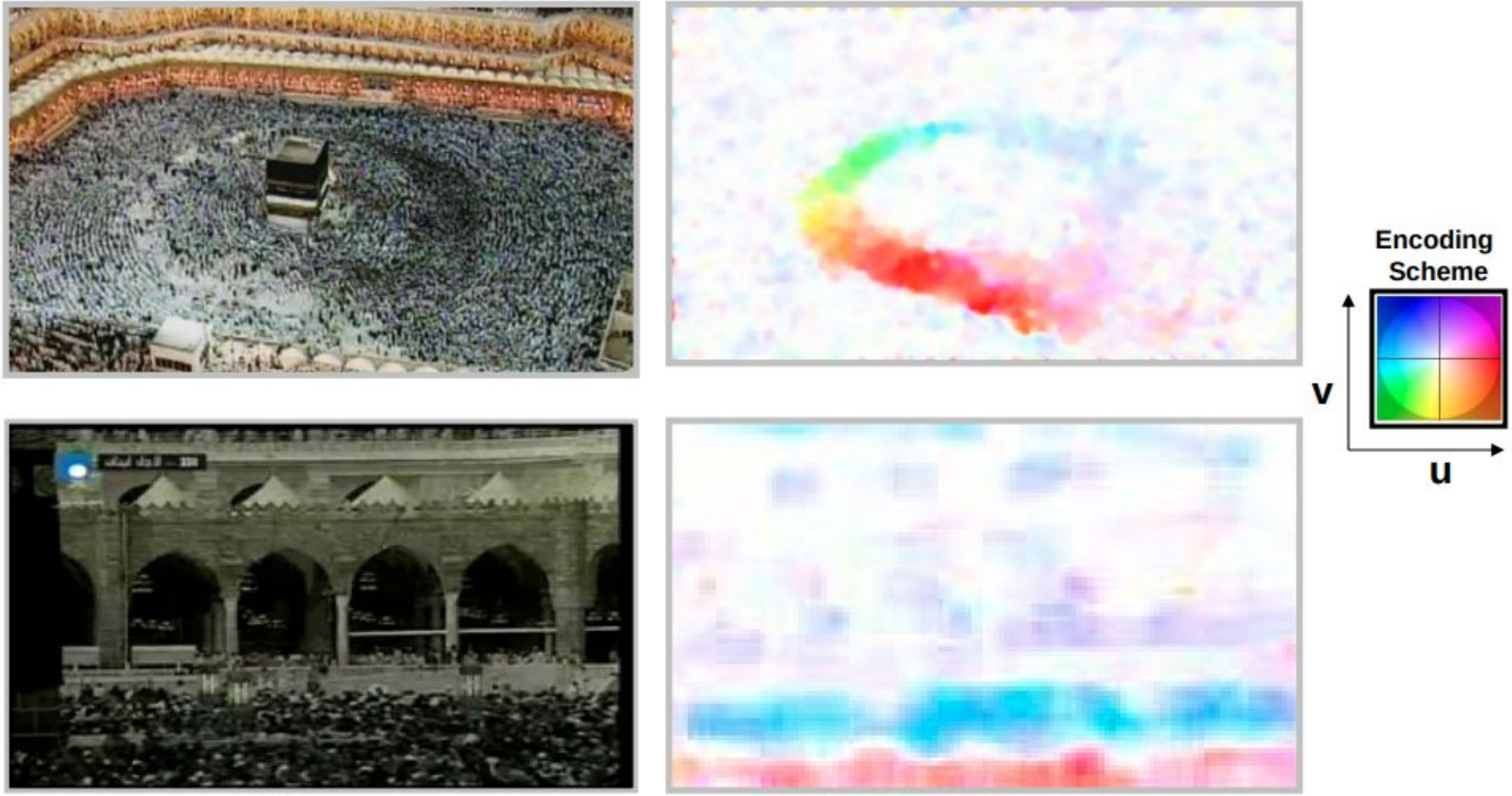
Stationary Sphere
Moving Light Source


Motion Field
Optical Flow

Motivation

Sparse

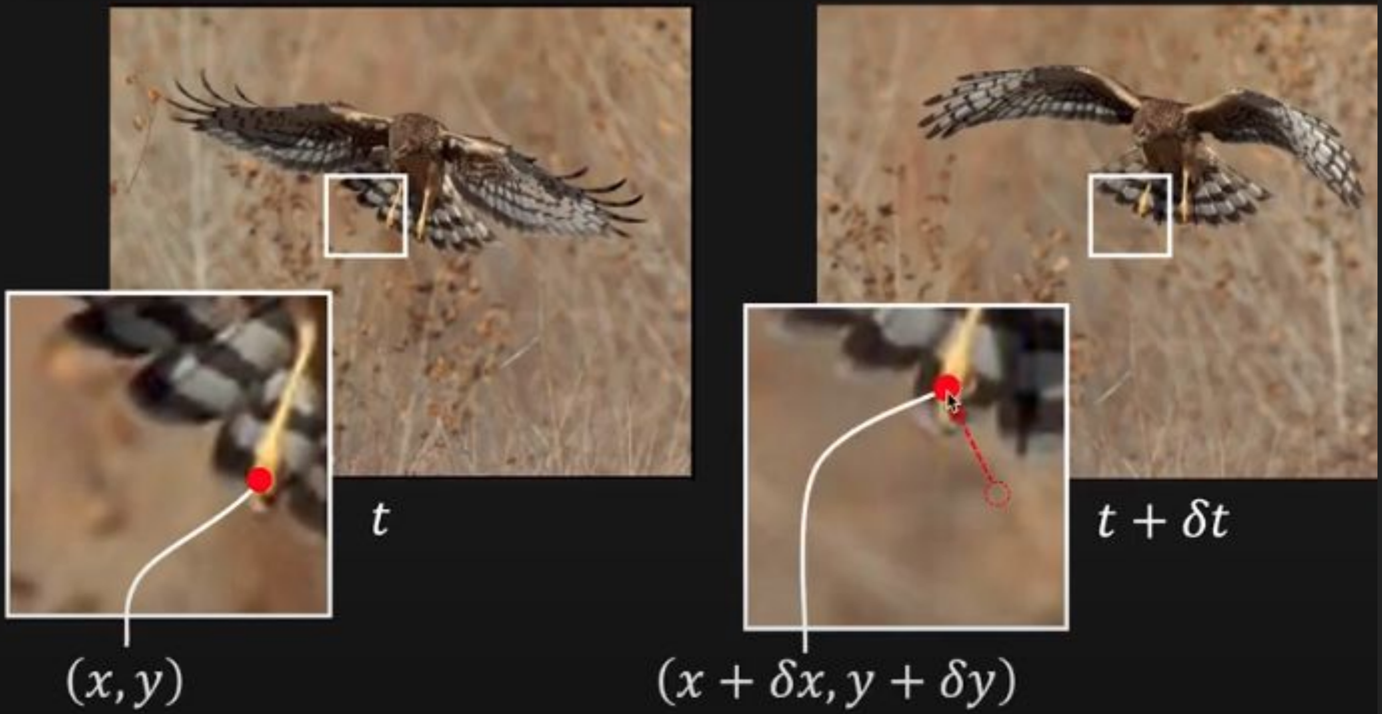
Dense



Videos

Color Coded Optical Flows

- Applications
  - Motion based segmentation
  - Structure from Motion(3D shape and Motion)
  - Alignment (Global motion compensation)
    - Camcorder video stabilization
    - UAV Video Analysis
  - Video Compression



