Optical flow

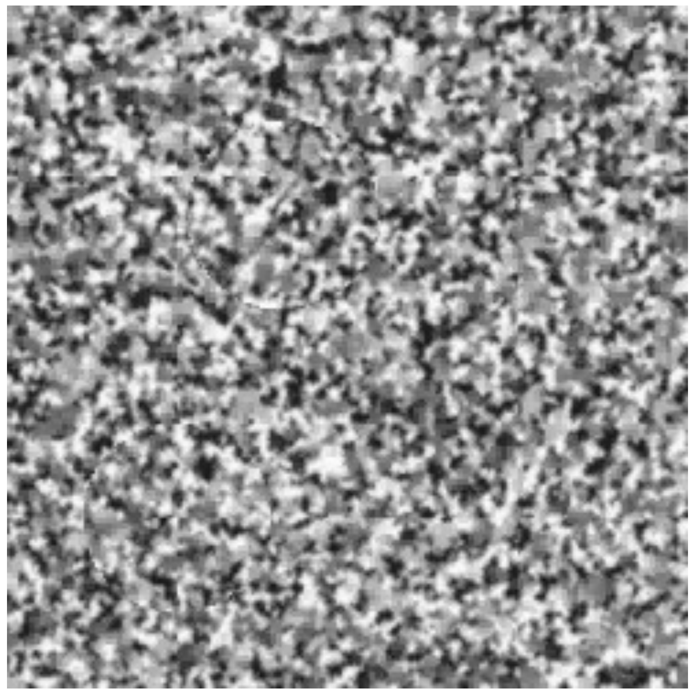
Many slides adapted from S. Seitz, R. Szeliski, M. Pollefeys

Outline

- Motion: motivation
- Brightness constancy constraint, aperture problem
- Estimating optical flow (Lucas-Kanade)
- Beyond basic flow estimation
- Applications

Motion is a powerful perceptual cue

- Sometimes, it is the only cue



Motion is a powerful perceptual cue

- Even “impoverished” motion data can evoke a strong percept

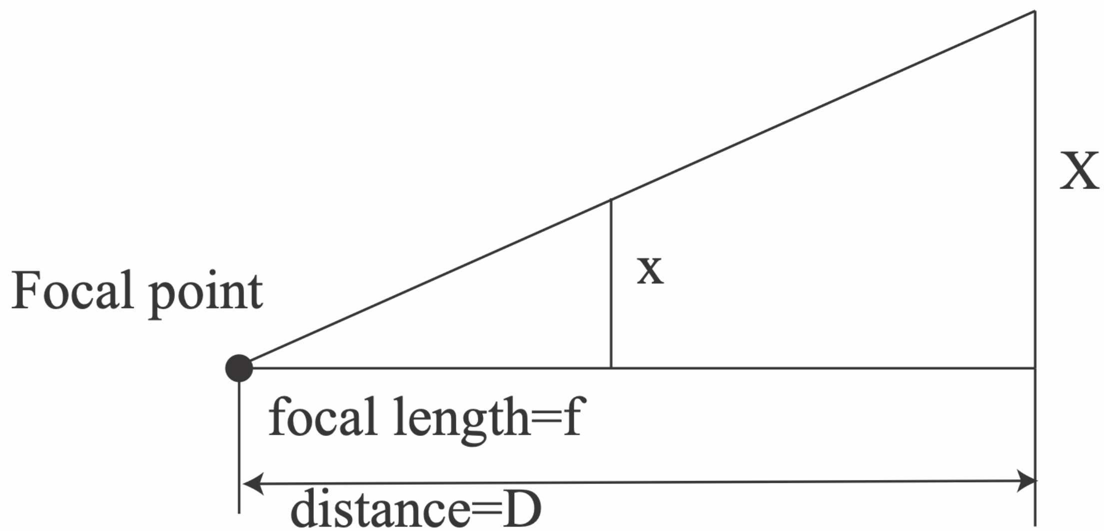
G. Johansson, “Visual Perception of Biological Motion and a Model For Its Analysis",
Perception and Psychophysics 14, 201-211, 1973.

Motion is a powerful perceptual cue

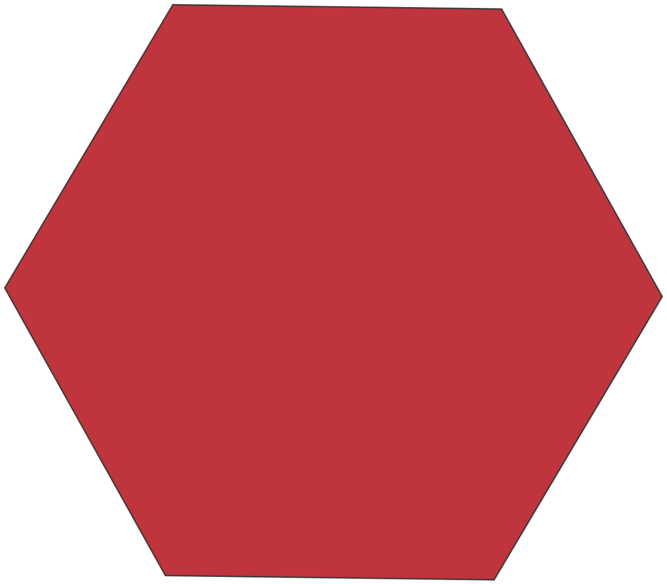
- Even “impoverished” motion data can evoke a strong percept

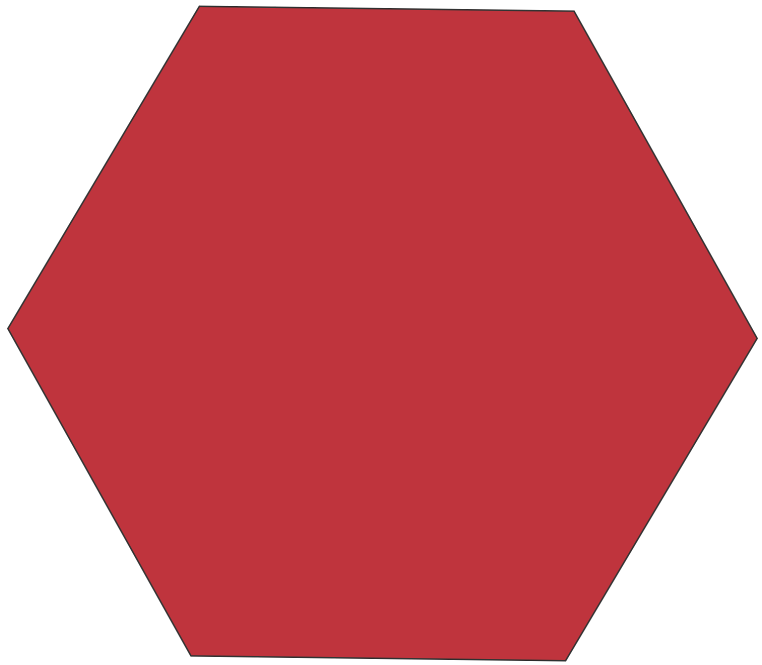
G. Johansson, “Visual Perception of Biological Motion and a Model For Its Analysis",
Perception and Psychophysics 14, 201-211, 1973.

