

Lecture 20: Motion estimation

Most slides from S. Lazebnik, which are based on other slides from S. Seitz, R. Szeliski, M. Pollefeys

Announcements

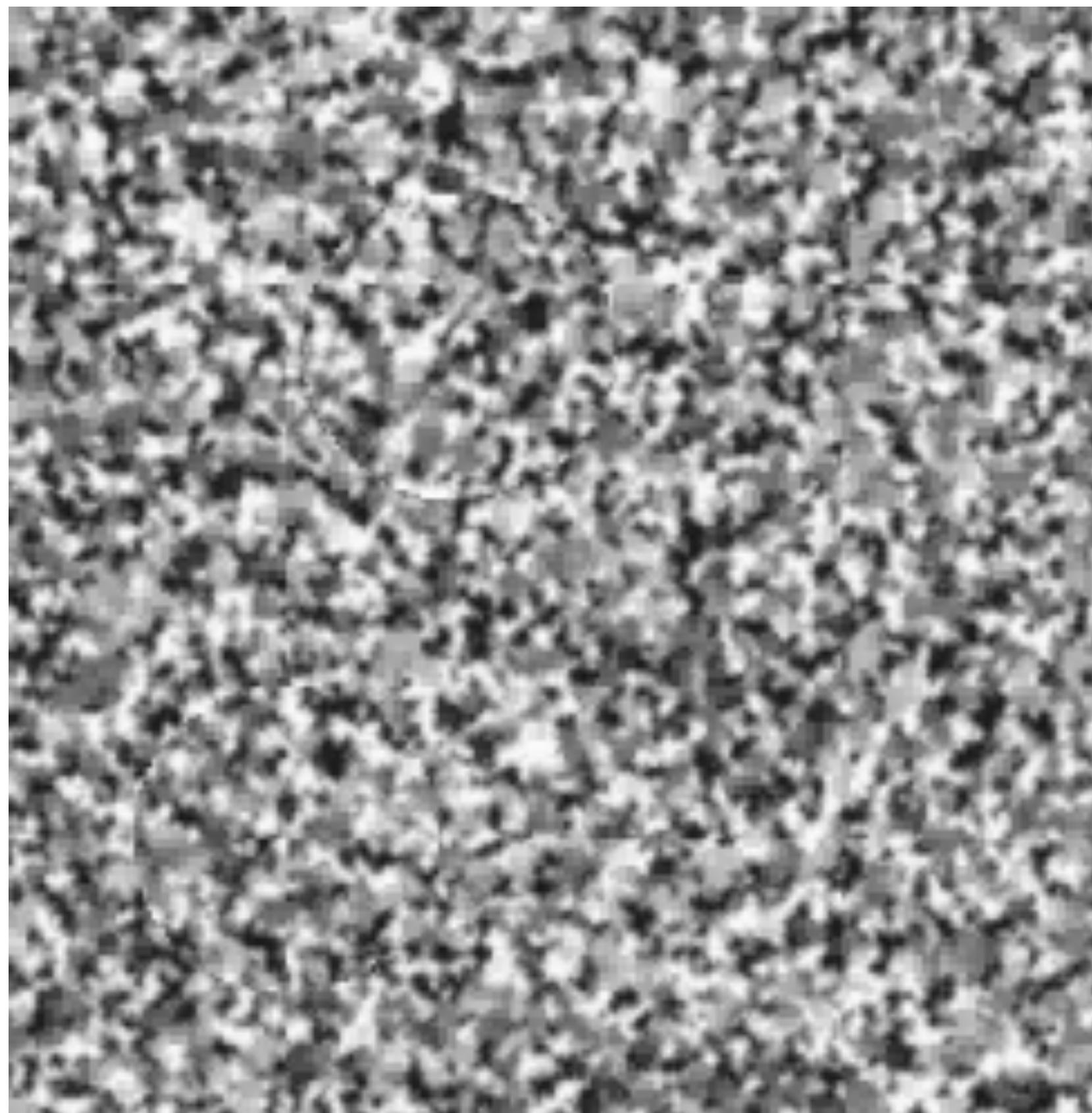
- PS9 and PS10 deadlines extended to April 21

Optical flow



What moved where?

Motion is a powerful perceptual cue



Motion is a powerful perceptual cue



G. Johansson, "Visual Perception of Biological Motion and a Model For Its Analysis",
Perception and Psychophysics 14, 201-211, 1973.
5

Source: S. Lazebnik

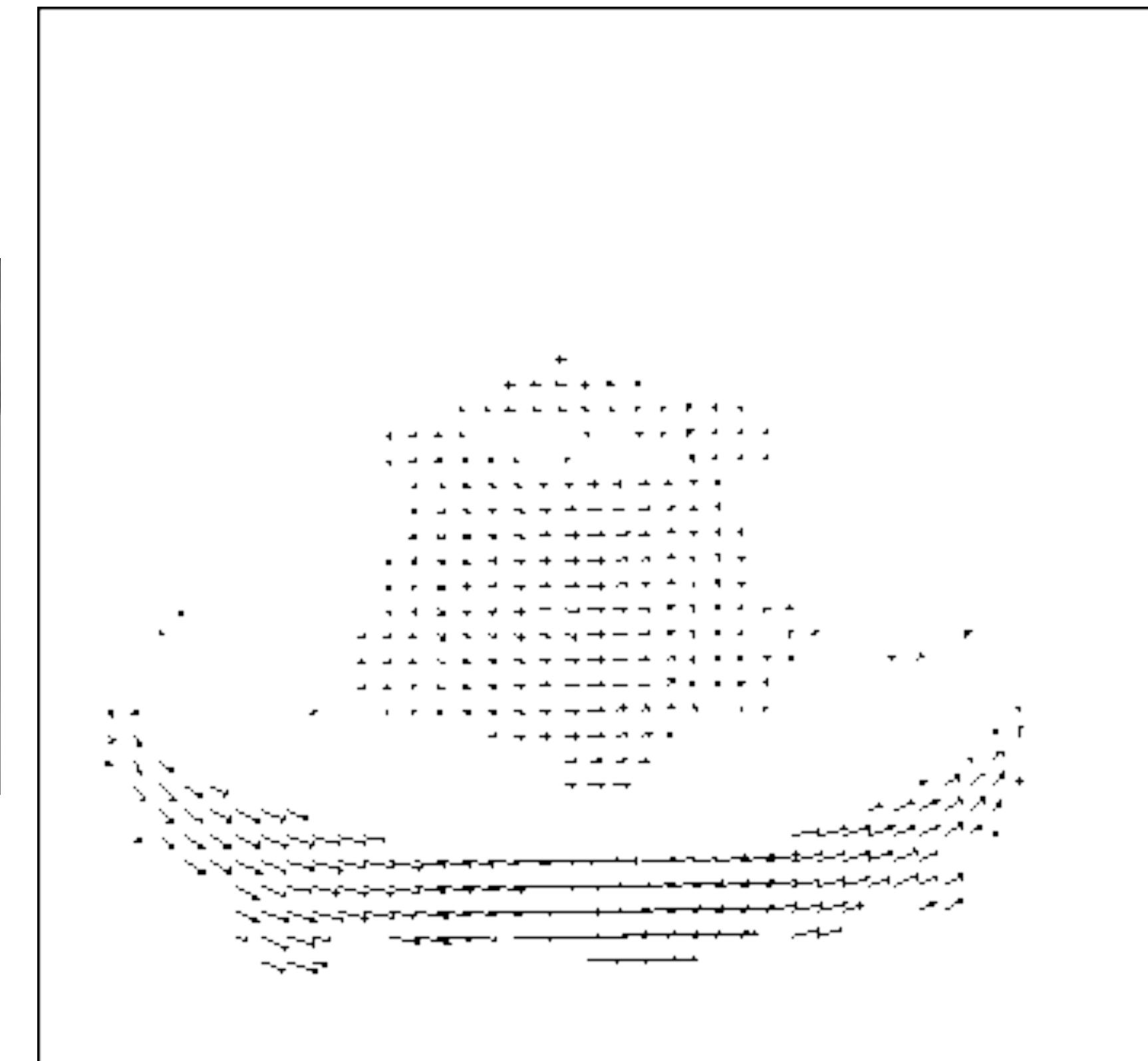
Motion is a powerful perceptual cue



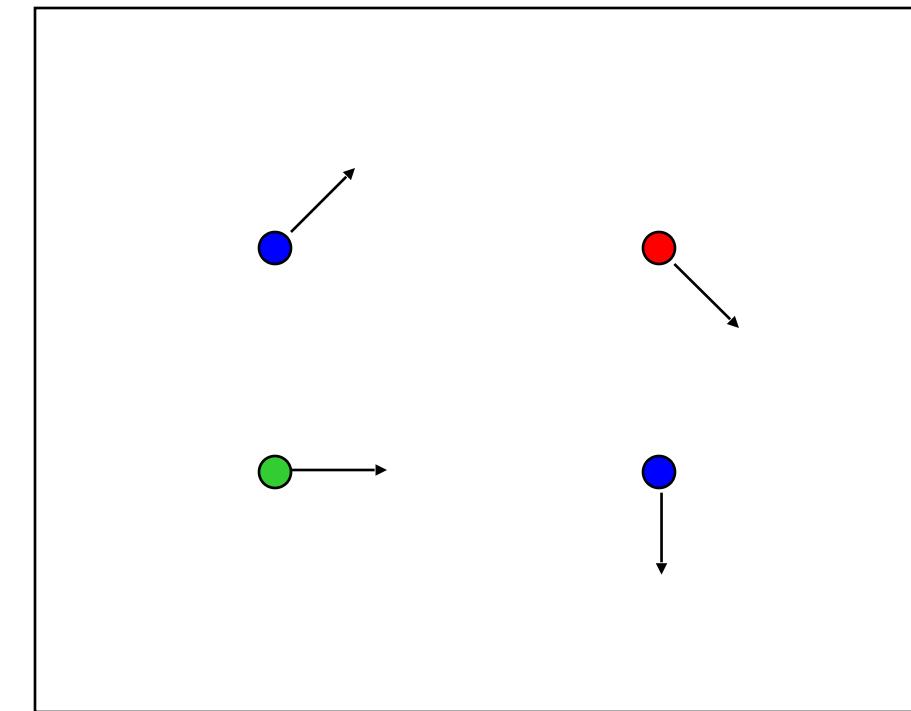
G. Johansson, "Visual Perception of Biological Motion and a Model For Its Analysis",
Perception and Psychophysics 14, 201-211, 1973.
6