Displacement: $(\delta x, \delta y)$ Optical Flow: $(u, v) = \left(\frac{\delta x}{\delta t}, \frac{\delta y}{\delta t}\right)$

Assumption #1:

Brightness of image point remains constant over time

$$\boxed{I(x+\delta x, y+\delta y, t+\delta t) = I(x, y, t)}$$

Assumption #2:

Displacement $(\delta x, \delta y)$ and time step δt are small

Taylor Series Expansion

Expand a function as an infinite sum of its derivatives

$$f(x+\delta x) = f(x) + \frac{\partial f}{\partial x}\delta x + \frac{\partial^2 f}{\partial x^2}\frac{\delta x^2}{2!} + \cdots + \frac{\partial^n f}{\partial x^n}\frac{\delta x^n}{n!}$$

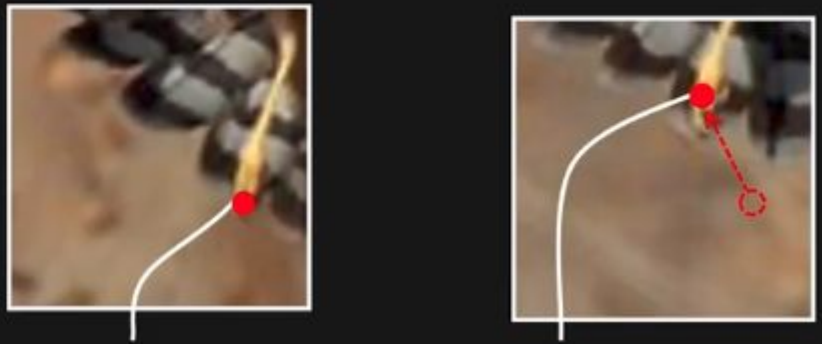
If δx is small:

$$f(x+\delta x) = f(x) + \frac{\partial f}{\partial x}\delta x + \boxed{O(\delta x^2)} \longrightarrow \text{Almost Zero}$$

For a function of three variables with small $\delta x, \delta y, \delta t$:

$$f(x+\delta x, y+\delta y, t+\delta t) \approx f(x,y,t) + \frac{\partial f}{\partial x}\delta x + \frac{\partial f}{\partial y}\delta y + \frac{\partial f}{\partial t}\delta$$

Optical Flow Constraint Equation



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

Assumption #2:

Displacement $(\delta x, \delta y)$ and time step δt are small

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

$$\boxed{I(x+\delta x, y+\delta y, t+\delta t) = I(x, y, t) + I_x \delta x + I_y \delta y + I_t \delta t}$$

Optical Flow Constraint Equation

$$I(x+\delta x, y+\delta y, t+\delta t) = I(x,y,t) \quad \text{------------ (1)}$$

$$I(x+\delta x, y+\delta y, t+\delta t) = I(x,y,t) + I_x \delta x + I_y \delta y + I_t \delta t \quad \text{------------ (2)}$$

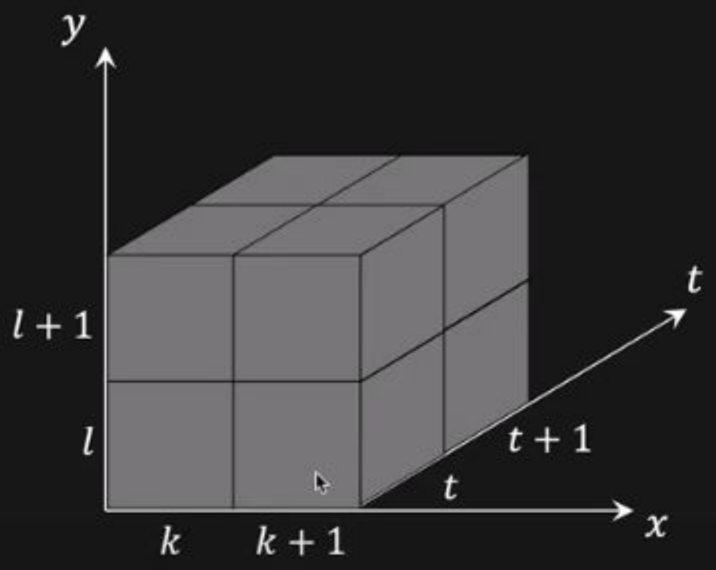
Subtract (1) from (2): $I_x \delta x + I_y \delta y + I_t \delta t = 0$

Divide by δt and take limit as $\delta t \to 0$: $I_x \dfrac{\partial x}{\partial t} + I_y \dfrac{\partial y}{\partial t} + I_t = 0$

Constraint Equation: $\boxed{I_x u + I_y v + I_t = 0}$ (u, v): Optical Flow

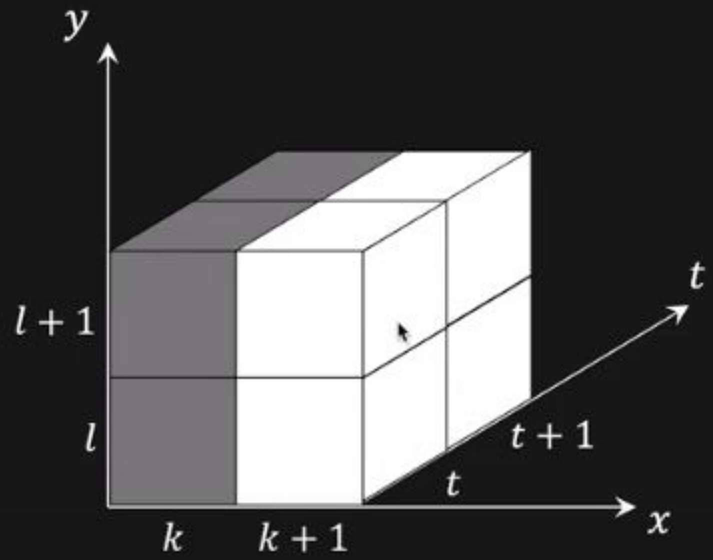
(I_x, I_y, I_t) can be easily computed from two frames