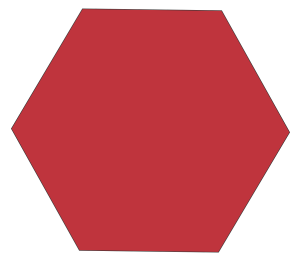
Motion gives time-to-contact cues

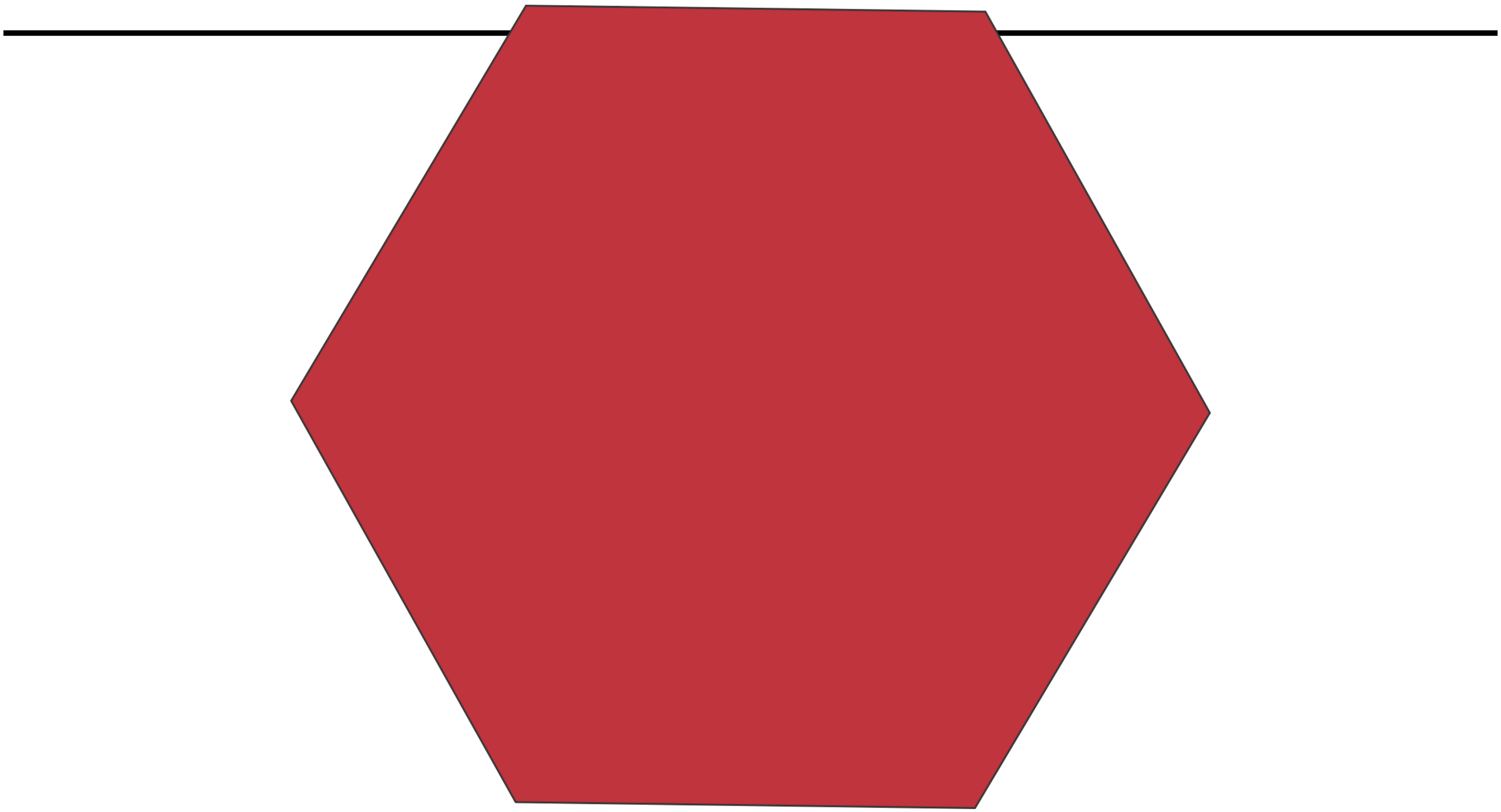


Fun fact: time to contact = $x/(dx/dt)$

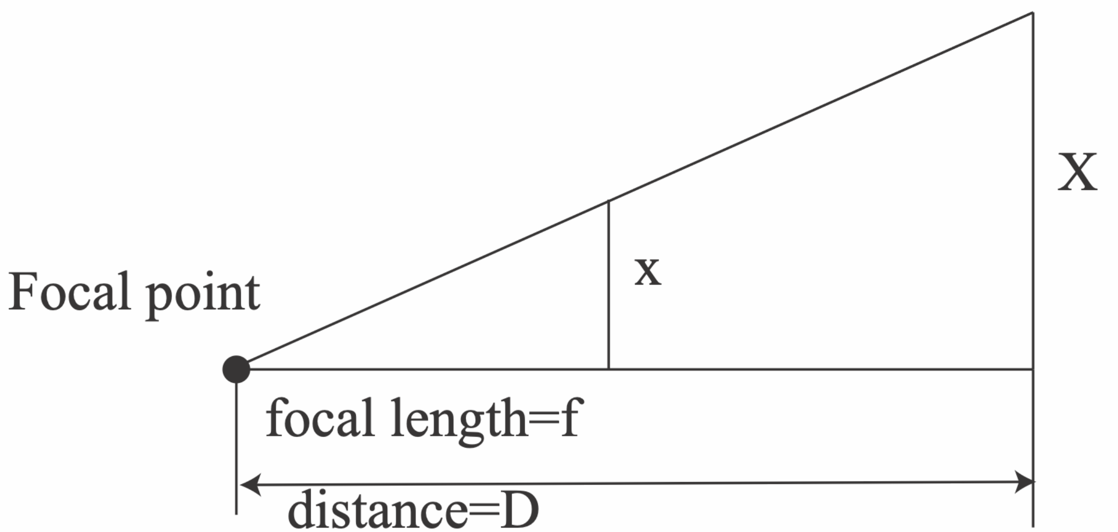








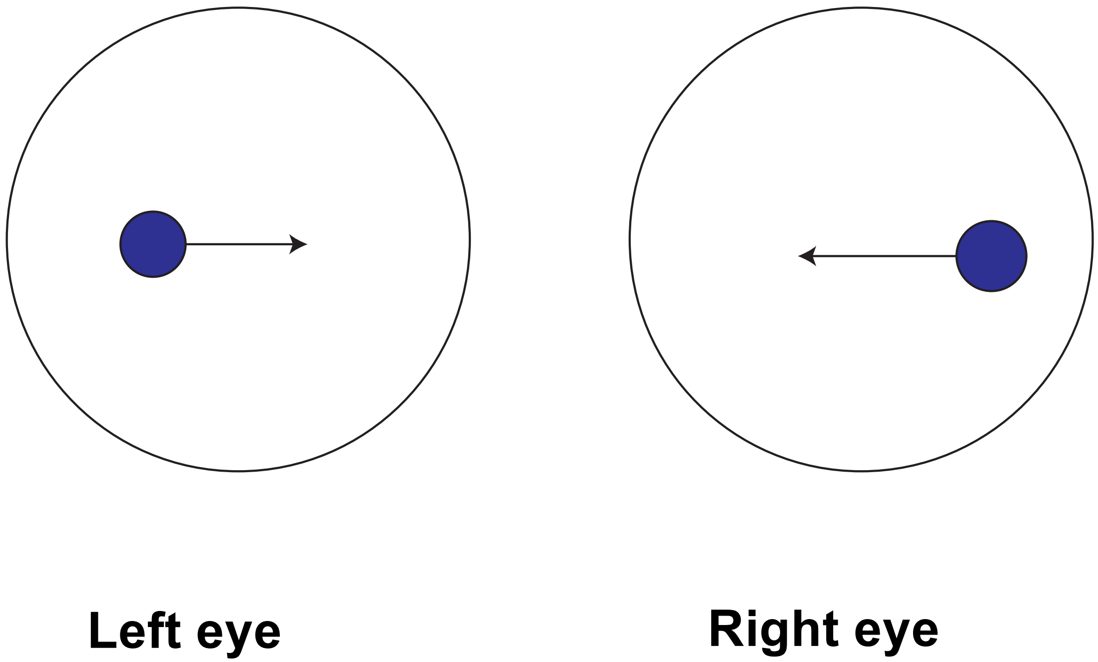
Rich source of reflexes in animals



Fun fact: time to contact = x/(dx/dt)

x/(dx/dt) big = problem soon!

Another amusing motion reflex (two eyes!)



