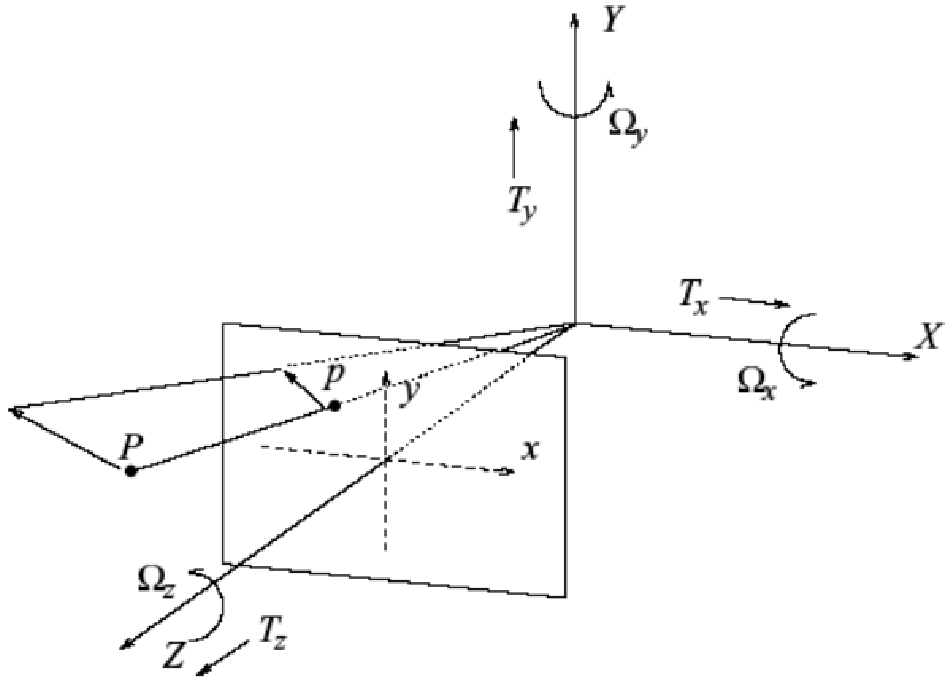
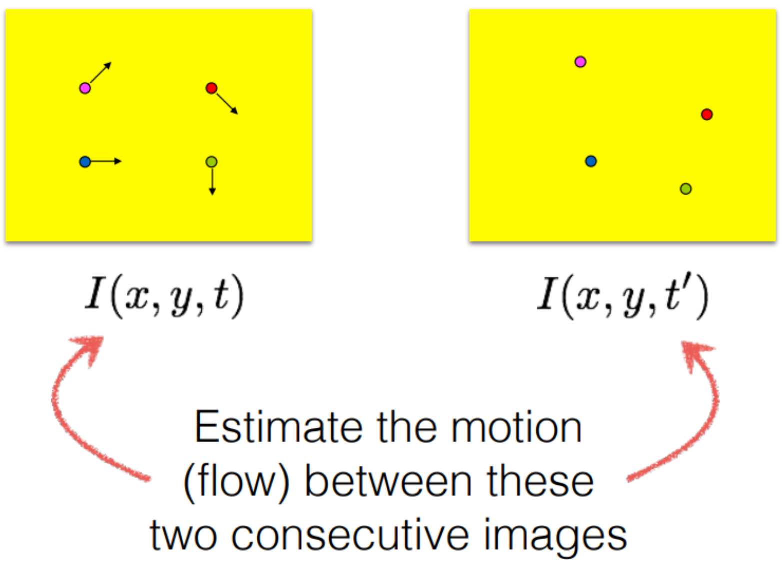
Optical Flow



Optical Flow
2D velocity field describing the apparent motion in the images

Problem Setting

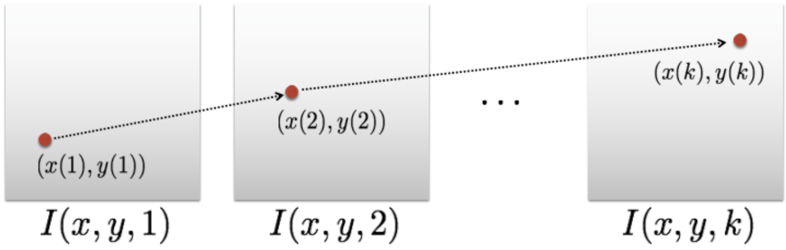
Background



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

Key assumptions:

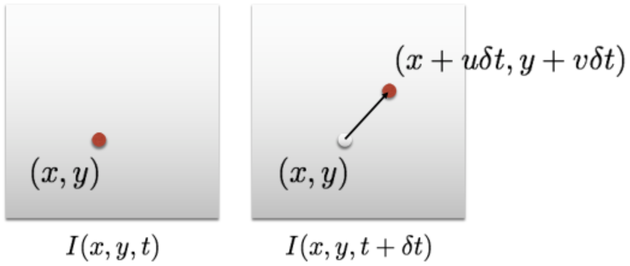
- Color constancy (brightness constancy): brightness of the point will remain the same



$$I(x(t), y(t), t) = \underset{\text{const}}{C}$$

Allows for **pixel to pixel comparison** (not features)

- Small motion: pixels only move a little bit



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

Brightness Constancy Equation

For a really small space-time step, corresponding pixels have the same intensity

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

Using the assumptions above we get the **Brightness Constancy Equation**

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

> **Proof**:
> Taylor expansion
>
> $$\begin{aligned} I(x + \delta x, y + \delta y, t + \delta t) &= I(x, y, t) + \frac{\partial I}{\partial x}\delta t + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t \\ &= I(x, y, t) \\ \implies \frac{\partial I}{\partial x}\delta t + \frac{\partial I}{\partial y}\delta y + \frac{\partial I}{\partial t}\delta t = 0 &\quad \text{divide by } \delta t \\ \implies \frac{\partial I}{\partial x}\frac{dx}{dt} + \frac{\partial I}{\partial y}\frac{dy}{dt} + \frac{\partial I}{\partial t} = 0 \end{aligned}$$
>
> Equivalently written as
>
> $$I_x u + I_y v + I_t = 0$$
>
> and the vector form is
>
> $$\nabla I^\top \mathbf{v} + I_t = 0$$

$I_x u + I_y v + I_t = 0$	**Spatial Derivative**	**Optical Flow**	**Temporal Derivative**
Formula	$I_x = \frac{\partial I}{\partial x} \quad I_y = \frac{\partial I}{\partial y}$	$u = \frac{dx}{dt} \quad v = \frac{dy}{dt}$	$I_t = \frac{\partial I}{\partial t}$
Calculation	Sobel filter, Derivative-of-Gaussian filter	(u, v) solution lies on a line, cannot be found uniquely with a single constrain	Frame differencing

We already know how to compute the gradients. The temporal derivative is calculated by frame differencing:

$$\underset{t}{\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 10 & 10 & 10 & 10 \\ 1 & 10 & 10 & 10 & 10 \\ 1 & 10 & 10 & 10 & 10 \\ 1 & 10 & 10 & 10 & 10 \end{bmatrix}} - \underset{t+1}{\begin{bmatrix} 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 1 & 1 & 1 \\ 1 & 1 & 10 & 10 & 10 \\ 1 & 1 & 10 & 10 & 10 \\ 1 & 1 & 10 & 10 & 10 \end{bmatrix}} = \underset{I_t = \frac{\partial I}{\partial t}}{\begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 \\ 0 & 9 & 9 & 9 & 9 \\ 0 & 9 & 0 & 0 & 0 \\ 0 & 9 & 0 & 0 & 0 \\ 0 & 9 & 0 & 0 & 0 \end{bmatrix}}$$