Computing Partial Derivatives I_x, I_y, I_t



$I_x(k,l,t) =$

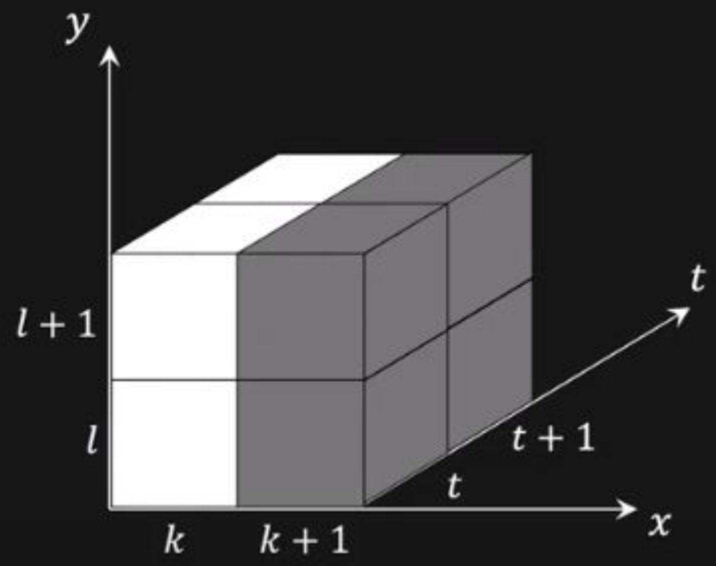
Computing Partial Derivatives I_x, I_y, I_t



$I_x(k,l,t) =$

$$\frac{1}{4}[I(k+1,l,t) + I(k+1,l,t+1) + I(k+1,l+1,t) + I(k+1,l+1,t+1)]$$

Computing Partial Derivatives I_x, I_y, I_t

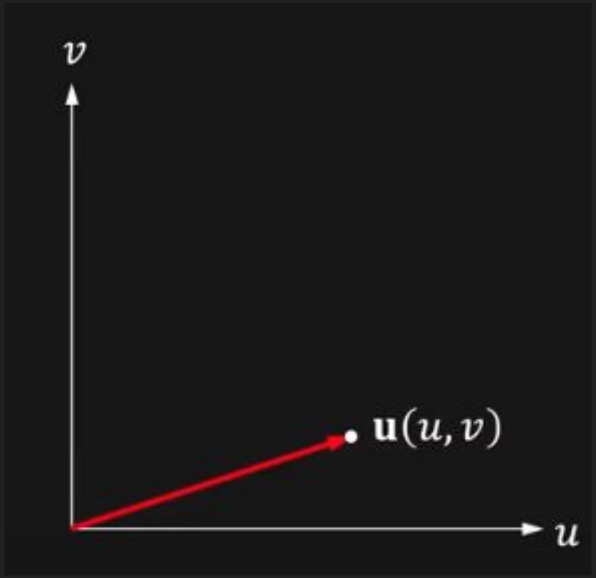


$$I_x(k,l,t) =$$

$$\tfrac{1}{4}[I(k+1,l,t) + I(k+1,l,t+1) + I(k+1,l+1,t) + I(k+1,l+1,t+1)]$$

$$-\tfrac{1}{4}[I(k,l,t) + I(k,l,t+1) + I(k,l+1,t) + I(k,l+1,t+1)]$$

Geometric Interpretation



For any point (x, y) in the image, its optical flow (u, v) lies on the line:

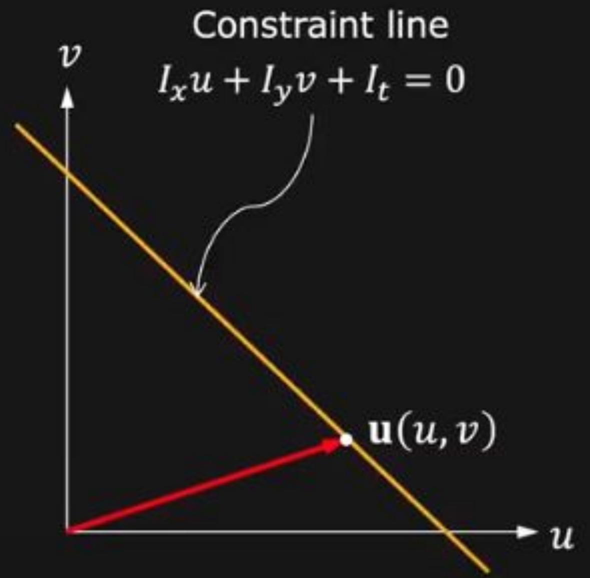
$$I_x u + I_y v + I_t = 0$$

Optical Flow can be split into two components.

$$\mathbf{u} = \mathbf{u}_n + \mathbf{u}_p$$

$\mathbf{u}_n$: Normal Flow

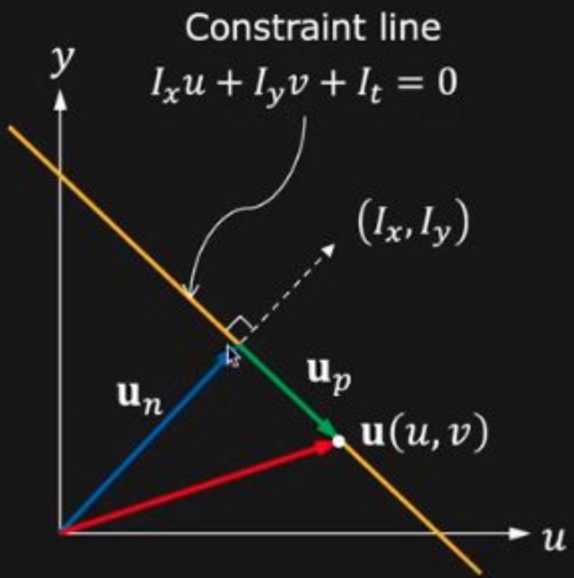
$\mathbf{u}_p$: Parallel Flow



Direction of Normal Flow:

Unit vector perpendicular to the constraint line:

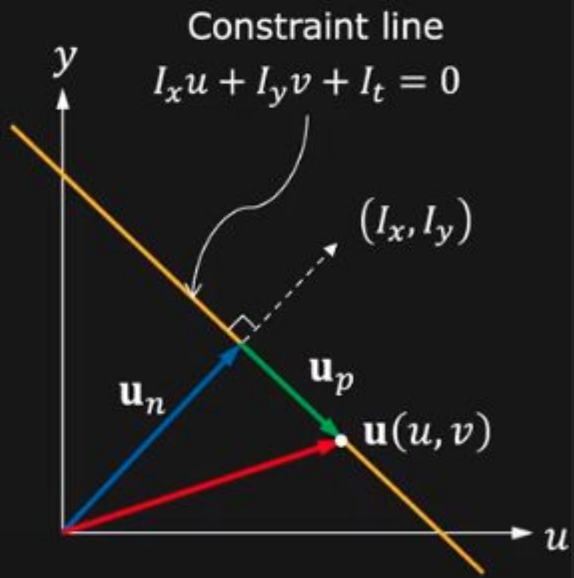
$$\hat{\mathbf{u}}_n = \frac{(I_x, I_y)}{\sqrt{I_x^2 + I_y^2}}$$



Direction of Normal Flow:

Unit vector perpendicular to the constraint line:

$$\hat{\mathbf{u}}_n = \frac{(I_x, I_y)}{\sqrt{I_x^2 + I_y^2}}$$



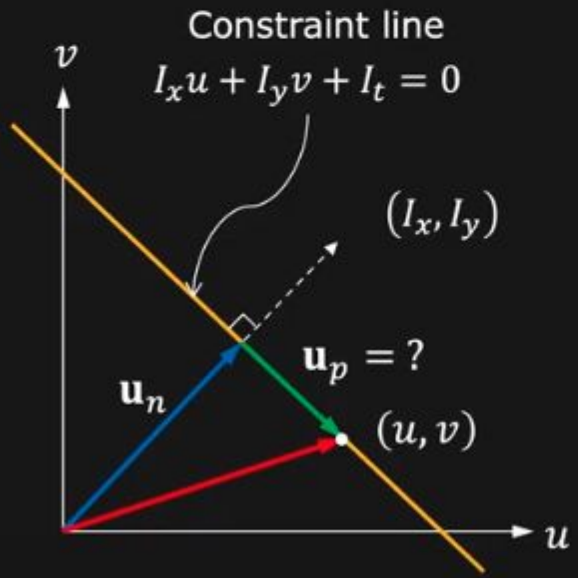
Magnitude of Normal Flow:

Distance of origin from the constraint line:

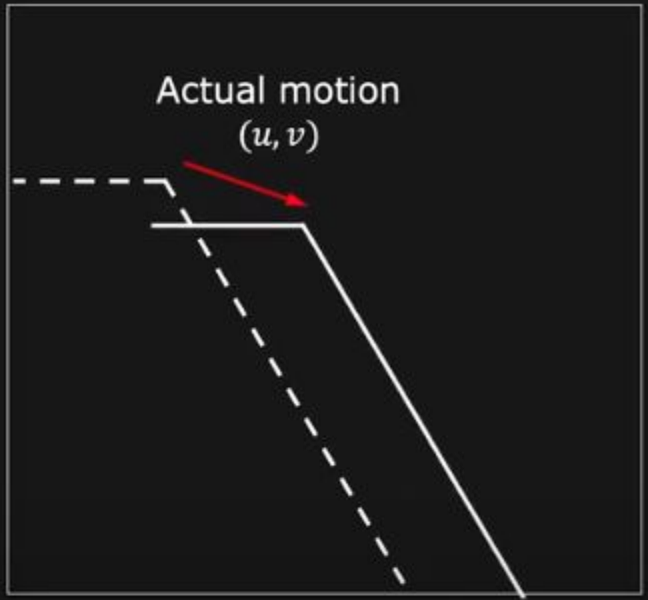
$$|\mathbf{u}_n| = \frac{|I_t|}{\sqrt{I_x^2 + I_y^2}}$$

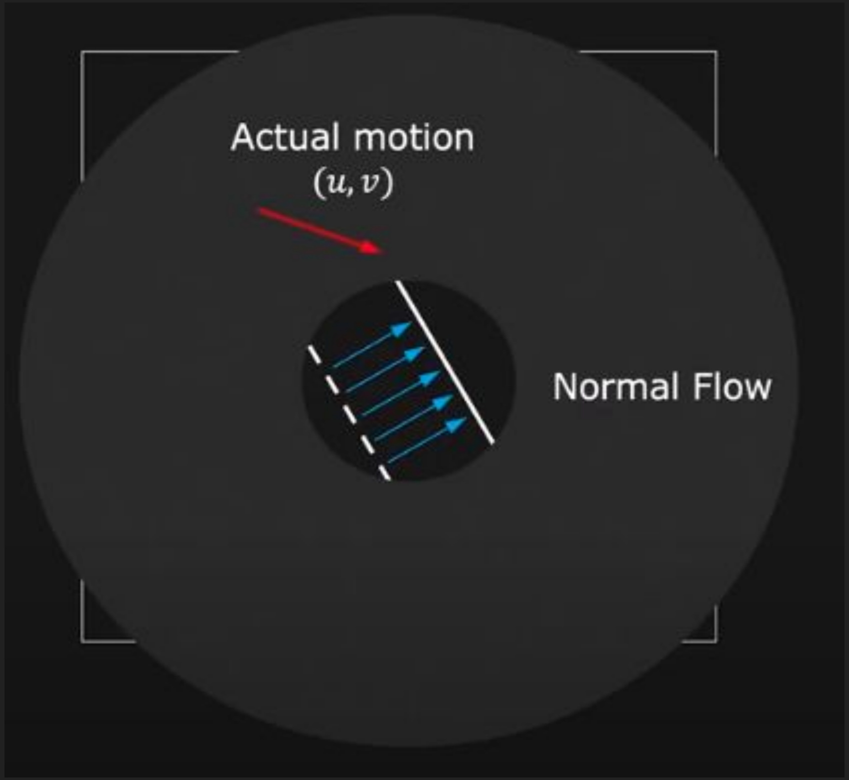
$$\boxed{\mathbf{u}_n = \frac{|I_t|}{(I_x^2 + I_y^2)}(I_x, I_y)}$$

We cannot determine $\mathbf{u}_p$, the optical flow component parallel to the constraint line.



Aperture Problem





Actual motion
(u, v)
Normal Flow

Constraint Equation: $\boxed{I_x u + I_y v + I_t = 0}$

2 unknowns, 1 equation.

Lucas Kanade Solution

Assumption: For each pixel, assume Motion Field, and hence Optical Flow (u, v), is constant within a small neighborhood W.

That is for all points $(k, l) \in W$:

$$I_x(k,l)u + I_y(k,l)v + I_t(k,l) = 0$$

For all points $(k,l) \in W$: $I_x(k,l)u + I_y(k,l)v + I_t(k,l) = 0$

Let the size of window W be $n \times n$

In matrix form:

$$\begin{bmatrix} I_x(1,1) & I_y(1,1) \\ I_x(k,l) & I_y(k,l) \\ \vdots & \vdots \\ I_x(n,n) & I_y(n,n) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = \begin{bmatrix} I_t(1,1) \\ I_t(k,l) \\ \vdots \\ I_t(n,n) \end{bmatrix}$$

For all points $(k,l) \in W$: $I_x(k,l)u + I_y(k,l)v + I_t(k,l) = 0$

Let the size of window W be $n \times n$

In matrix form:

$$\underbrace{\boxed{\begin{bmatrix} I_x(1,1) & I_y(1,1) \\ I_x(k,l) & I_y(k,l) \\ \vdots & \vdots \\ I_x(n,n) & I_y(n,n) \end{bmatrix}}}_{A} \underbrace{\boxed{\begin{bmatrix} u \\ v \end{bmatrix}}}_{\mathbf{u}} = \underbrace{\boxed{\begin{bmatrix} I_t(1,1) \\ I_t(k,l) \\ \vdots \\ I_t(n,n) \end{bmatrix}}}_{B}$$

A	$\mathbf{u}$	B
(Known)	(Unknown)	(Known)
$n^2 \times 2$	2×1	$n^2 \times 1$

Solve linear system: $A\mathbf{u} = B$

$$A^T A\mathbf{u} = A^T B$$ (Least-Squares using Pseudo-Inverse)