Optical flow

- **Optical flow** is the *apparent motion* of brightness patterns in the image
  - Can be caused by camera motion, object motion, or changes of lighting in the scene

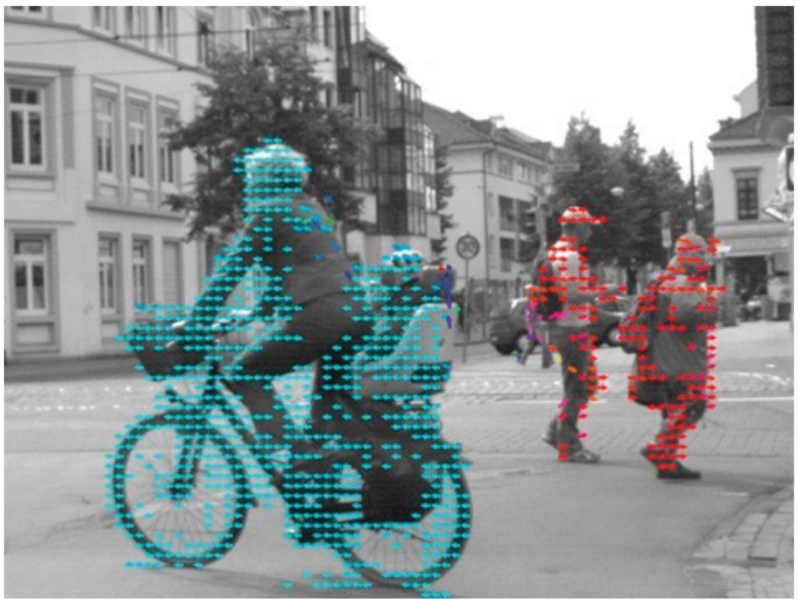


Image source

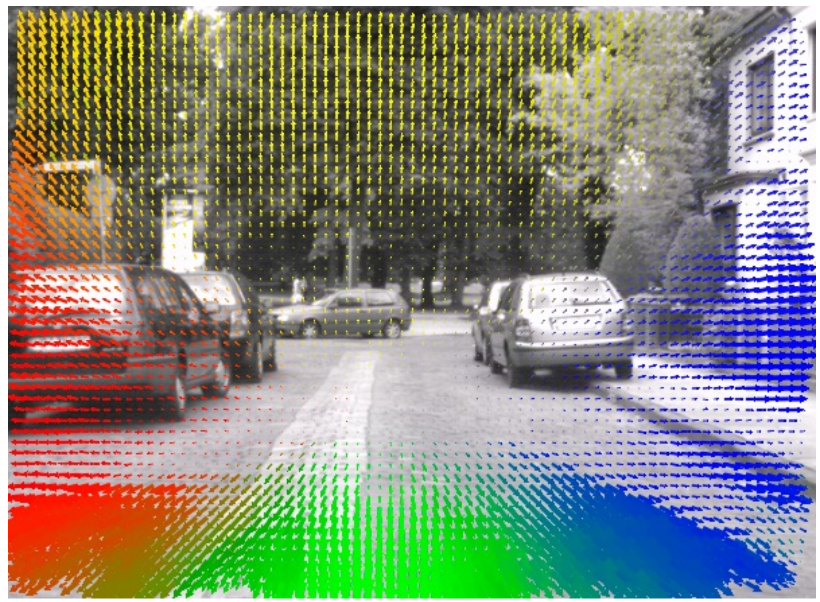
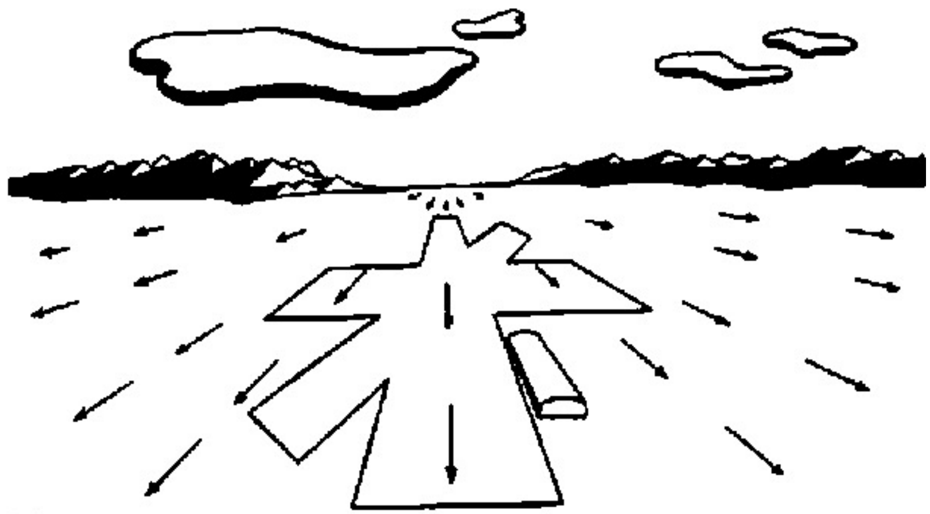
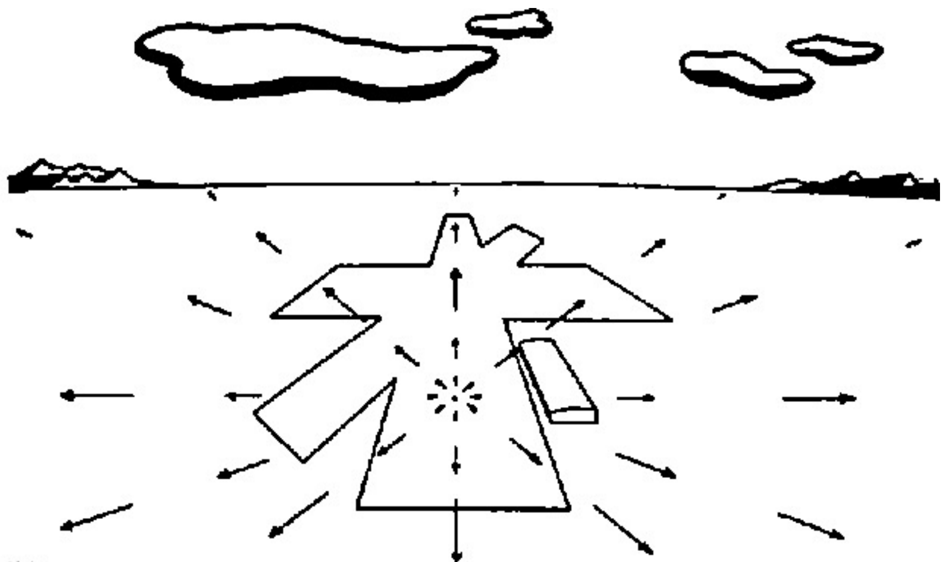