Source: S. Lazebnik

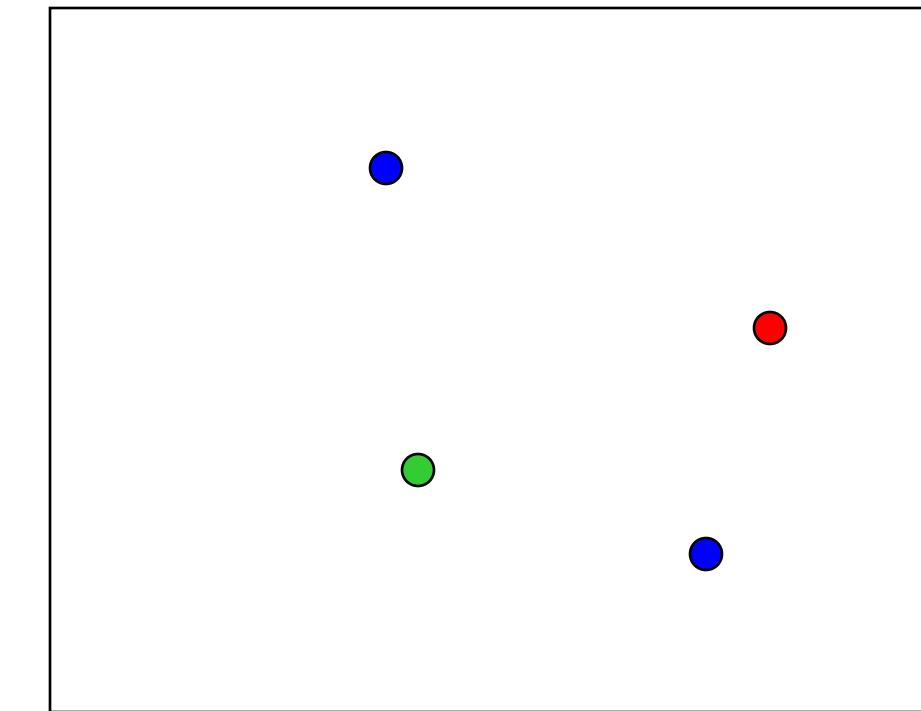
Motion field



Estimating optical flow



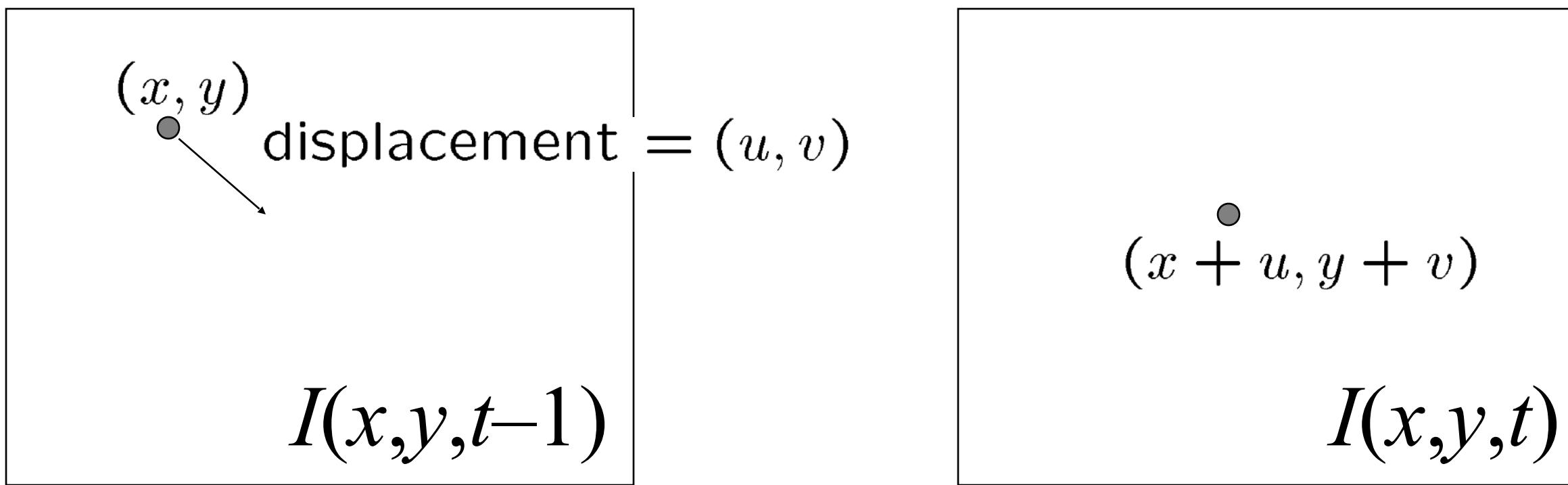
$I(x,y,t-1)$



$I(x,y,t)$

- Given two subsequent frames, estimate the motion field $u(x,y)$ and $v(x,y)$ between them
- Common assumptions
 - **Brightness constancy:** projection of the same point looks the same in every frame
 - **Small motion:** points do not move very far
 - **Spatial coherence:** points move like their neighbors

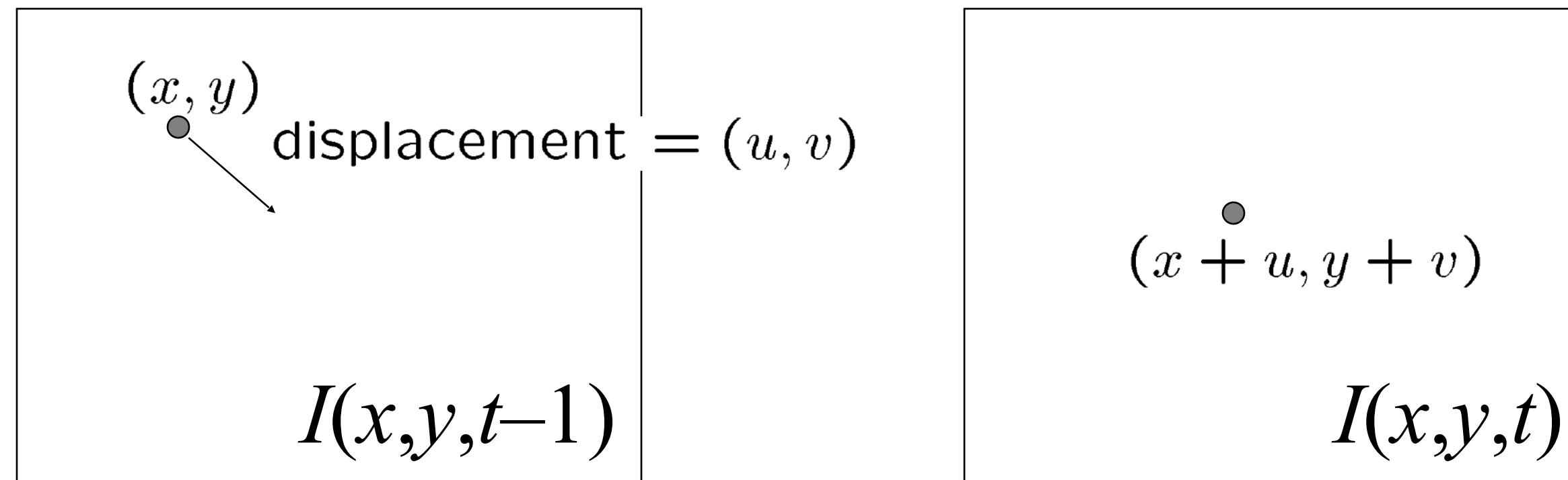
The brightness constancy constraint



Simple loss function [Lucas & Kanade 1981]. Find flow that minimizes:

$$\mathcal{L}(u, v) = \sum_{x, y} [I(x, y, t - 1) - I(x + u(x, y), y + v(x, y), t)]^2$$

The brightness constancy constraint



Brightness Constancy Equation:

$$I(x, y, t - 1) = I(x + u(x, y), y + v(x, y), t)$$

Linearizing the right side using Taylor expansion:

Derivative in y direction

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

Hence, $I_x u + I_y v + I_t \approx 0$

The brightness constancy constraint

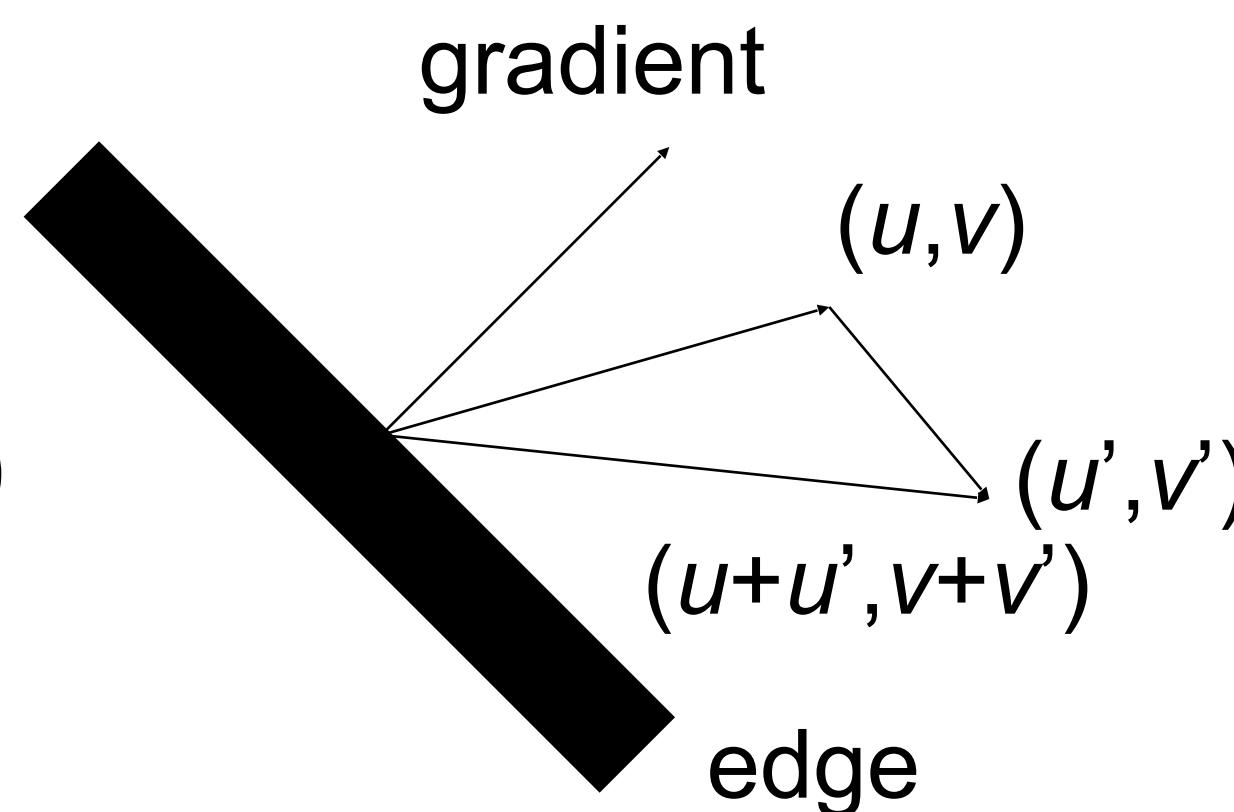
$$I_x u + I_y v + I_t = 0$$

- How many equations and unknowns per pixel?
 - One equation, two unknowns
- What does this constraint mean?

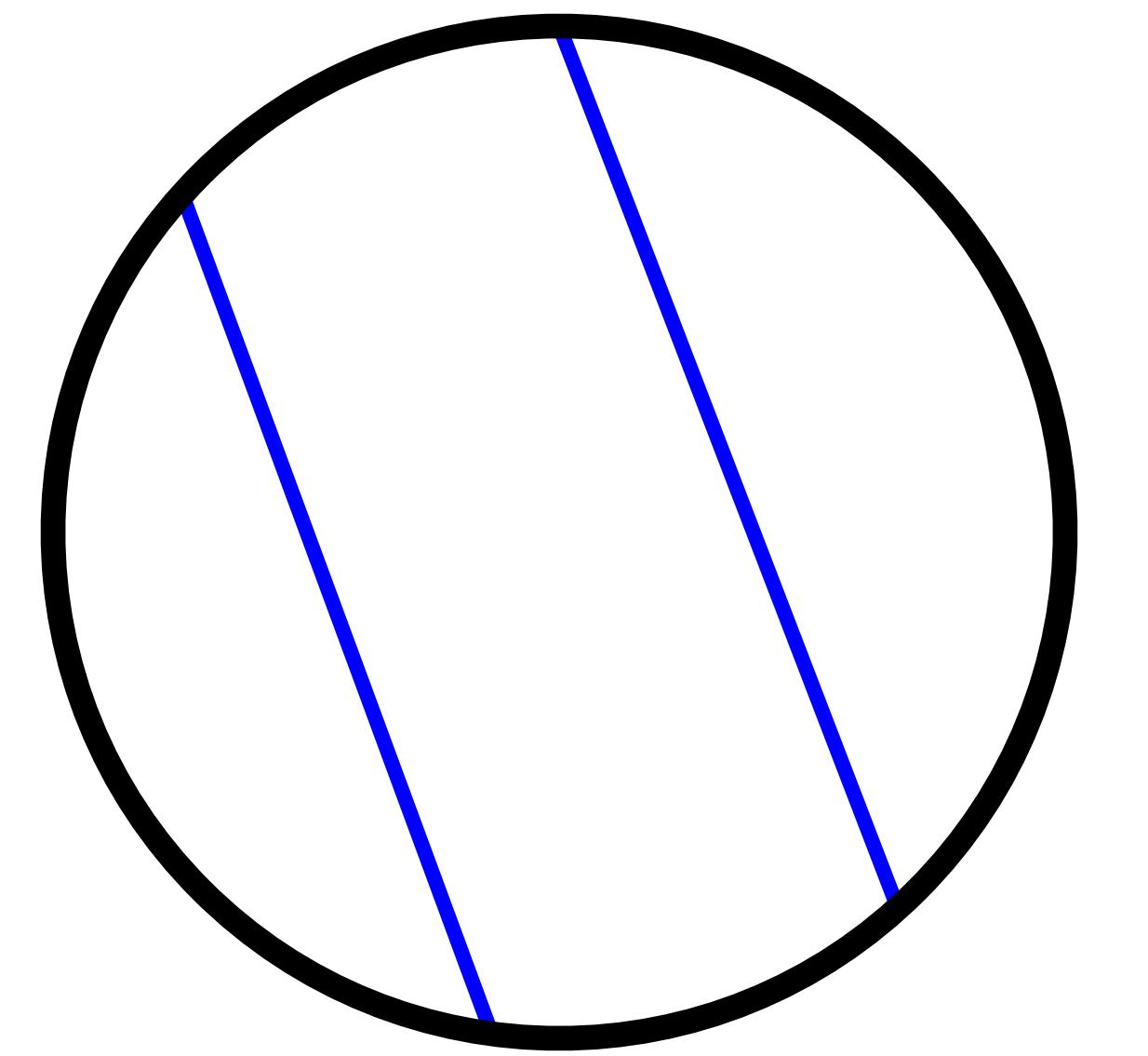
$$\nabla I \cdot (u, v) + I_t = 0$$

- The component of the flow perpendicular to the gradient (i.e., parallel to the edge) is unknown!

If (u, v) satisfies the equation,
so does $(u+u', v+v')$ if $\nabla I \cdot (u', v') = 0$

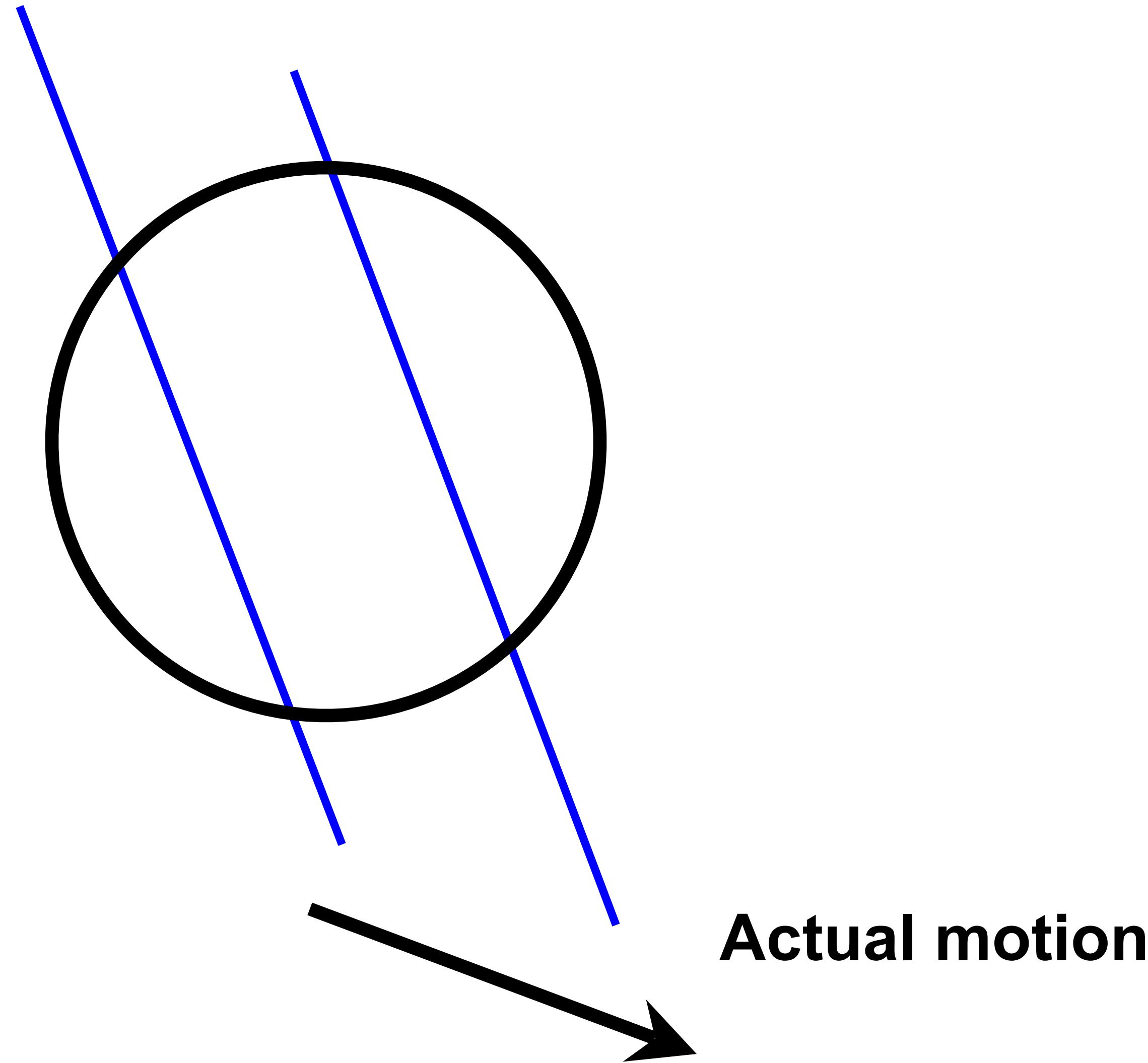


The aperture problem



Perceived motion

The aperture problem



The barber pole illusion



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

The barber pole illusion



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

Solving the aperture problem

- How to get more equations for a pixel?
- **Spatial coherence constraint:** assume the pixel's neighbors have the same (u, v)
 - E.g., if we use a 5×5 window, that gives us 25 equations per pixel

$$\nabla I(\mathbf{x}_i) \cdot [u, v] + I_t(\mathbf{x}_i) = 0$$

$$\begin{bmatrix} I_x(\mathbf{x}_1) & I_y(\mathbf{x}_1) \\ I_x(\mathbf{x}_2) & I_y(\mathbf{x}_2) \\ \vdots & \vdots \\ I_x(\mathbf{x}_n) & I_y(\mathbf{x}_n) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(\mathbf{x}_1) \\ I_t(\mathbf{x}_2) \\ \vdots \\ I_t(\mathbf{x}_n) \end{bmatrix}$$

Lucas-Kanade flow

Least squares problem:

$$\begin{bmatrix} I_x(\mathbf{x}_1) & I_y(\mathbf{x}_1) \\ I_x(\mathbf{x}_2) & I_y(\mathbf{x}_2) \\ \vdots & \vdots \\ I_x(\mathbf{x}_n) & I_y(\mathbf{x}_n) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(\mathbf{x}_1) \\ I_t(\mathbf{x}_2) \\ \vdots \\ I_t(\mathbf{x}_n) \end{bmatrix}$$

When is this system solvable?