In matrix form:

$$\underbrace{\boxed{\begin{bmatrix} \sum_W I_x I_x & \sum_W I_x I_y \\ \sum_W I_x I_y & \sum_W I_y I_y \end{bmatrix}}}_{A^T A} \underbrace{\boxed{\begin{bmatrix} u \\ v \end{bmatrix}}}_{\mathbf{u}} = \underbrace{\boxed{\begin{bmatrix} -\sum_W I_x I_t \\ -\sum_W I_y I_t \end{bmatrix}}}_{A^T B}$$

Indices (k, l) not written for simplicity

$A^T A$	$\mathbf{u}$	$A^T B$
(Known)	(Unknown)	(Known)
2×2	2×1	2×1

$$\boxed{\mathbf{u} = (A^T A)^{-1} A^T B}$$

Fast and Easy to Solve

When Does Optical Flow Estimation Work?

$$\boxed{A\mathbf{u} = B} \qquad \boxed{A^T A\mathbf{u} = A^T B}$$

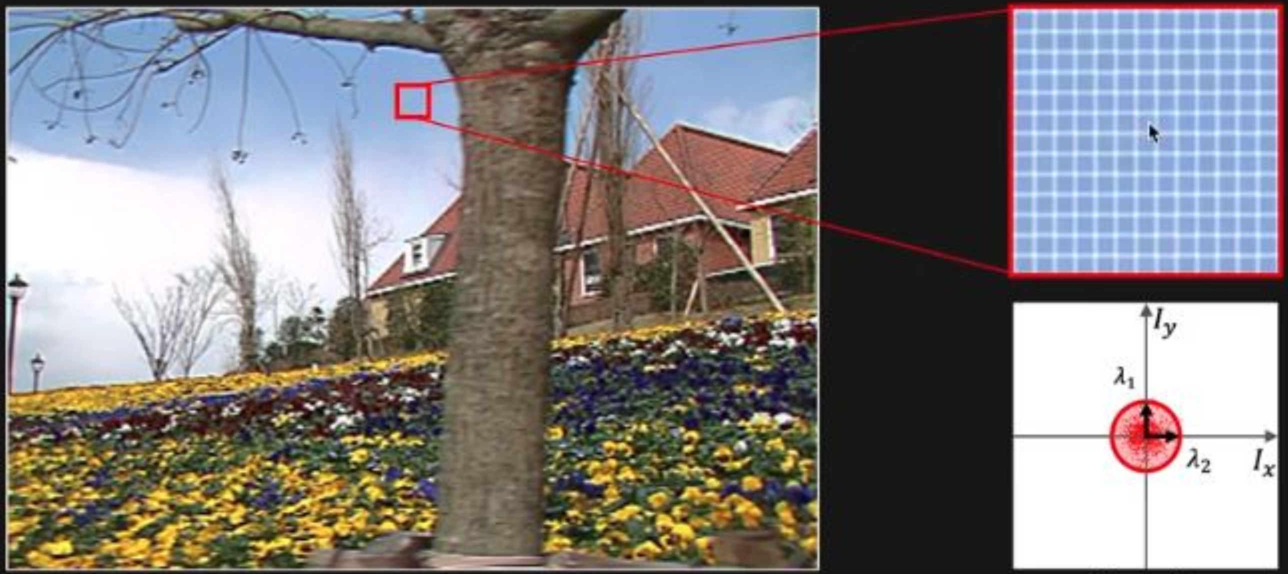
- $A^T A$ must be invertible. That is $\det(A^T A) \neq 0$
- $A^T A$ must be well-conditioned.

  If λ_1 and λ_2 are eigen values of $A^T A$, then

  $\lambda_1 > \epsilon$ and $\lambda_2 > \epsilon$

  $\lambda_1 \geq \lambda_2$ but not $\lambda_1 \ggg \lambda_2$

Smooth Regions (Bad)