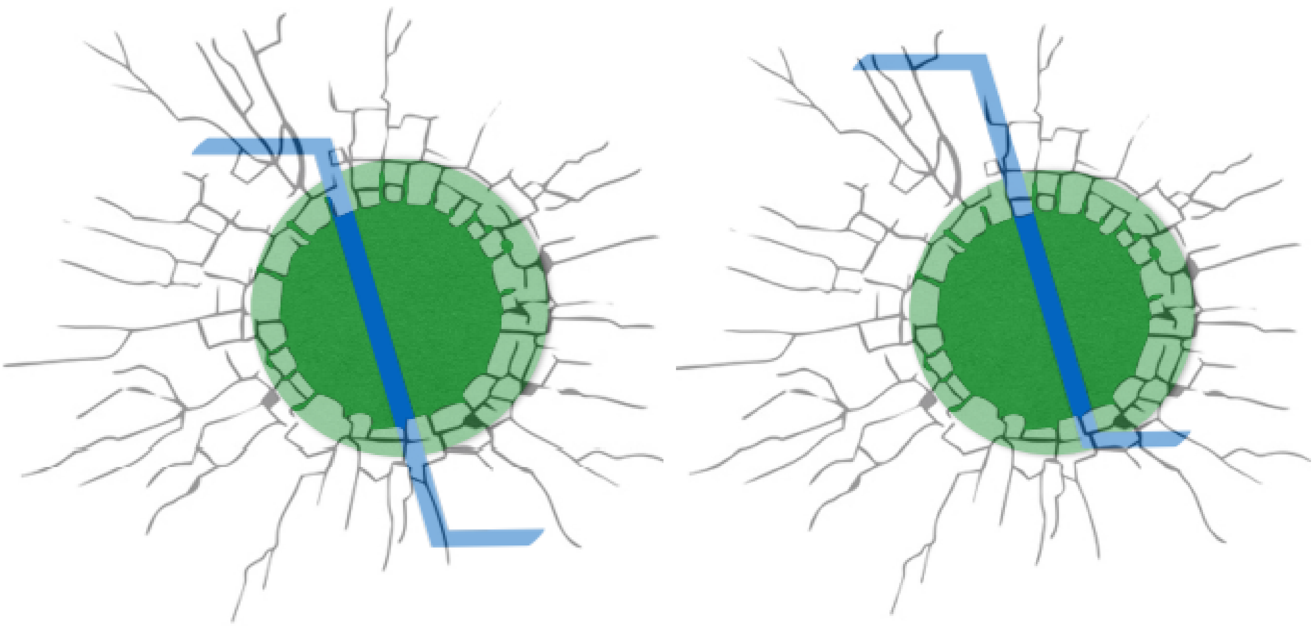
Apparently this equation

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

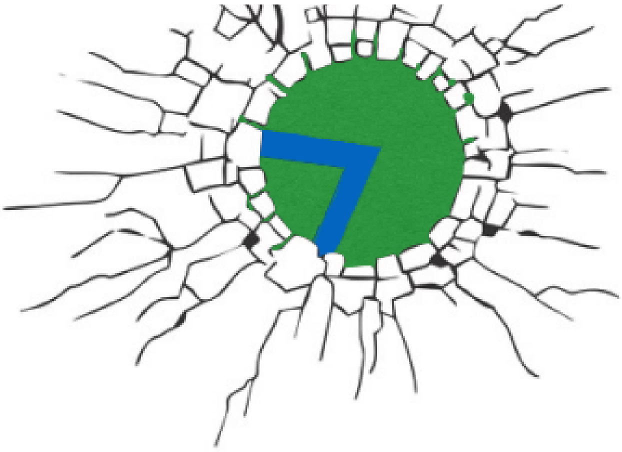
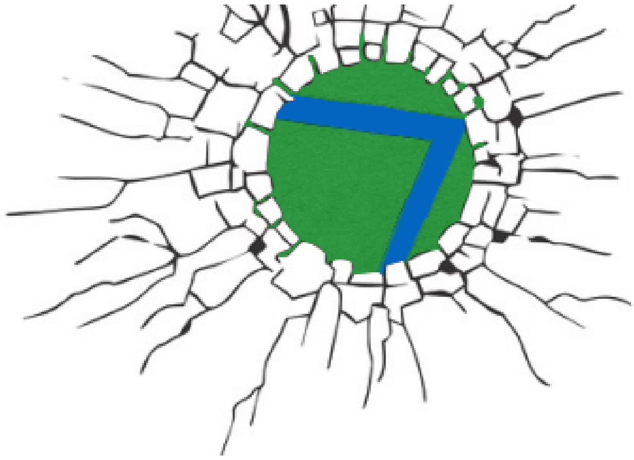
does not have a unique solution. We need more constraints to solve (u, v).

Small Aperture Problem

When the image patch contains only a line, we could not perceive the motion.