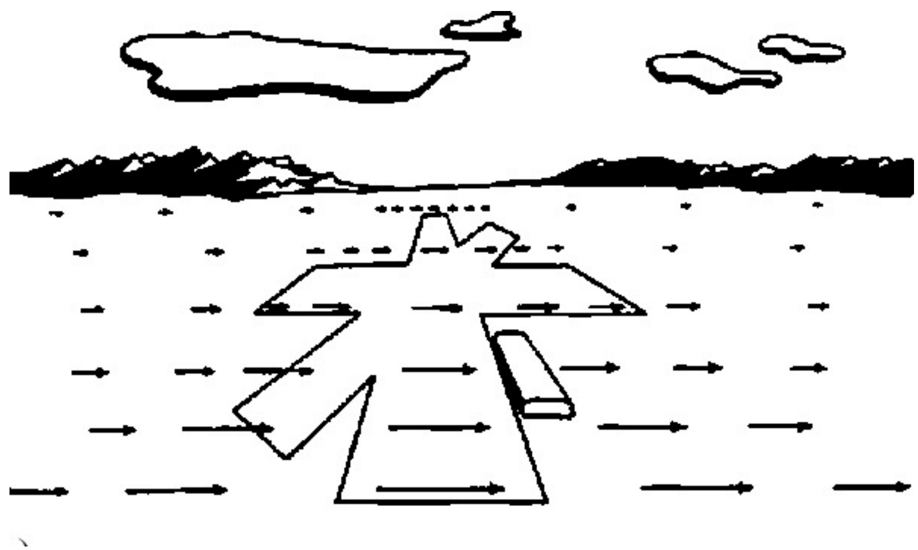
Image source



(a)

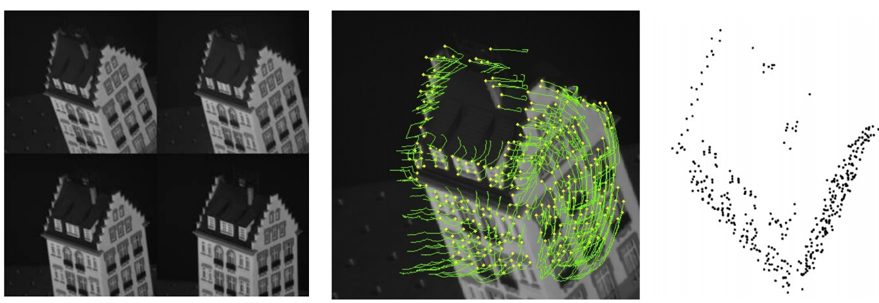


(b)

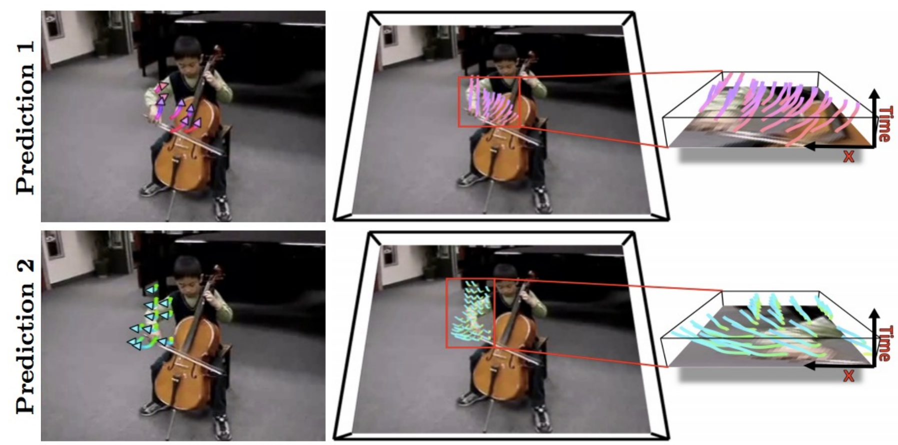


Uses of optical flow in computer vision

- Video analysis and enhancement (stabilization, shot boundary detection, motion magnification, etc.)
- Object tracking and segmentation in videos
- Structure from motion, stereo matching
- Event and activity recognition
- Self-supervised and predictive learning



Source: Tomasi & Kanade

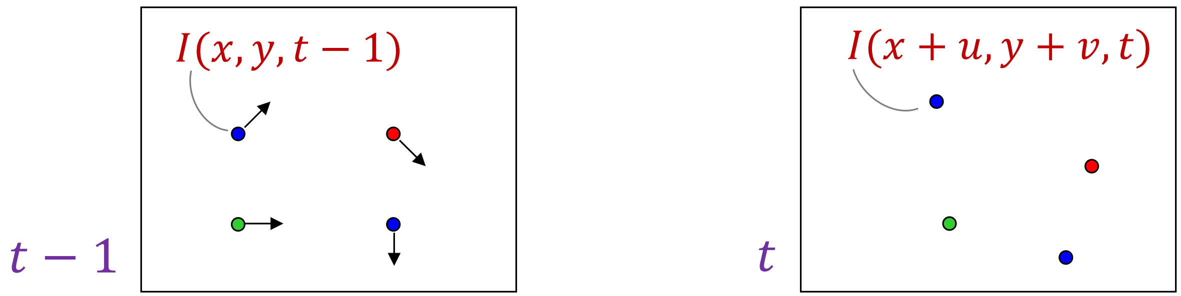


Source: Walker et al.