Lucas-Kanade optical flow

$$\begin{bmatrix} I_x(\mathbf{x}_1) & I_y(\mathbf{x}_1) \\ I_x(\mathbf{x}_2) & I_y(\mathbf{x}_2) \\ \vdots & \vdots \\ I_x(\mathbf{x}_n) & I_y(\mathbf{x}_n) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(\mathbf{x}_1) \\ I_t(\mathbf{x}_2) \\ \vdots \\ I_t(\mathbf{x}_n) \end{bmatrix}$$

$\mathbf{A} \quad \mathbf{d} = \mathbf{b}$
 $n \times 2 \quad 2 \times 1 \quad n \times 1$

- Solution given by $(\mathbf{A}^T \mathbf{A})\mathbf{d} = \mathbf{A}^T \mathbf{b}$

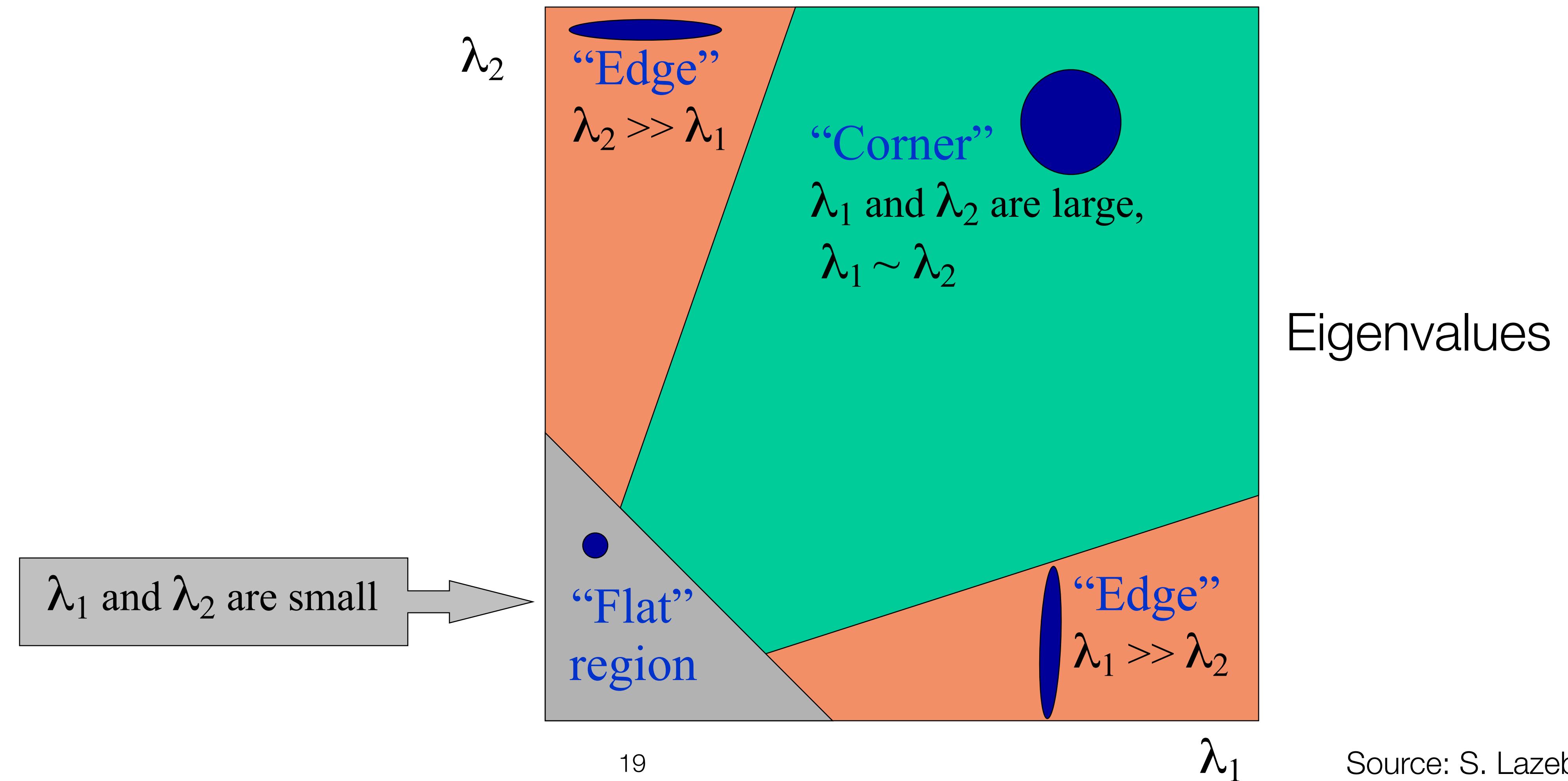
$$\begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} \sum I_x I_t \\ \sum I_y I_t \end{bmatrix}$$

(summations are over all
pixels in the window)

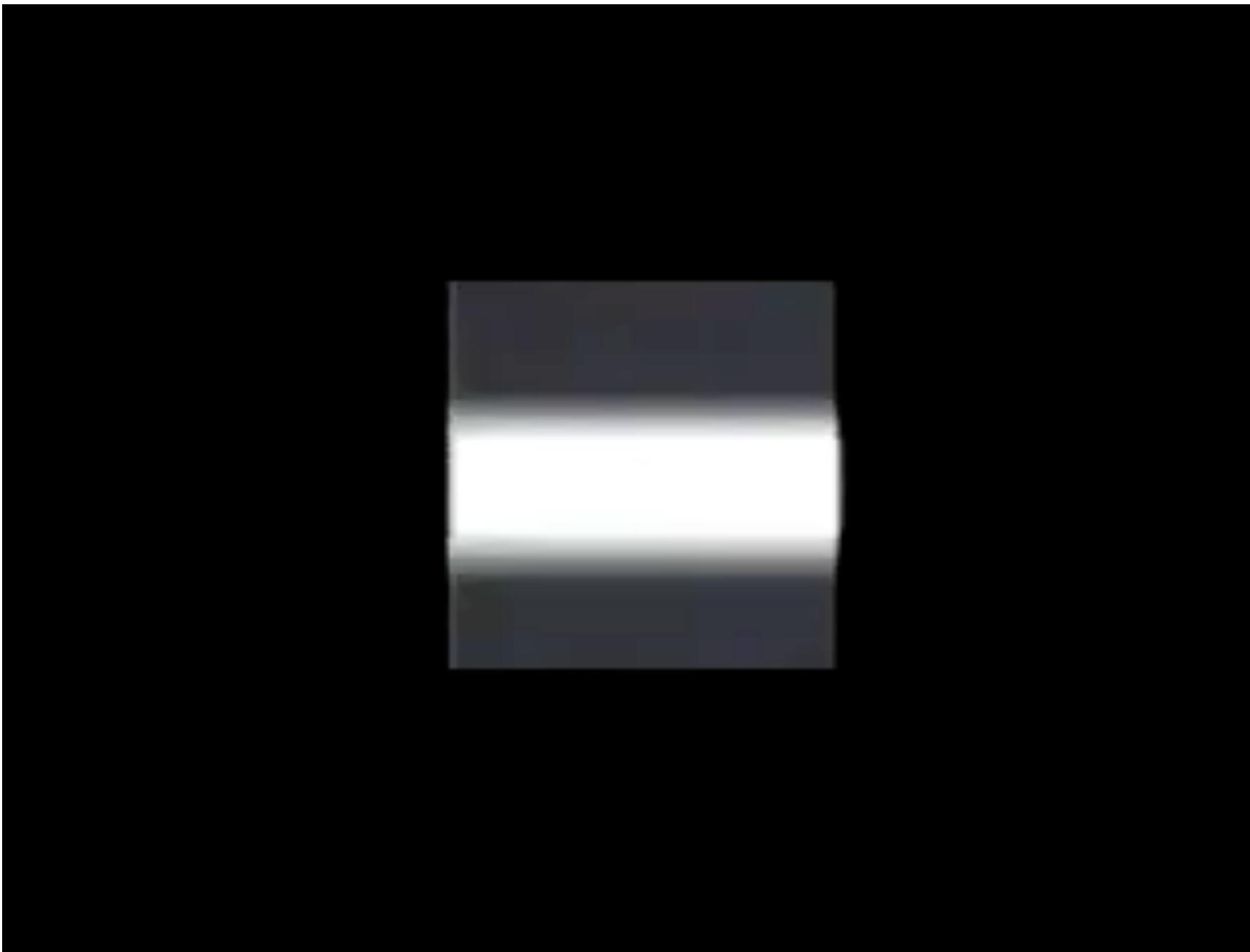
$\mathbf{M} = \mathbf{A}^T \mathbf{A}$ is the “second
moment” matrix

Analyzing the second moment matrix

- Estimation of optical flow is well-conditioned precisely for regions with high “cornerness”:



Conditions for solvability



Conditions for solvability

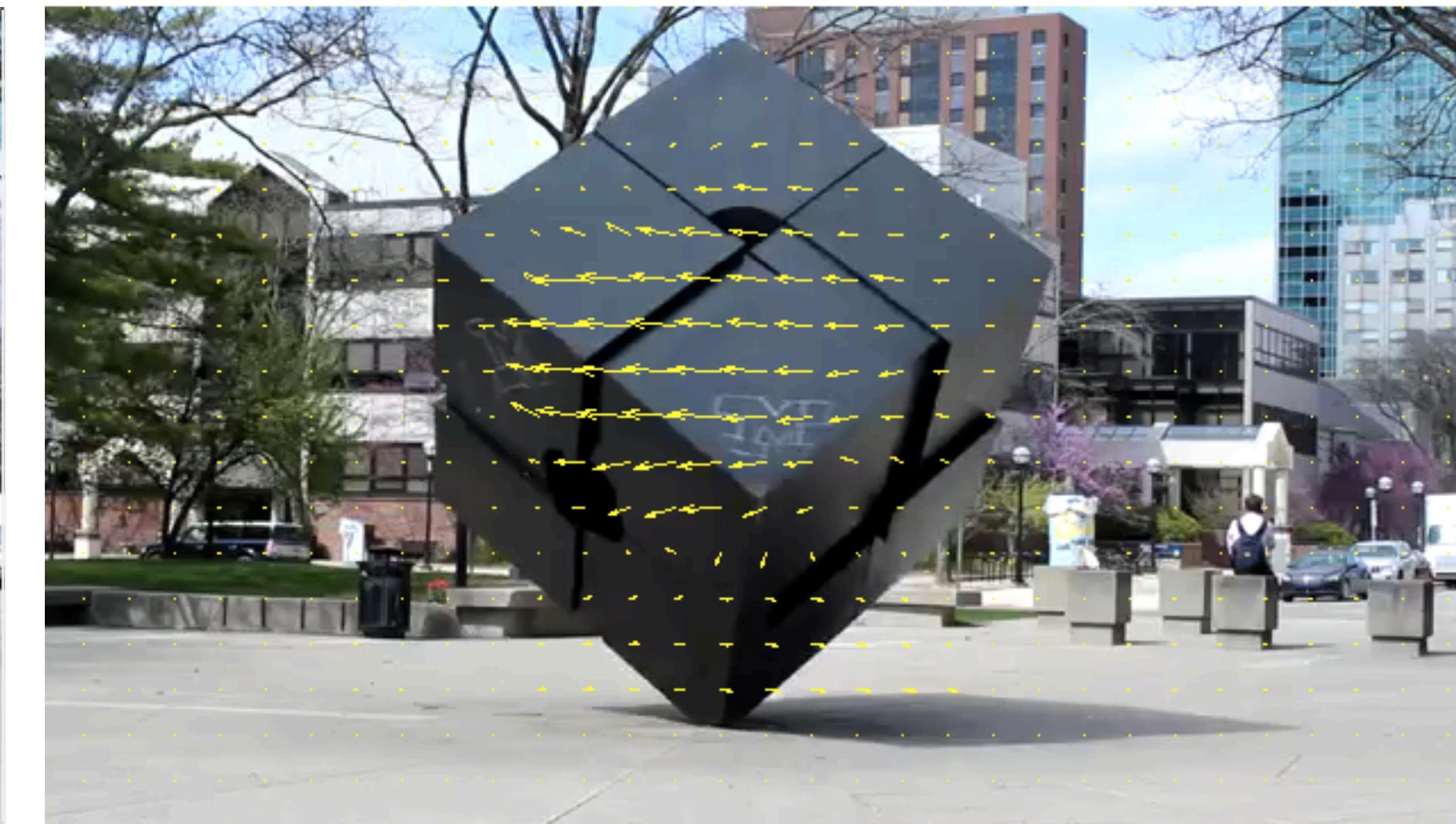


Lucas-Kanade flow example

Input frames



Output



Fixing the errors in Lucas-Kanade

- The motion is large (larger than a pixel)
 - Iterative refinement
 - Multi-resolution (coarse-to-fine) estimation
- Local ambiguity
 - Smooth using graphical model refinement

Large motions



Idea #1: iterative estimation

Goal: minimize matching error

$$\mathcal{L}(u, v) = \sum_{x,y} \sum_{x'=x-N}^{x+N} \sum_{y'=y-N}^{y+N} [I(x', y', t-1) - I(x' + u(x, y), y' + v(x, y), t)]^2$$

where the window width/height is $2N+1$

Iterative algorithm:

Initialize flow: $u_0(x, y) = v_0(x, y) = 0$

For each iteration i :