But patches with different gradients could avoid aperture problem

Lucas-Kanade Optical Flow

Characterization

- **Constant** flow: flow is constant for all pixels
- **Local** method: sparse system

Assumptions

- Flow is locally smooth
- Neighboring pixels have the same displacement

Method

Consider a 5×5 image patch, which gives us 25 equations

$$\begin{gathered} I_x(\mathbf{p}_1)u + I_y(\mathbf{p}_1)v = -I_t(\mathbf{p}_1) \\ I_x(\mathbf{p}_2)u + I_y(\mathbf{p}_2)v = -I_t(\mathbf{p}_2) \\ \vdots \\ I_x(\mathbf{p}_{25})u + I_y(\mathbf{p}_{25})v = -I_t(\mathbf{p}_{25}) \end{gathered}$$

which is equivalent to

$$A^\top A\mathbf{x} = -A^\top \mathbf{b}$$

where

$$A = \begin{pmatrix} I_x(\mathbf{p}_1) & I_y(\mathbf{p}_1) \\ I_x(\mathbf{p}_2) & I_y(\mathbf{p}_2) \\ \vdots & \vdots \\ I_x(\mathbf{p}_{25}) & I_y(\mathbf{p}_{25}) \end{pmatrix}$$

and the solution writes

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

The factor $A^\top A$ is exactly the Harris corner detector

$$A^\top A = \begin{bmatrix} \sum_{p \in P} I_x I_x & \sum_{p \in P} I_x I_y \\ \sum_{p \in P} I_y I_x & \sum_{p \in P} I_y I_y \end{bmatrix}$$

and the r.h.s. equals

$$-\begin{bmatrix}\sum I_x I_t \\ \sum I_y I_t\end{bmatrix}$$

$\Longrightarrow$ Implications:

- Corners are when λ_1, λ_2 are big; this is also when Lucas-Kanade optical flow works best
- Corners are regions with two different directions of gradient
- Corners are good places to computer flow

Horn-Schunck Optical Flow

Characterization

- **Smooth** flow: flow can vary from pixel to pixel
- **Global** method: dense system

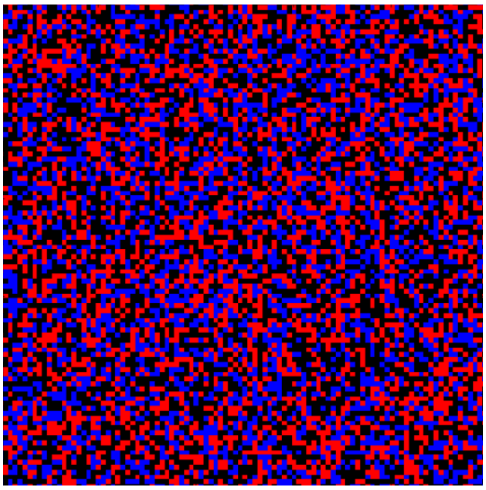
Idea:

1. Enforce brightness constancy
   For every pixel

$$\min_{u,v} \sum_{i,j} [I_x u_{ij} + I_y v_{ij} + I_t]^2$$

2. Enforce smooth flow field
   Consider the following penalty function to enforce smoothness

$$\min_{\mathbf{u}} \sum_{i,j} (\mathbf{u}_{i,j} - \mathbf{u}_{i+1,j})^2$$

	
Big penalty	Small penalty

Method

Imagine that we are in a continuous scalar filed I and vector filed $\mathbf{u} = (u, v)$, we want to minimize

$$E(u,v) = \underbrace{E_s(u,v)}_{\text{smoothness}} + \overbrace{\lambda}^{\text{weight}} \underbrace{E_d(u,v)}_{\text{brightness constancy}}$$

In continuous form

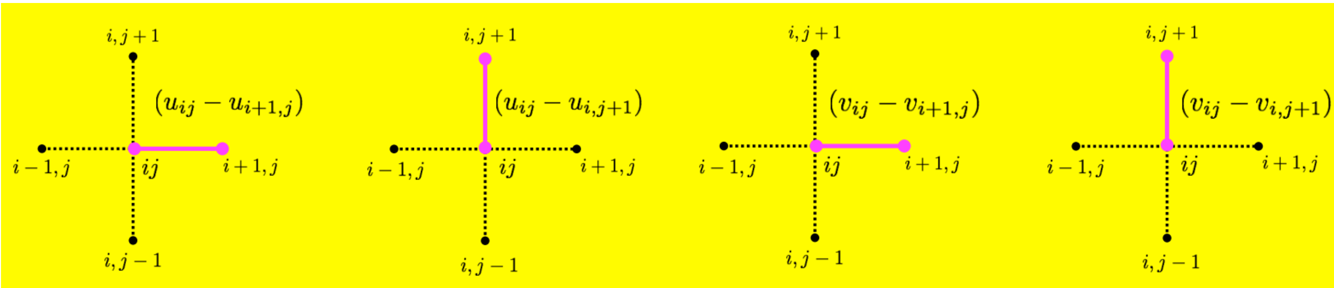
$$E(u,v) = \iint \underbrace{\left(I(x+u(x,y), y+v(x,y), t+1) - I(x,y,t)\right)^2}_{\text{quadratic penalty for brightness change}} + \lambda \underbrace{\left(\|\nabla u(x,y)\|^2 + \|\nabla v(x,y)\|^2\right)}_{\text{quadratic penalty for flow change}} dxdy$$