Outline

- Motion: motivation
- Brightness constancy constraint, aperture problem

Estimating optical flow



- Given frames at times $t-1$ and t, estimate the apparent motion field $u(x,y)$ and $v(x,y)$ between them
- **Brightness constancy constraint**: projection of the same point looks the same in every frame

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

- Additional assumptions:
  - **Small motion**: points do not move very far
  - **Spatial coherence**: points move like their neighbors

Estimating optical flow

- Brightness constancy constraint:

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

- Linearize the right-hand side using Taylor expansion:

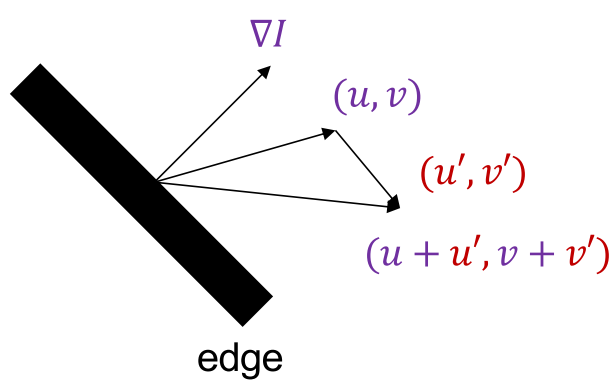
$$I(x, y, t-1) \approx I(x, y, t) + I_x u(x, y) + I_y v(x, y)$$
$$I_x u(x, y) + I_y v(x, y) + \underbrace{I(x, y, t) - I(x, y, t-1)}_{\text{What could this be?}} = 0$$

- Hence, $I_x u(x, y) + I_y v(x, y) + I_t = 0$

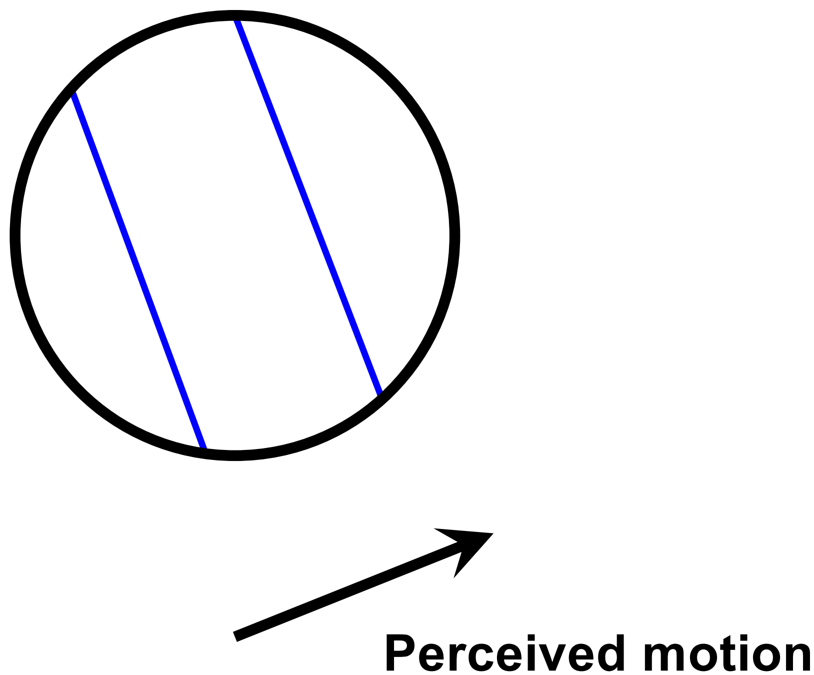
The brightness constancy constraint

$$I_x u(x,y) + I_y v(x,y) + I_t = 0$$

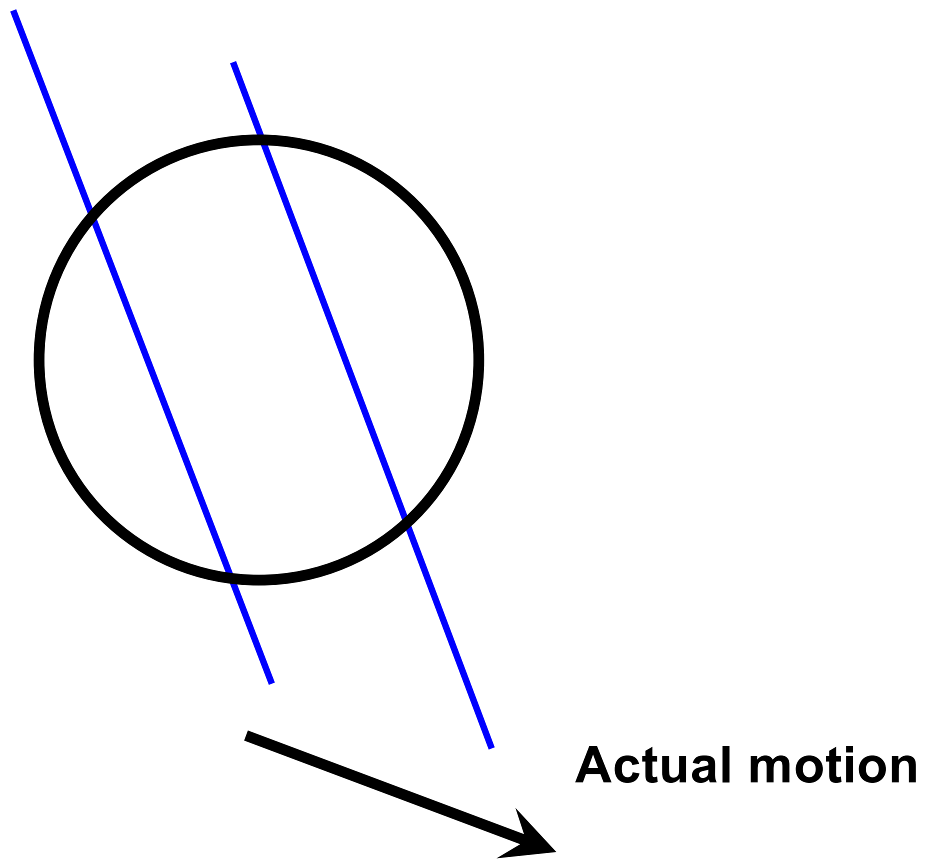
- Given the gradients I_x, I_y and I_t, can we uniquely recover the motion (u,v)?
  - Suppose (u,v) satisfies the constraint: $\nabla I \cdot (u,v) + I_t = 0$
  - Then $\nabla I \cdot (u+u', v+v') + I_t = 0$ for any (u',v') s. t. $\nabla I \cdot (u',v') = 0$
  - Interpretation: the component of the flow perpendicular to the gradient (i.e., parallel to the edge) cannot be recovered!