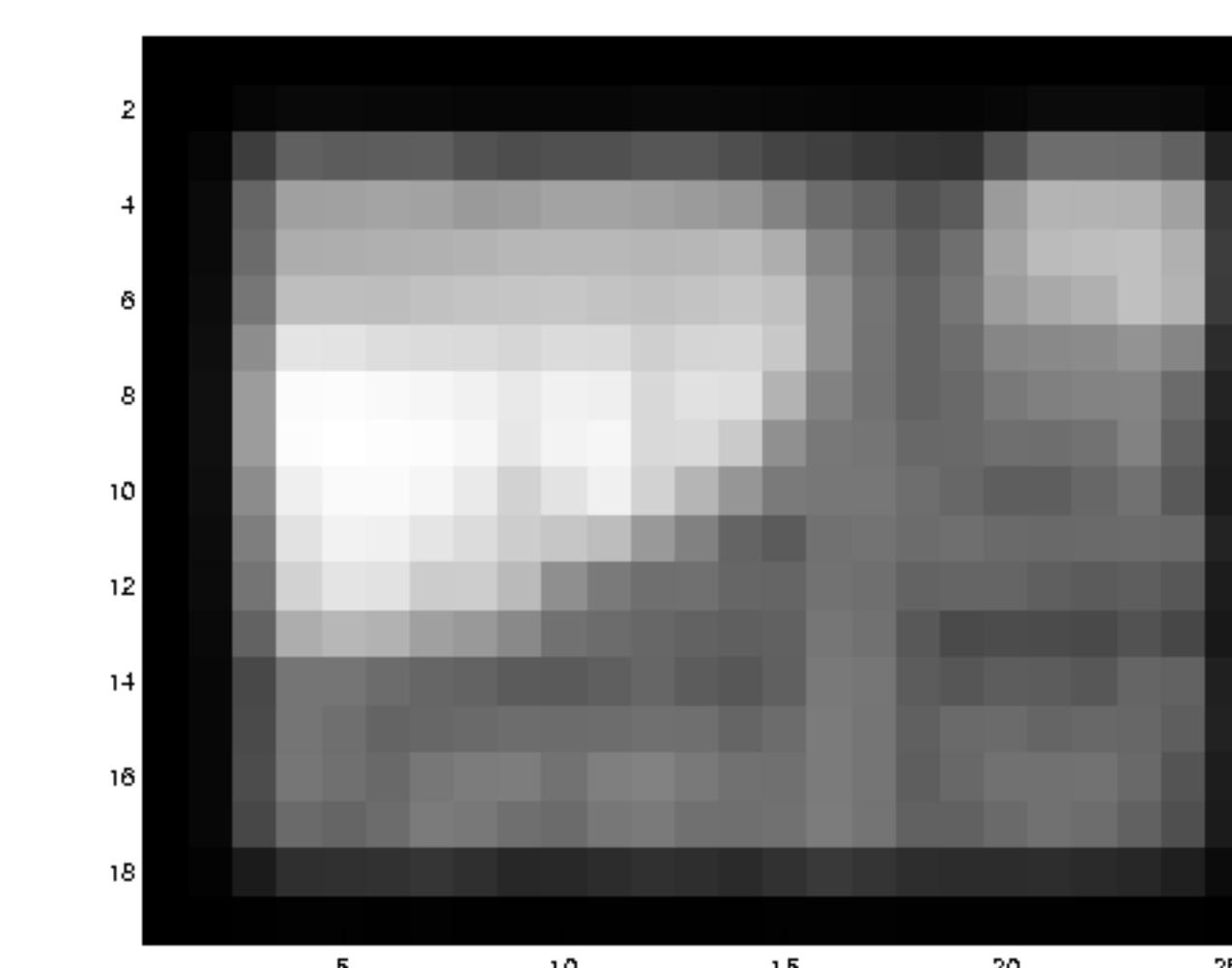
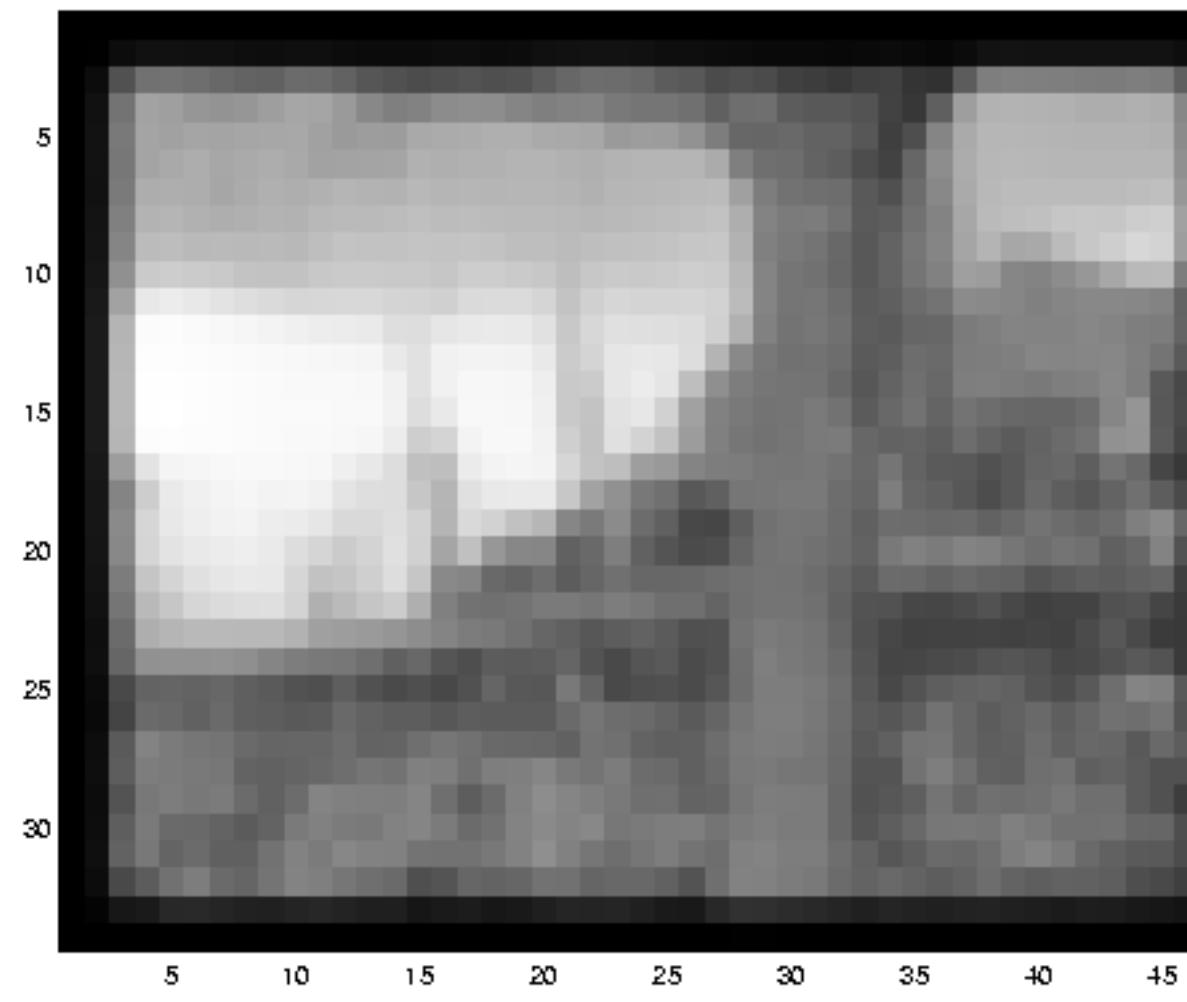
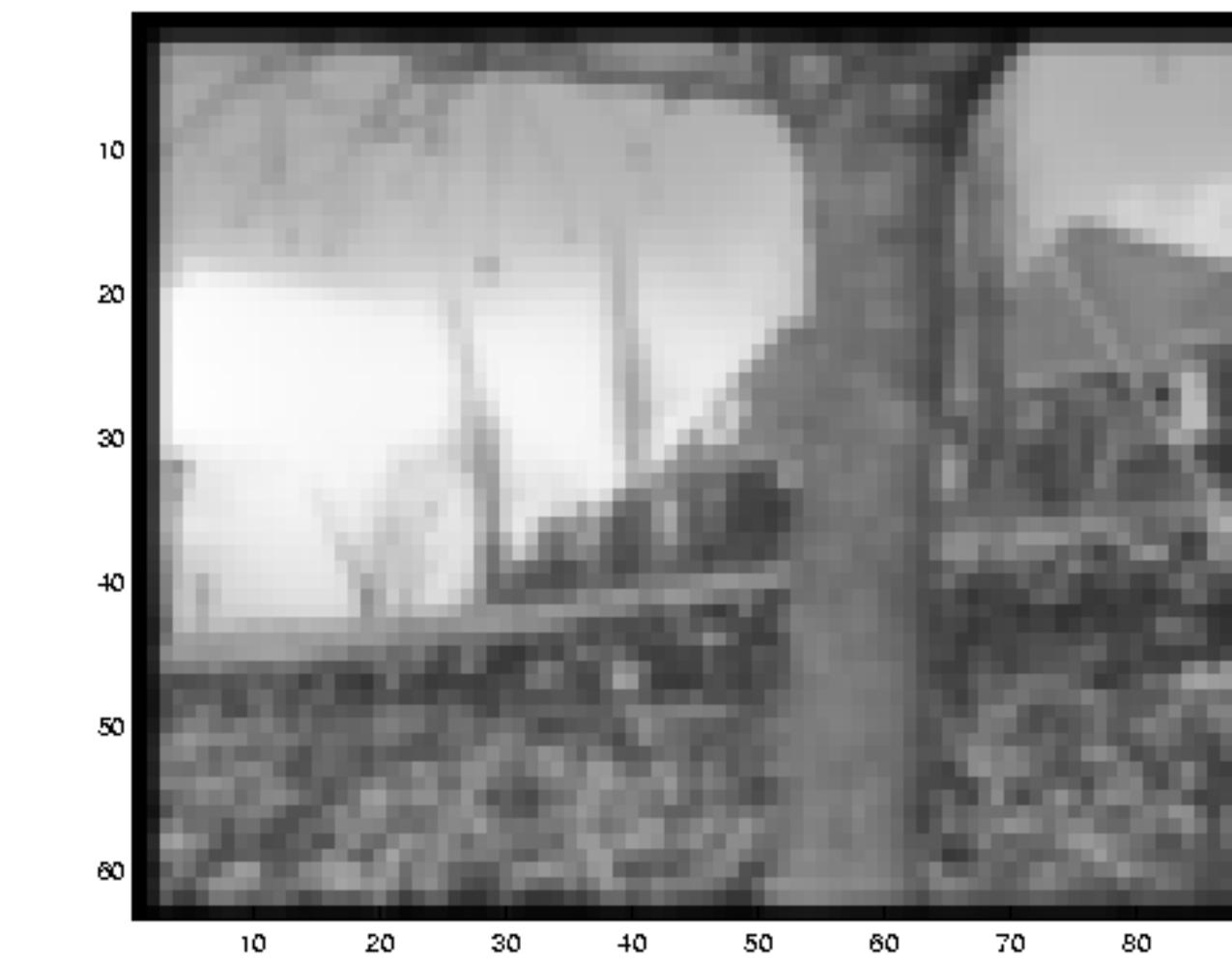
1. Linearize around current solution. Find an update for problem:

$$\operatorname{argmin}_{\Delta u, \Delta v} \mathcal{L}(u_i + \Delta u, v_i + \Delta v)$$

by solving linear least squares problem for each pixel: $Ad = b$

2. Apply updates: $u_{i+1} = u_i + \Delta u$ and $v_{i+1} = v_i + \Delta v$

Idea #2: Multi-scale estimation



Idea #2: Multi-scale estimation

Initialize flow: $u_0(x, y) = v_0(x, y) = 0$

For each scale s :

 Initialize flow from previous (coarser) scale

 For each iteration i :

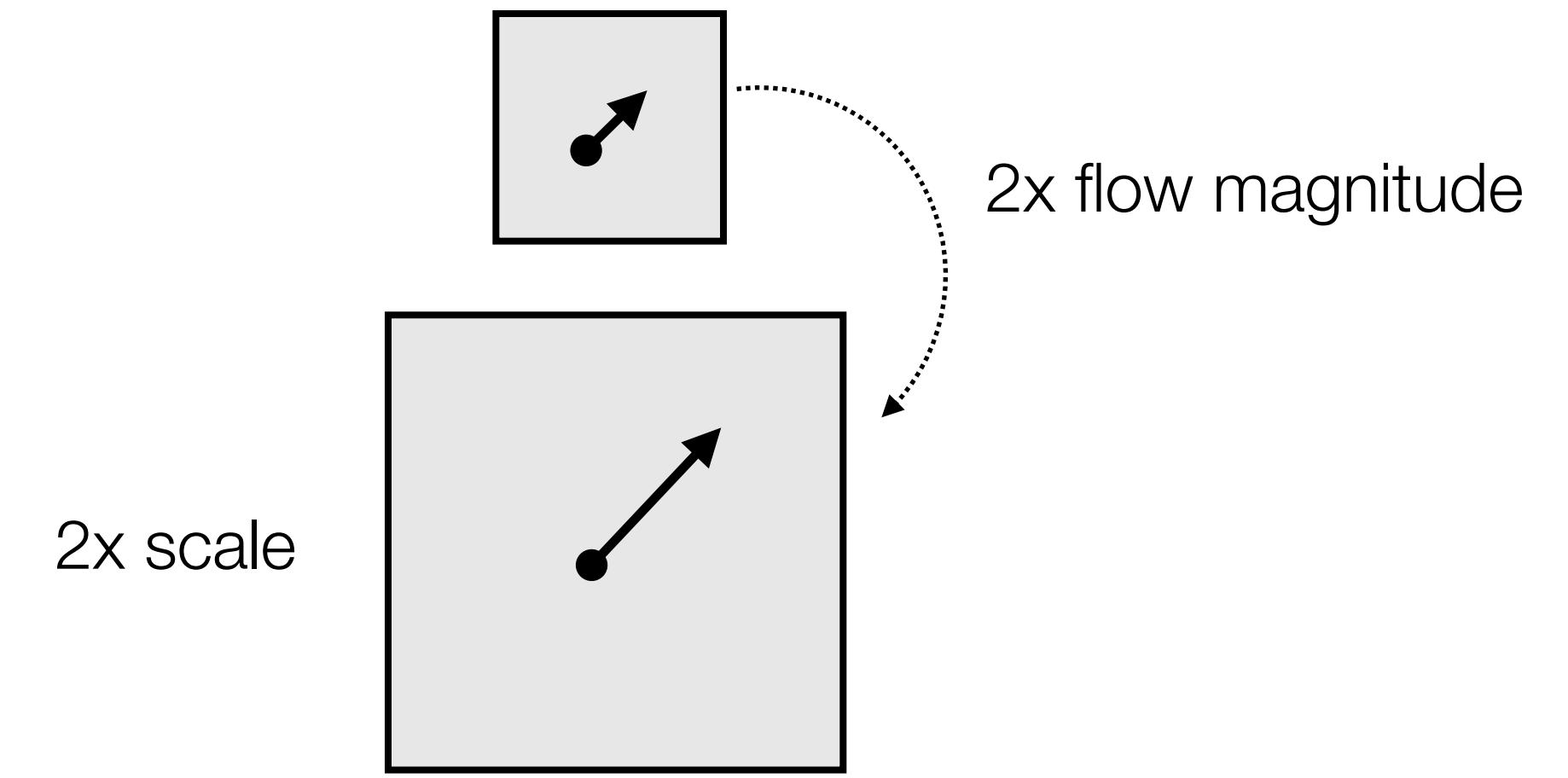
1. Linearize around current solution. Find an update for problem:

$$\operatorname{argmin}_{\Delta u, \Delta v} \mathcal{L}(u_i + \Delta u, v_i + \Delta v)$$

 by solving linear least squares problem for each pixel: $Ad = b$

2. Apply updates: $u_{i+1} = u_i + \Delta u$ and $v_{i+1} = v_i + \Delta v$

3. Extra trick for smoother flow: apply median filter to u_{i+1} and v_{i+1}



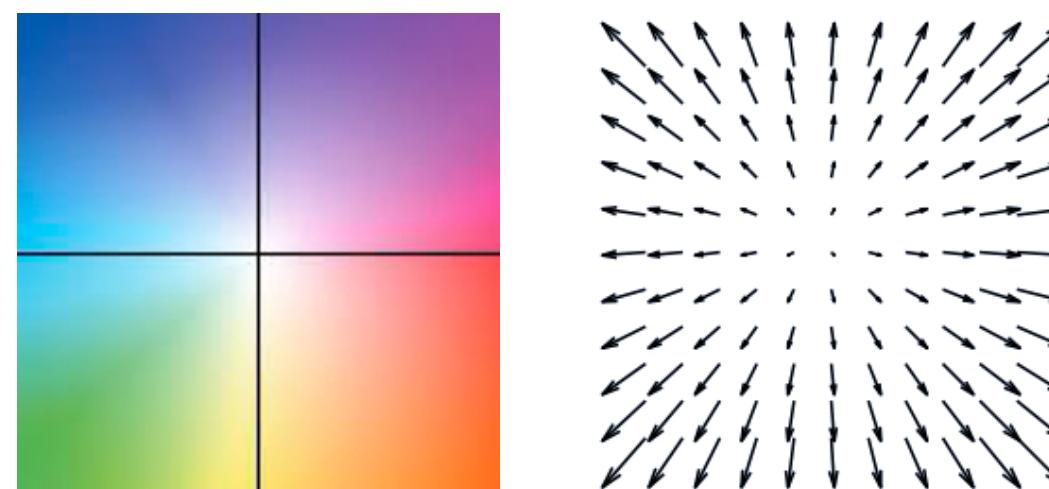
Example



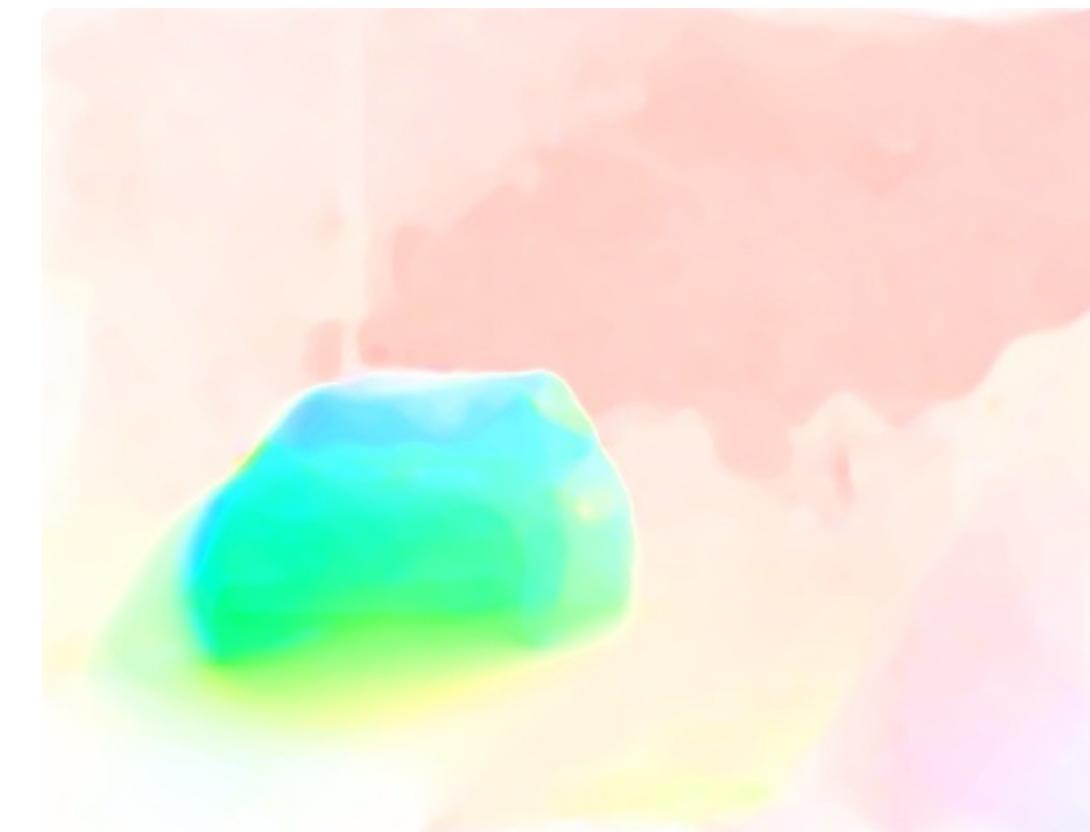
Input two frames



Coarse-to-fine LK



Flow visualization



Coarse-to-fine LK with median filtering

Smoothness assumption

Goal: minimize matching error + smoothness [Horn and Schunck 1981]

$$\sum_{x,y} [I(x, y, t-1) - I(u(x), v(y), t)]^2 + \sum_p \sum_{p' \in \mathcal{N}} (u(p) - u(p'))^2 + (v(p) - v(p'))^2$$

$E_d(u, v)$ match cost $E_s(u, v)$ smoothness

where p and p' are neighboring pixels

- This is a probabilistic graphical model!
- Can solve using least squares (as in Lucas-Kanade)
- Or using discrete optimization (e.g. belief propagation)