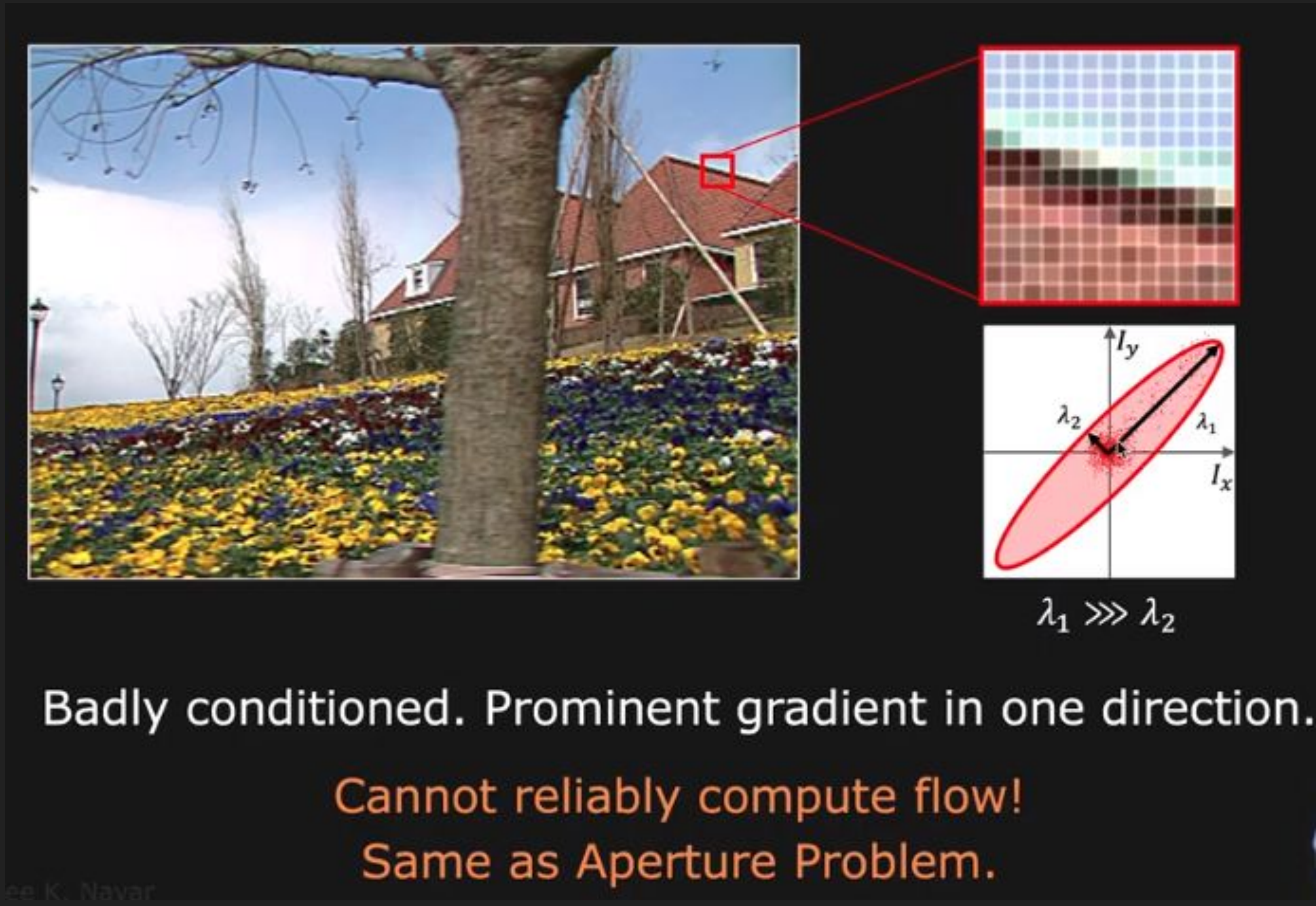


$\lambda_1 \sim \lambda_2$
Both are Small

Equations for all pixels in window are more or less the sam

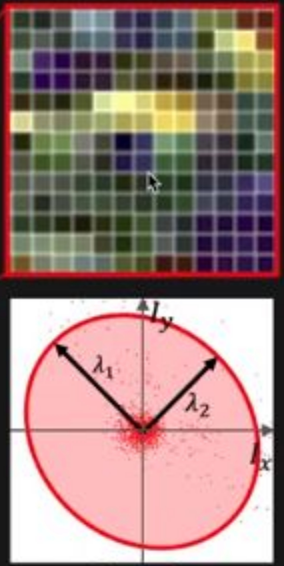
Cannot reliably compute flow!

Edges (bad)



I_y
I_x
λ_1
λ_2
$\lambda_1 \ggg \lambda_2$
Badly conditioned. Prominent gradient in one direction.
Cannot reliably compute flow!
Same as Aperture Problem.

Textured Regions (Good)

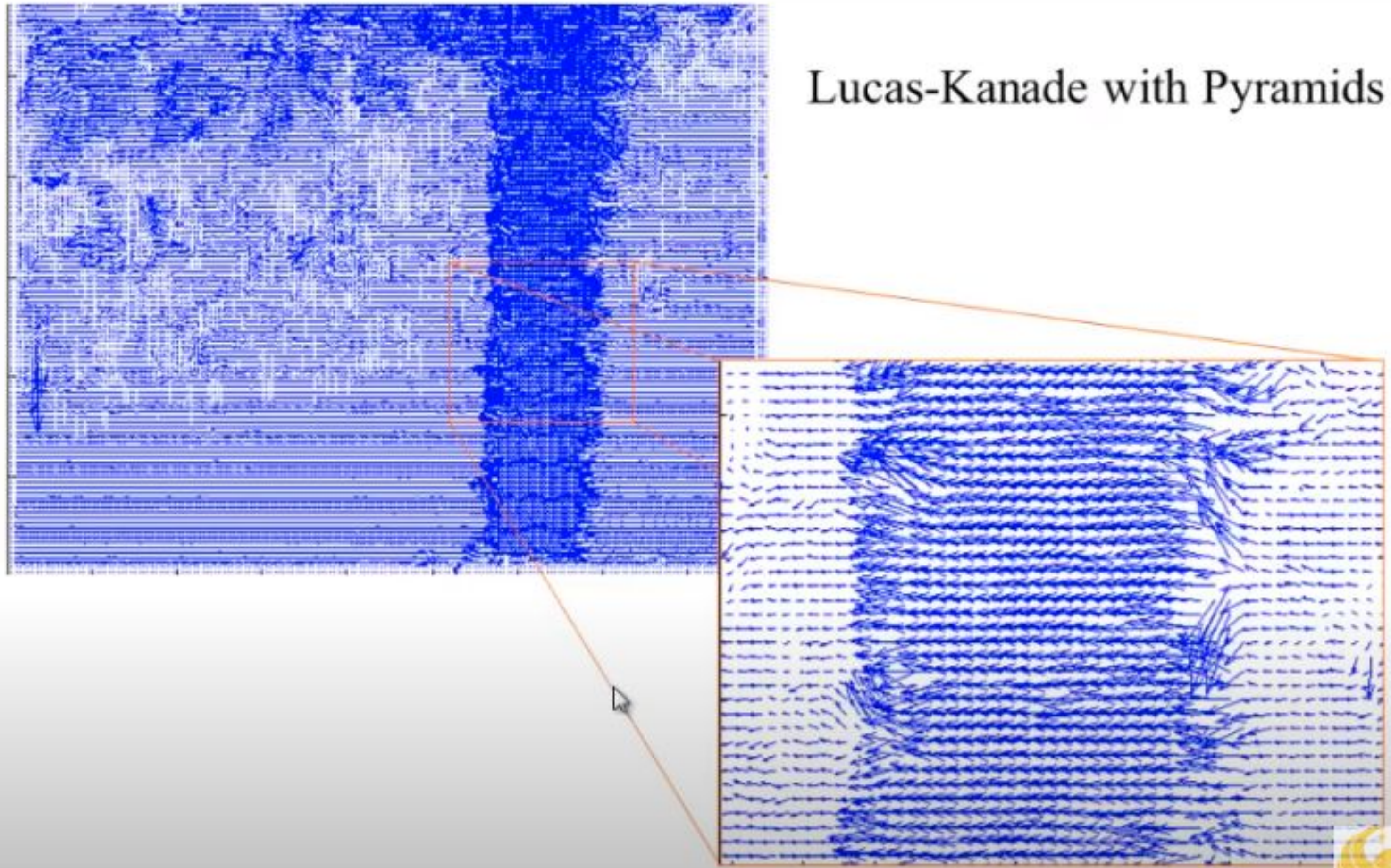


$\lambda_1 \sim \lambda_2$
Both are Large

Well conditioned. Large and diverse gradient magnitudes

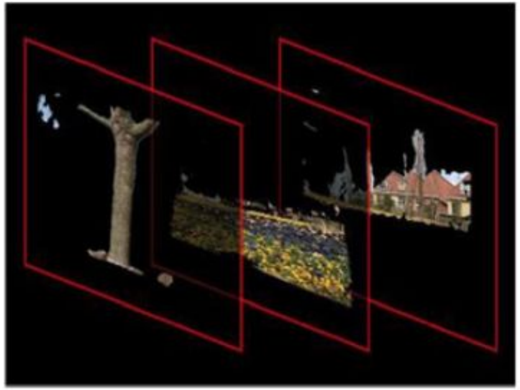
Can reliably compute optical flow.