The aperture problem



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

Outline

- Motion: motivation
- Motion field and optical flow definition
- Brightness constancy constraint, aperture problem
- Estimating optical flow (Lucas-Kanade)

Solving the aperture problem

$$I_x u(x,y) + I_y v(x,y) + I_t = 0$$

- How to get more equations for a pixel?
  - **Spatial coherence constraint:** assume the pixel's neighbors have the same (u,v)
  - If we have n pixels in the neighborhood, then we can set up a linear least squares system:

$$\begin{bmatrix} I_x(x_1,y_1) & I_y(x_1,y_1) \\ \vdots & \vdots \\ I_x(x_n,y_n) & I_y(x_n,y_n) \end{bmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = -\begin{bmatrix} I_t(x_1,y_1) \\ \vdots \\ I_t(x_n,y_n) \end{bmatrix}$$

Estimating optical flow

$$\underbrace{\begin{bmatrix} I_x(x_1,y_1) & I_y(x_1,y_1) \\ \vdots & \vdots \\ I_x(x_n,y_n) & I_y(x_n,y_n) \end{bmatrix}}_{A} \underbrace{\begin{pmatrix} u \\ v \end{pmatrix}}_{d} = -\underbrace{\begin{bmatrix} I_t(x_1,y_1) \\ \vdots \\ I_t(x_n,y_n) \end{bmatrix}}_{b}$$

- Solution:

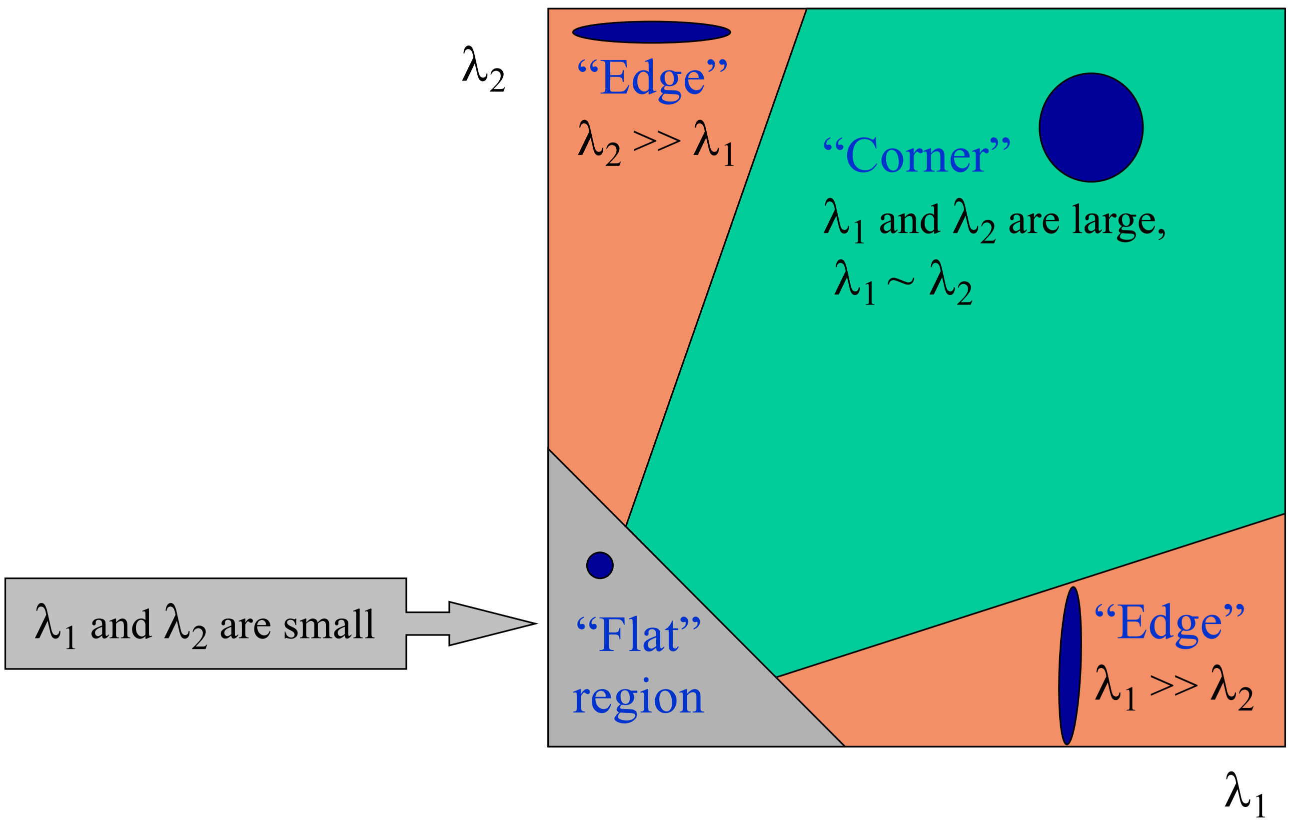
$$\underbrace{\begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix}}_{A^T A} \underbrace{\begin{pmatrix} u \\ v \end{pmatrix}}_{d} = -\underbrace{\begin{bmatrix} \sum I_x I_t \\ \sum I_y I_t \end{bmatrix}}_{A^T b}$$

What does $A^T A$ remind you of?

- When is the system solvable?