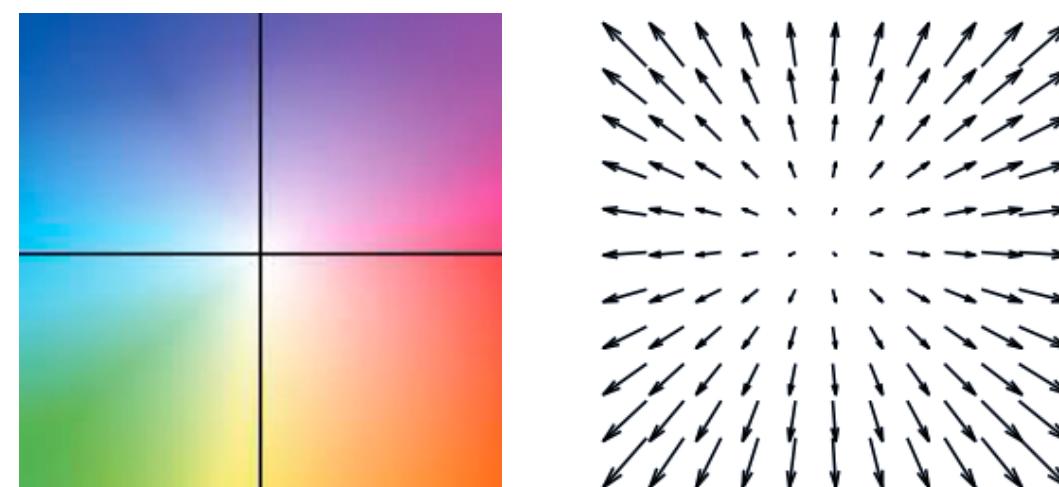
Smoothness assumption



Input two frames



Horn-Schunck



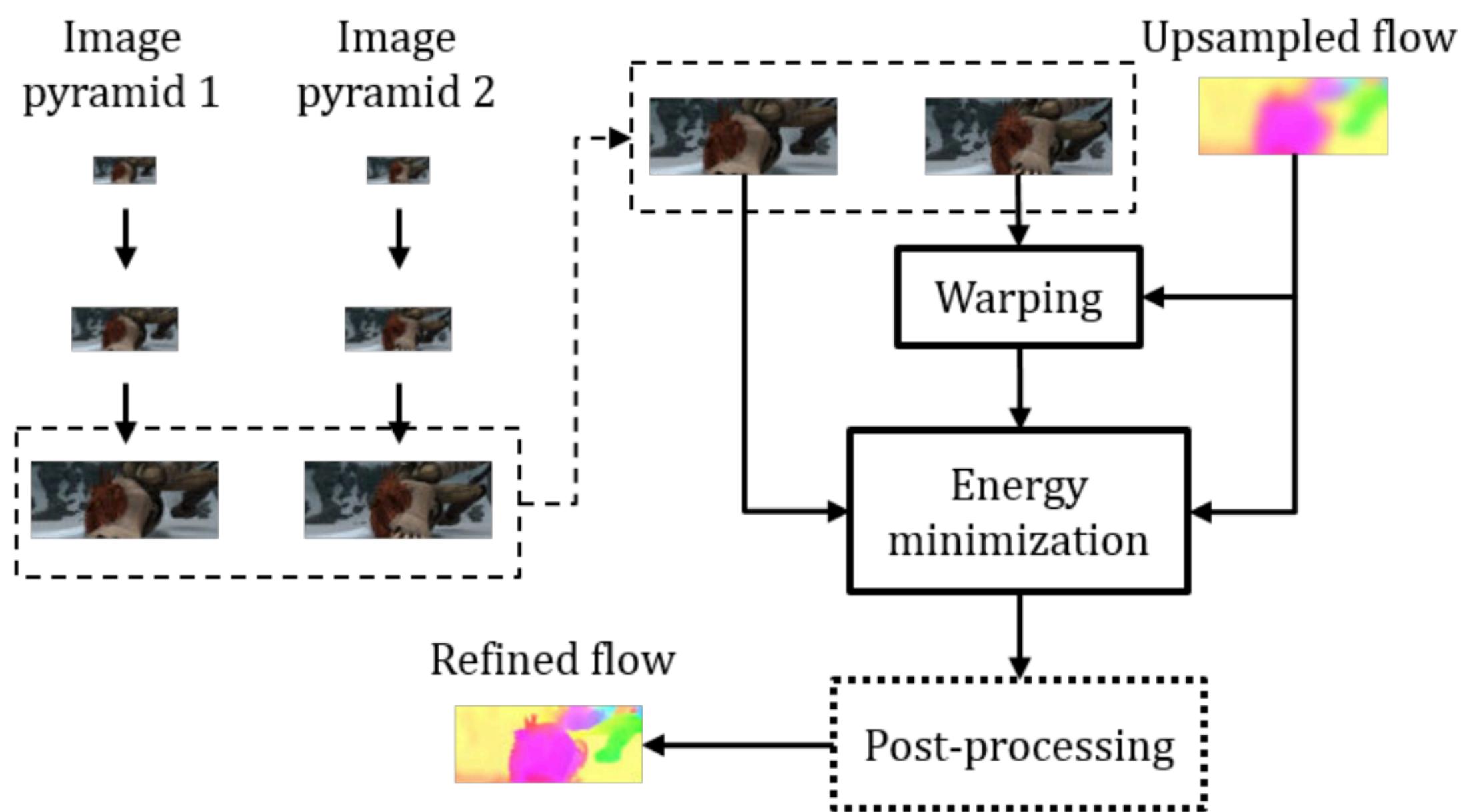
Flow visualization



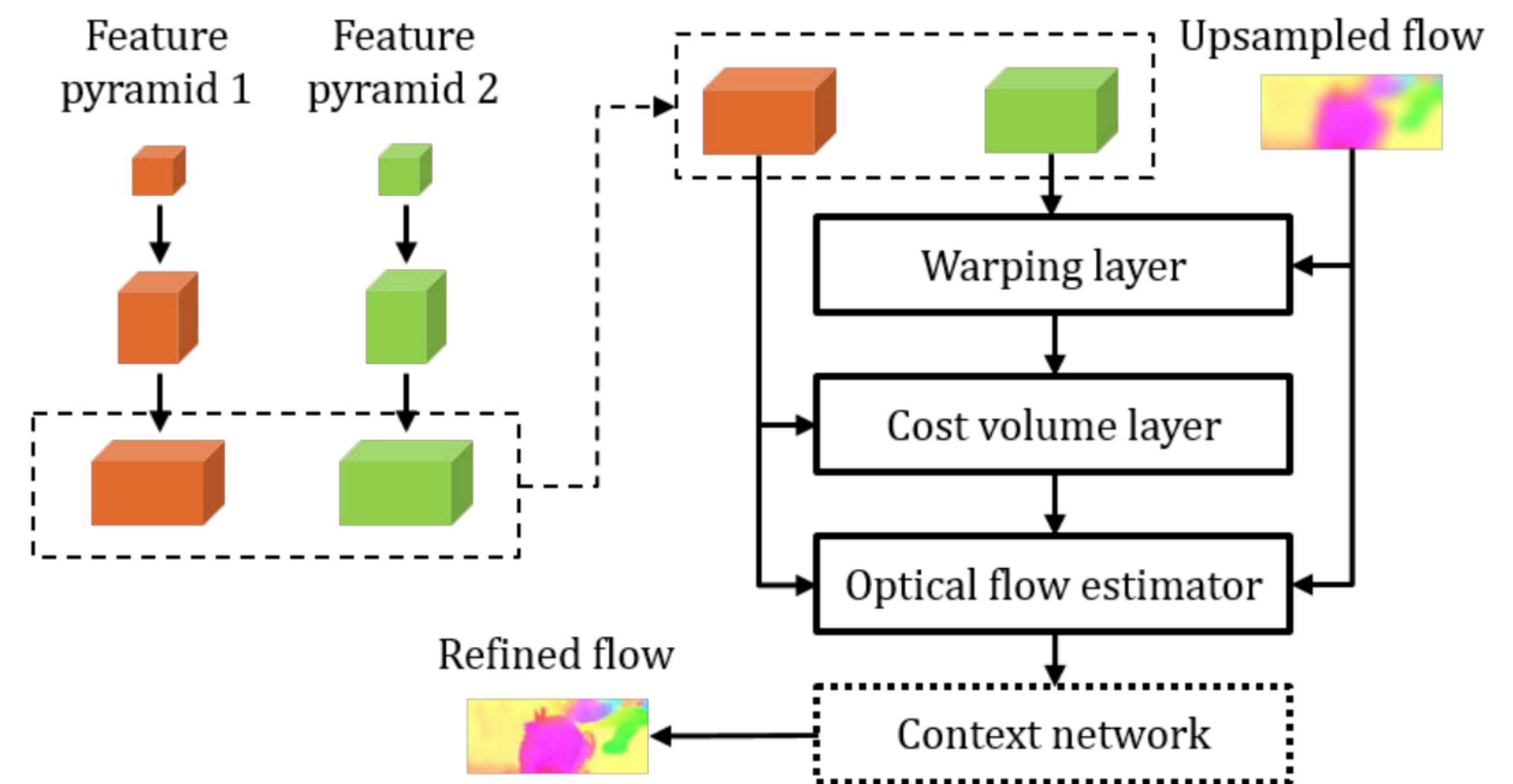
Coarse-to-fine LK



Flow networks

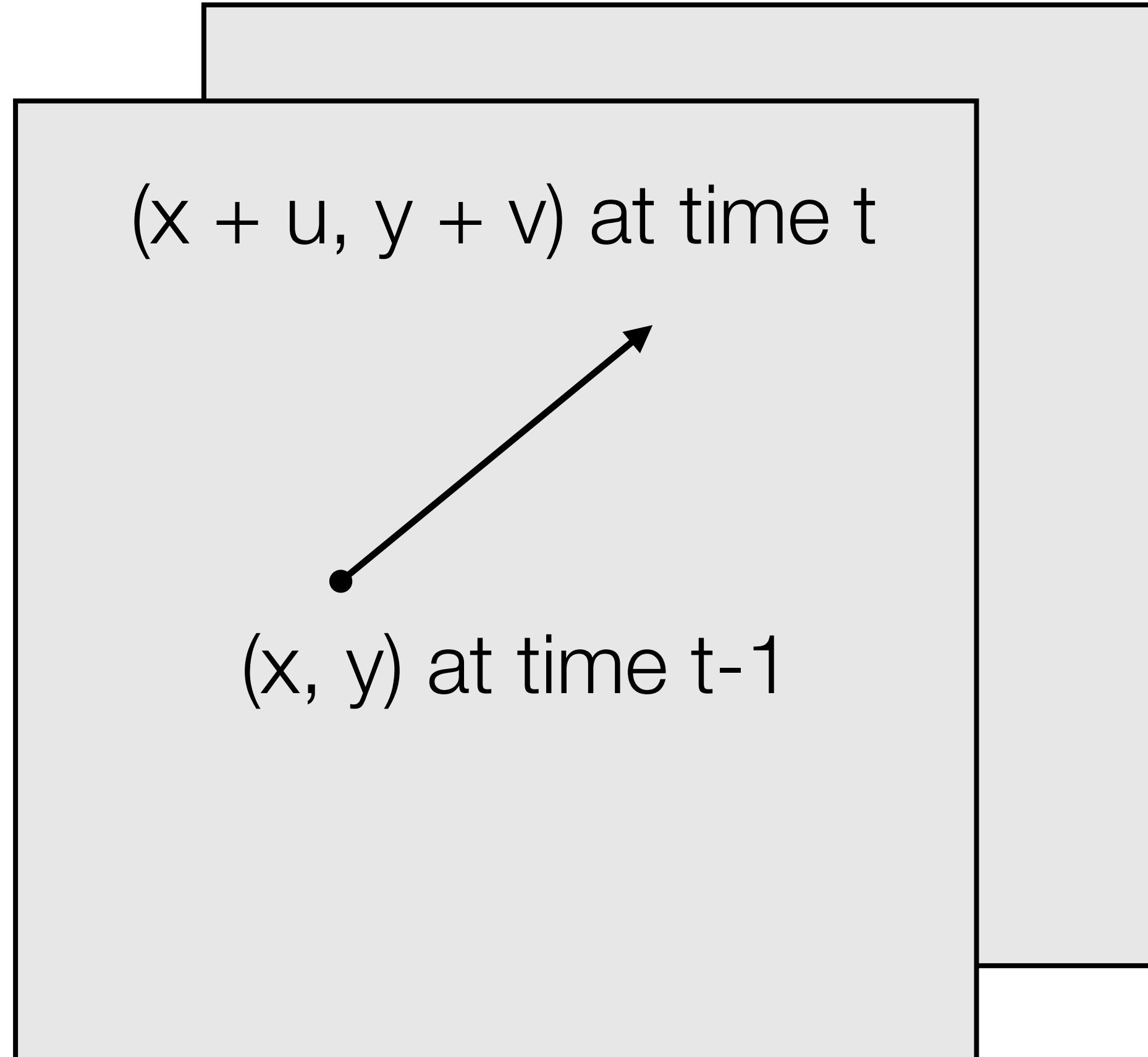


Traditional coarse-to-fine flow



PWC-net

Simple application: slow motion



- Flow used in lots of familiar places!
- E.g., video compression, denoising, action recognition, ...
- One application: use flow to estimate where pixel will be *between* frames
- Synthesize intermediate frames

Super SloMo: High Quality Estimation of Multiple Intermediate Frames for Video Interpolation

Huaizu Jiang¹, Deqing Sun², Varun Jampani²

Ming-Hsuan Yang^{3,2}, Erik Learned-Miller¹, Jan Kautz²

¹UMass Amherst ²NVIDIA ³UC Merced

(No audio commentary)

Next lecture: light and color