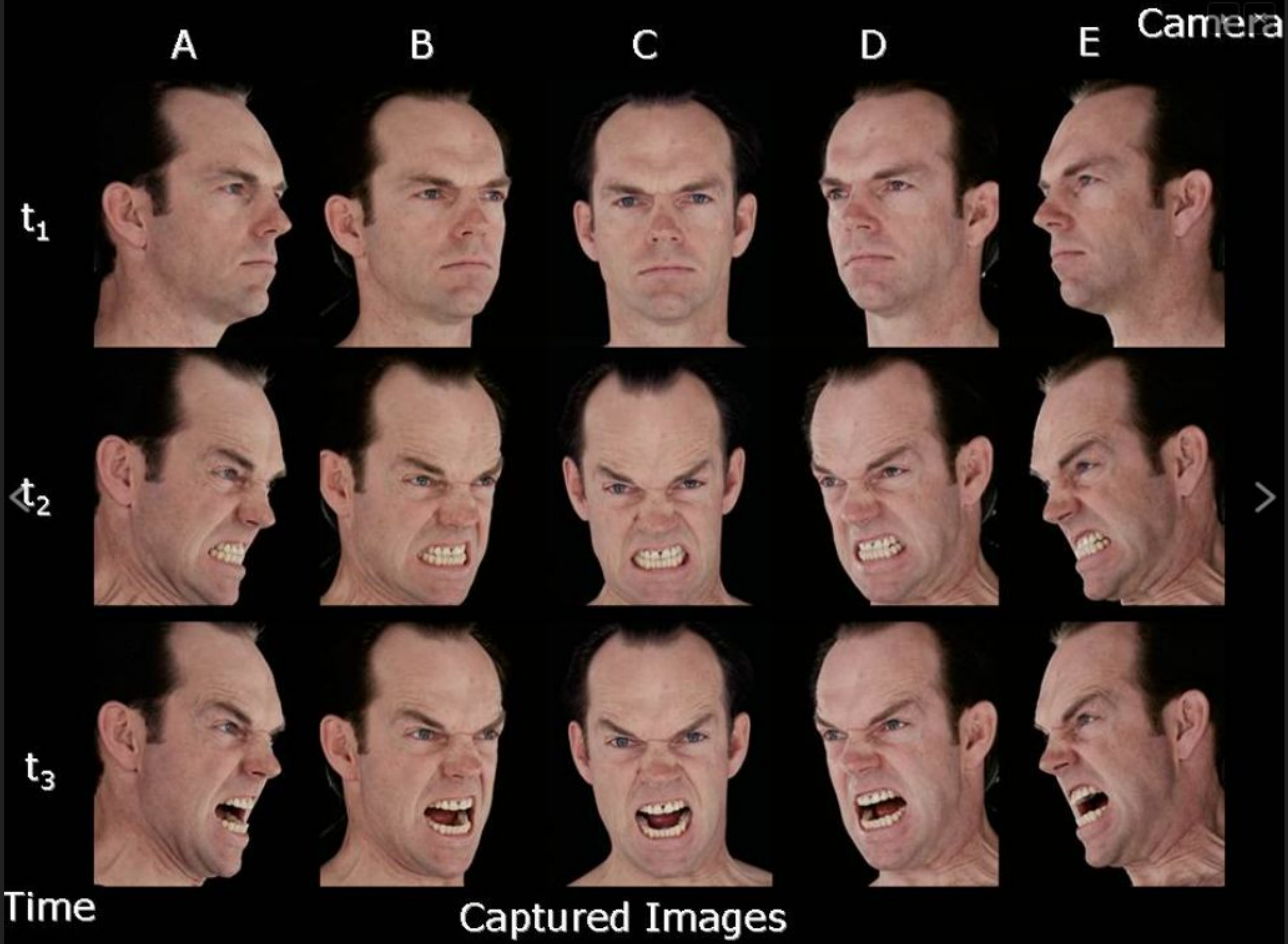
Lucas-Kanade with Pyramids



Comments

- Horn-Schunck and Lucas-Kanade optical methods work only for small motion.
- If object moves faster, the brightness changes rapidly,
  - 2x2 or 3x3 masks fail to estimate spatiotemporal derivatives.
- Pyramids can be used to compute large optical flow vectors.




Camera
A
B
C
D
E
t1
t2
t3
Time
Captured Images

What if we have Large Motion?

What if we have Large Motion?