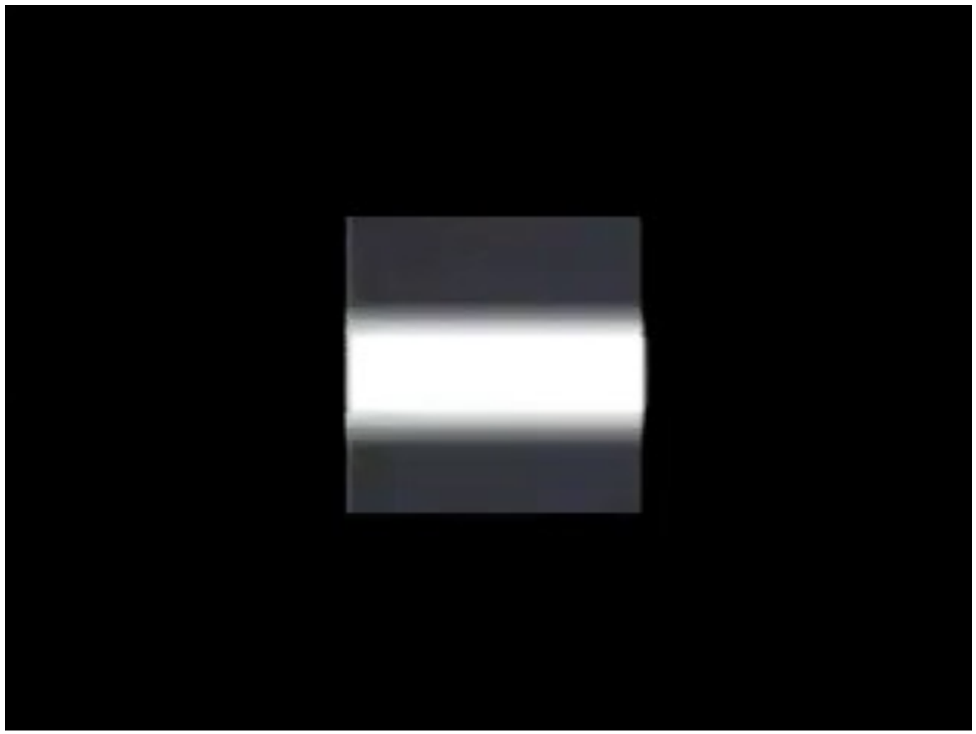
Recall: second moment matrix

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



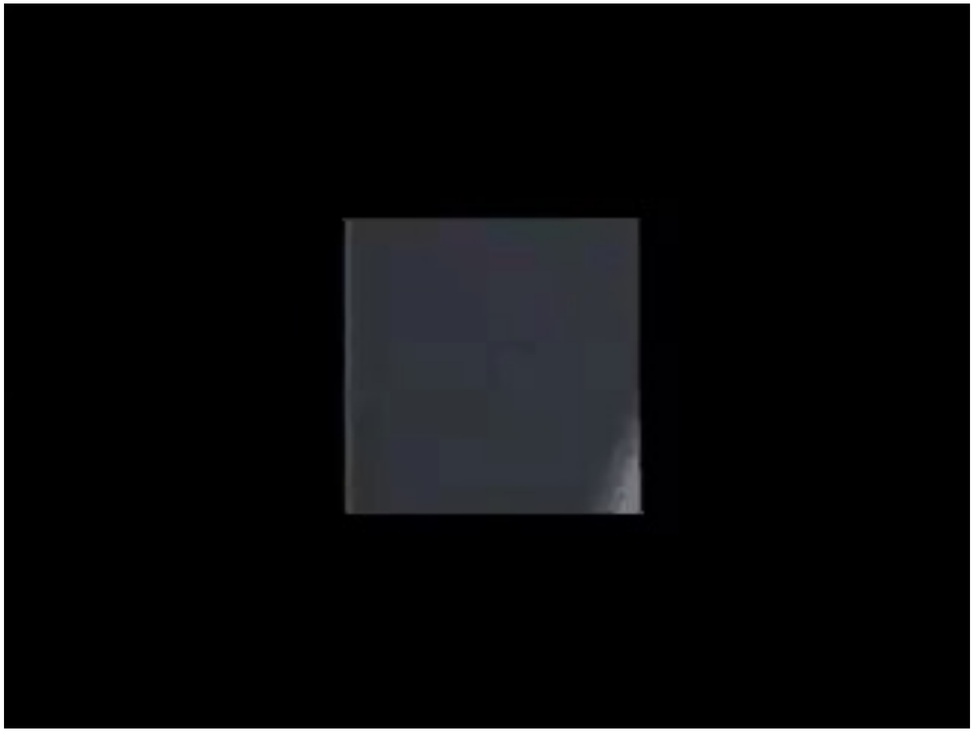
Conditions for solvability

- “Bad” case: single straight edge



Conditions for solvability

- “Good” case



Lucas-Kanade flow

$$\underbrace{\begin{bmatrix} I_x(x_1, y_1) & I_y(x_1, y_1) \\ \vdots & \vdots \\ I_x(x_n, y_n) & I_y(x_n, y_n) \end{bmatrix}}_{A} \underbrace{\begin{pmatrix} u \\ v \end{pmatrix}}_{d} = -\underbrace{\begin{bmatrix} I_t(x_1, y_1) \\ \vdots \\ I_t(x_n, y_n) \end{bmatrix}}_{b}$$

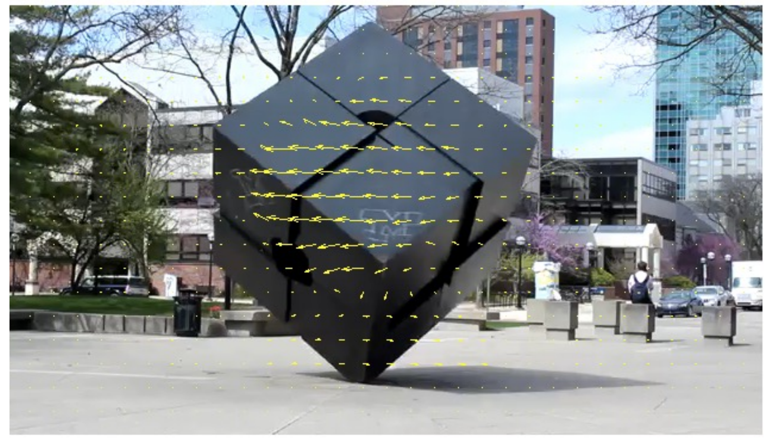
- Solution is given by $d = (A^T A)^{-1} A^T b$
- Lucas-Kanade flow:
  - Find (u, v) minimizing $\sum_i \big(I(x_i + u, y_i + v, t) - I(x_i, y_i, t-1)\big)^2$, use Taylor approximation of $I(x_i + u, y_i + v, t)$ for small shifts (u, v) to obtain closed-form solution

B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 674–679, 1981

Lucas-Kanade flow example

Input frames

Output



Source: MATLAB Central File Exchange

Outline

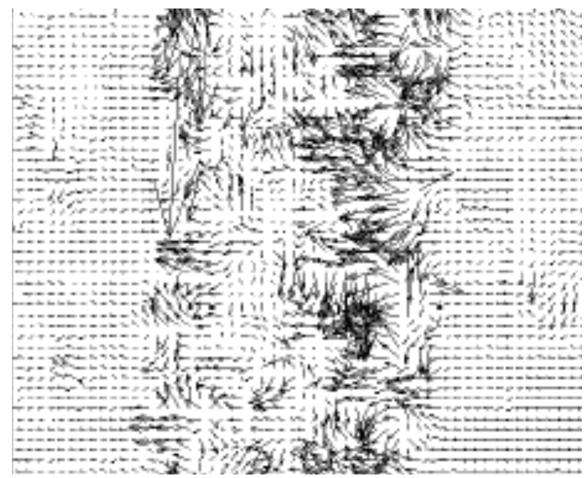
- Motion: motivation
- Motion field and optical flow definition
- Brightness constancy constraint, aperture problem
- Estimating optical flow (Lucas-Kanade)
- Beyond basic flow estimation

When does Lucas-Kanade fail?