In discretized version, we have

$$\min_{u,v} \sum_{i,j} \{ \underbrace{E_s(i,j)}_{\text{smoothness}} + \underbrace{\lambda E_d(i,j)}_{\text{brightness constancy}} \}$$

where

$$\begin{aligned}
E_d(i,j) &= [I_x u_{ij} + I_y v_{ij} + I_t]^2 \\
E_s(i,j) &= \frac{1}{4}[(\mathbf{u}_{i+1,j} - \mathbf{u}_{i,j})^2 + (\mathbf{u}_{i,j+1} - \mathbf{u}_{i,j})^2] \\
&= \frac{1}{4}[(u_{i,j} - u_{i+1,j})^2 + (u_{i,j} - u_{i,j+1})^2 + (v_{i,j} - v_{i+1,j})^2 + (v_{i,j} - u_{i,j+1})^2]
\end{aligned}$$



To solve the minimization problem, we take the derivatives w.r.t u_{kl} and v_{kl}

$$\begin{aligned}
\frac{\partial E}{\partial u_{kl}} &= 2\left(u_{kl} - \bar{u}_{kl}\right) + 2\lambda\left(I_x u_{kl} + I_y v_{kl} + I_t\right) I_x \\
\frac{\partial E}{\partial v_{kl}} &= 2\left(v_{kl} - \bar{v}_{kl}\right) + 2\lambda\left(I_x u_{kl} + I_y v_{kl} + I_t\right) I_y
\end{aligned}$$

where $\bar{u}_{kl} = \frac{1}{4}(u_{i+1,j} + u_{i,j+1} + u_{i-1,j} + u_{i,j-1})$ and $\bar{v}_{kl} = \frac{1}{4}(v_{i+1,j} + v_{i,j+1} + v_{i-1,j} + v_{i,j-1})$.

Set these two equations to zero, we have

$$\begin{aligned}
\left(1 + \lambda I_x^2\right) u_{kl} + \lambda I_x I_y v_{kl} &= \bar{u}_{kl} - \lambda I_x I_t \\
\lambda I_x I_y u_{kl} + \left(1 + \lambda I_y^2\right) v_{kl} &= \bar{v}_{kl} - \lambda I_y I_t
\end{aligned}$$

Rearrange the equations to get

$$\begin{aligned}
\left\{1 + \lambda\left(I_x^2 + I_y^2\right)\right\} u_{kl} &= \left(1 + \lambda I_x^2\right) \bar{u}_{kl} - \lambda I_x I_y \bar{v}_{kl} - \lambda I_x I_t \\
\left\{1 + \lambda\left(I_x^2 + I_y^2\right)\right\} v_{kl} &= \left(1 + \lambda I_y^2\right) \bar{v}_{kl} - \lambda I_x I_y \bar{u}_{kl} - \lambda I_y I_t
\end{aligned}$$

which turns the problem into a recursion problem: while not converged, we update (u_{kl}, v_{kl}) as

$$\begin{aligned}
\hat{u}_{kl} &= \bar{u}_{kl} - \frac{I_x \bar{u}_{kl} + I_y \bar{v}_{kl} + I_t}{\lambda^{-1} + I_x^2 + I_y^2} I_x \\
\hat{v}_{kl} &= \bar{v}_{kl} - \frac{I_x \bar{u}_{kl} + I_y \bar{v}_{kl} + I_t}{\lambda^{-1} + I_x^2 + I_y^2} I_y
\end{aligned}$$