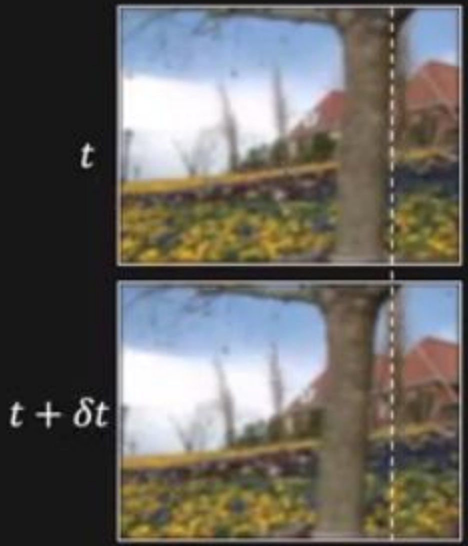
Taylor Series approximation of

$I(x + \delta x, y + \delta y, t + \delta t)$ is not valid

Our simple linear constraint equation not valid

$$I_x u + I_y v + I_t \neq 0$$

Large Motion: Coarse-to-Fine Estimation

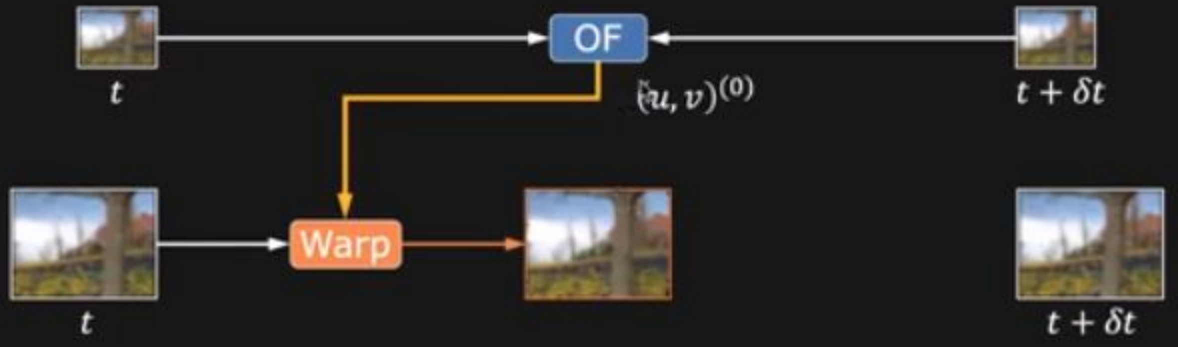


Resolution
$N \times N$

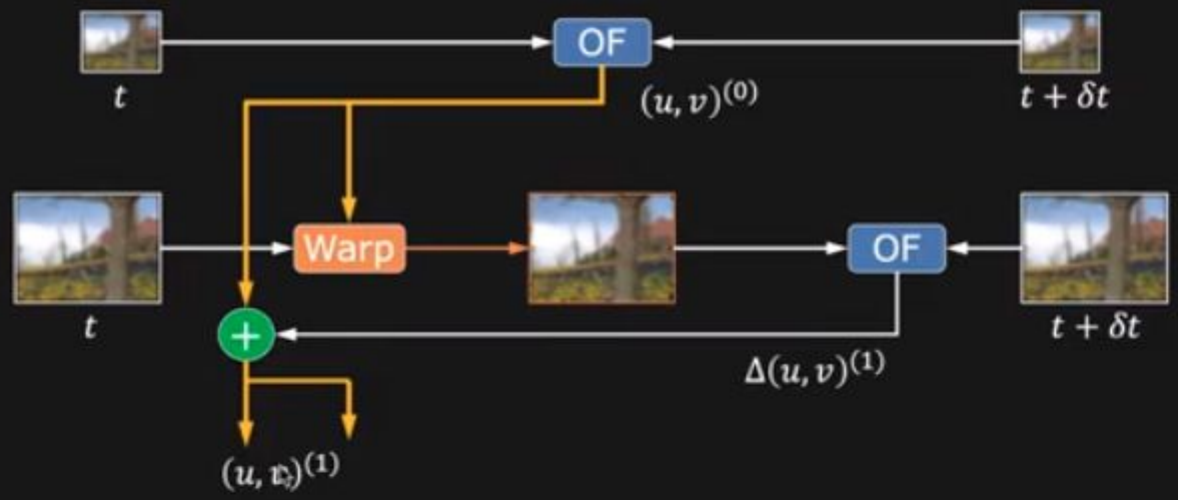
At lowest resolution, motion $\leq$ 1 pixel

Coarse-to-Fine Estimation Algorithm

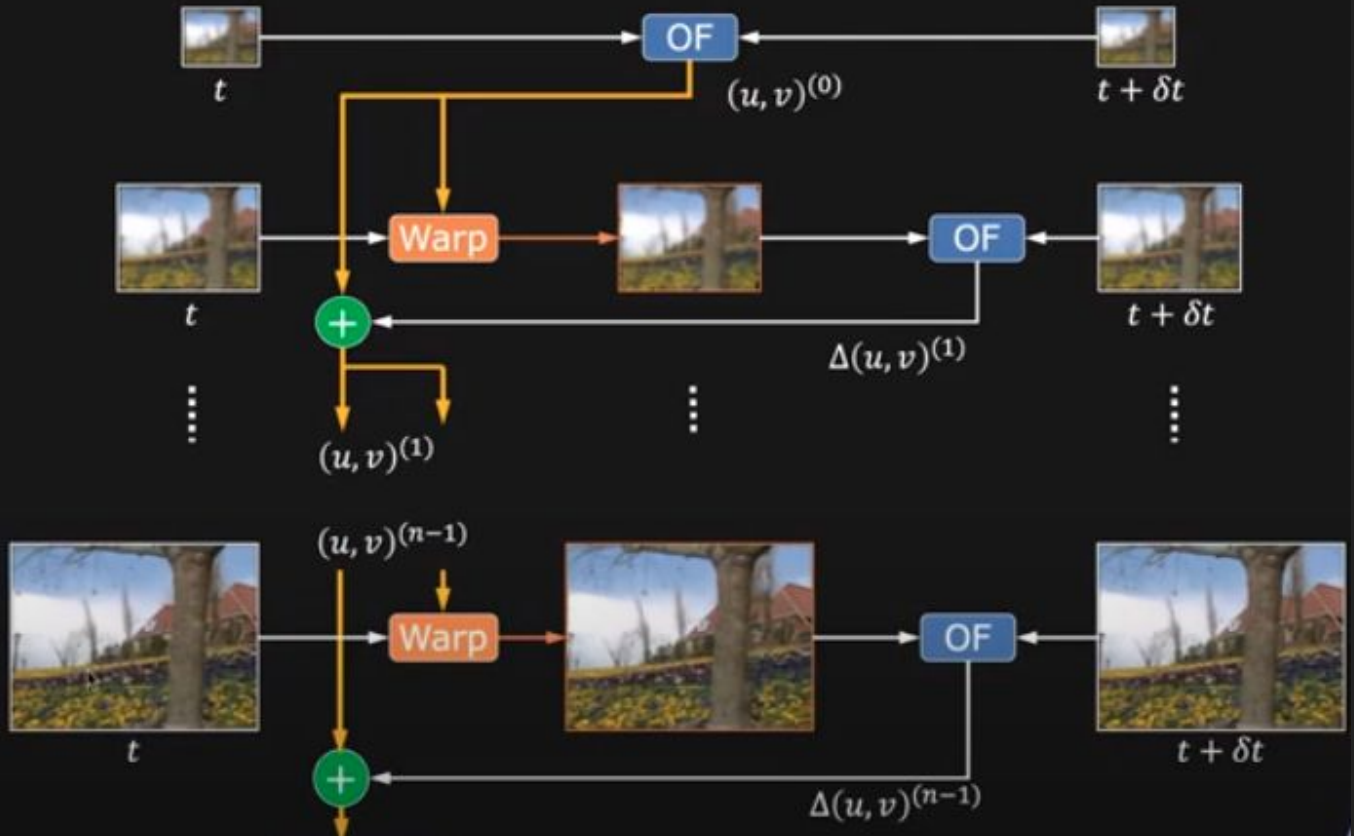
Coarse-to-Fine Estimation Algorithm



Coarse-to-Fine Estimation Algorithm



Coarse-to-Fine Estimation Algorithm



Alternative Approach: Template Matching

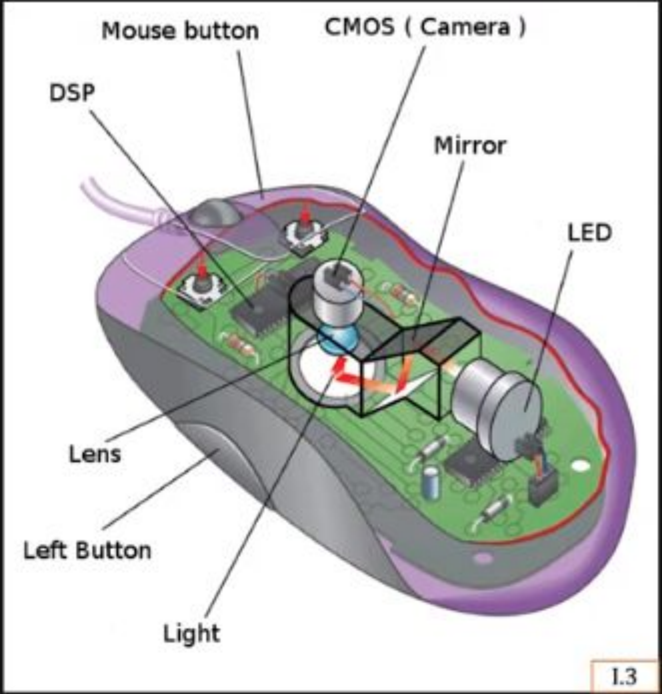
Determine Flow using Template Matching

Image I_1 at time t

Image I_2 at time $t + \delta t$

For each template window T in image I_1,
find the corresponding match in image I_2.

Optical Mouse



Estimating Mouse Movements

Traffic Monitoring

Finding Velocities of Vehicles

([source](#))