- The motion is large (larger than a pixel)
- A point does not move like its neighbors
- Brightness constancy does not hold

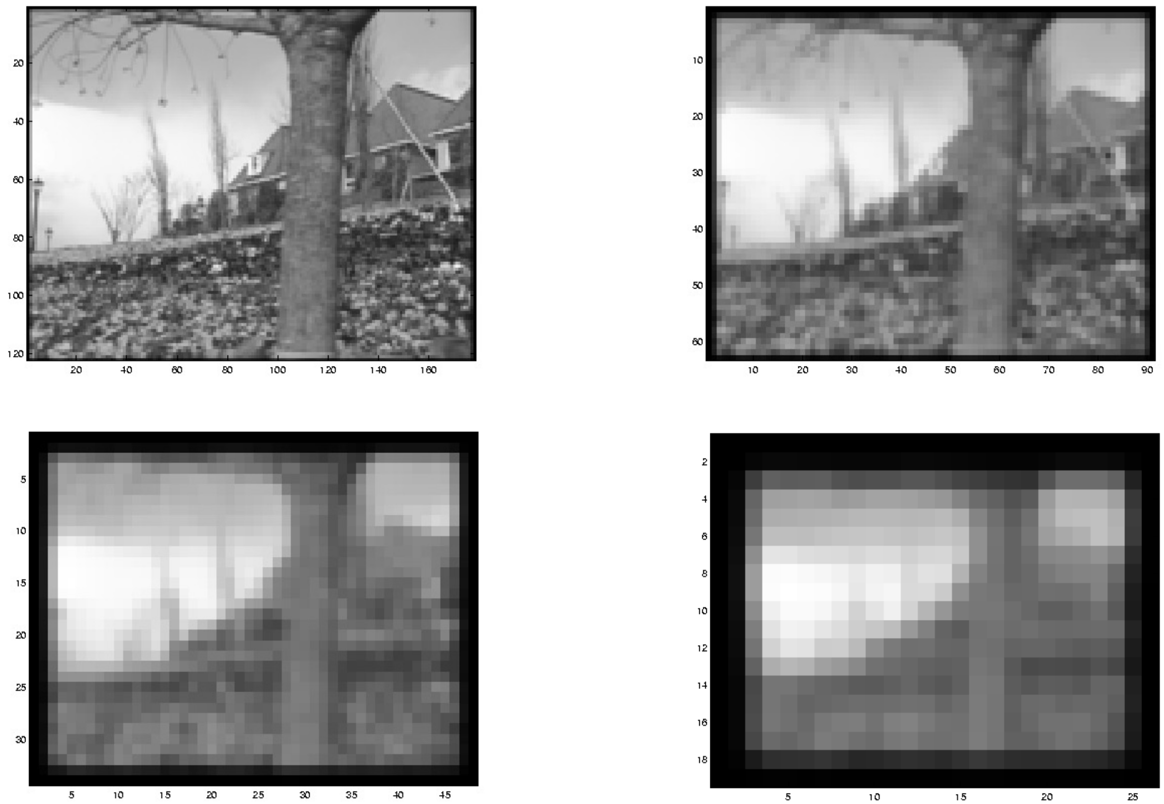
Fixing the errors in Lucas-Kanade

- The motion is large



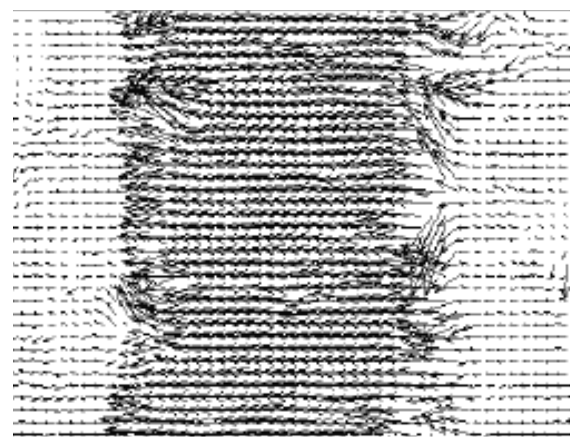
Fixing the errors in Lucas-Kanade

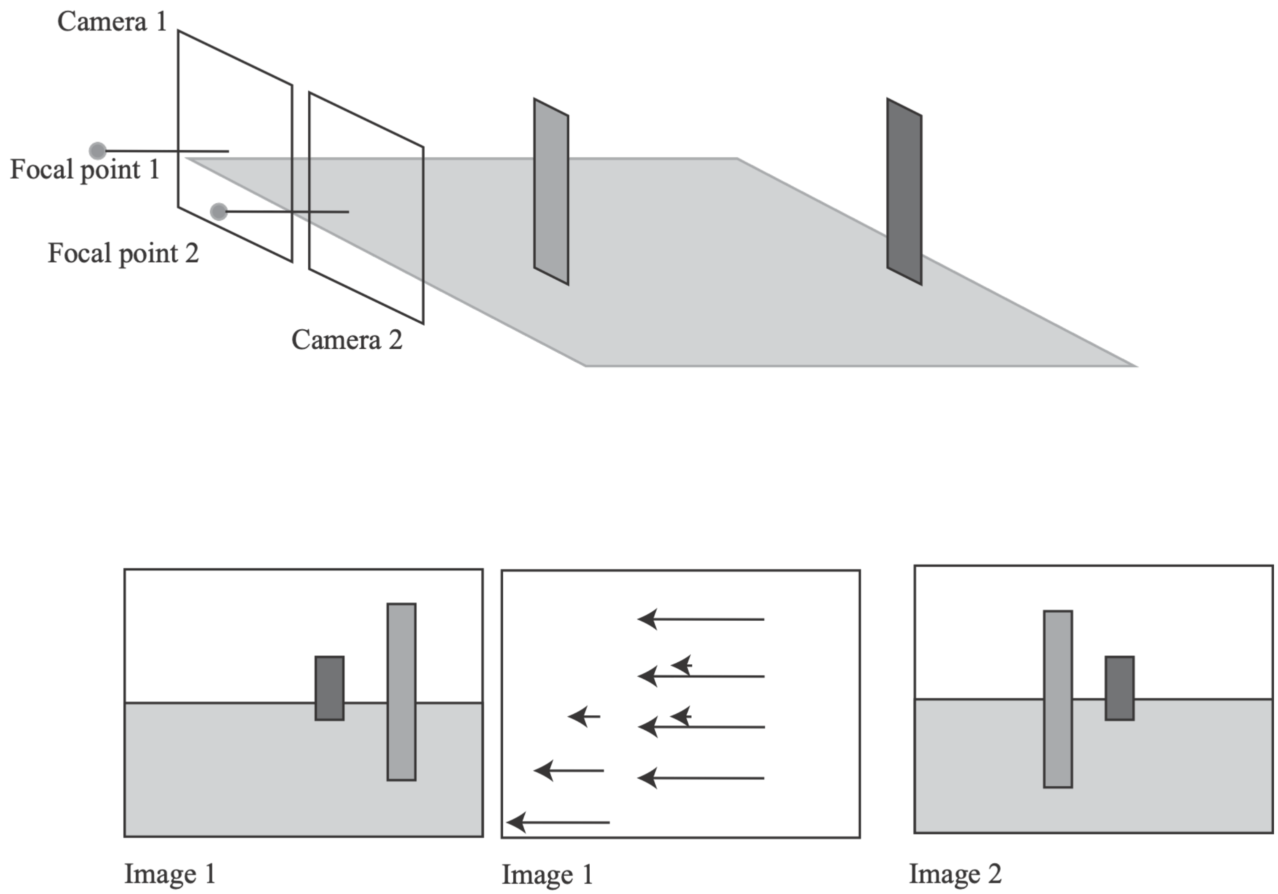
- The motion is large
  - Multi-resolution estimation, iterative refinement



Fixing the errors in Lucas-Kanade

- The motion is large
  - Multi-resolution estimation, iterative refinement




Camera 1
Focal point 1
Focal point 2
Camera 2
Image 1
Image 1
Image 2

Parametric flow models

Divide image into regions; in each region, flow is:

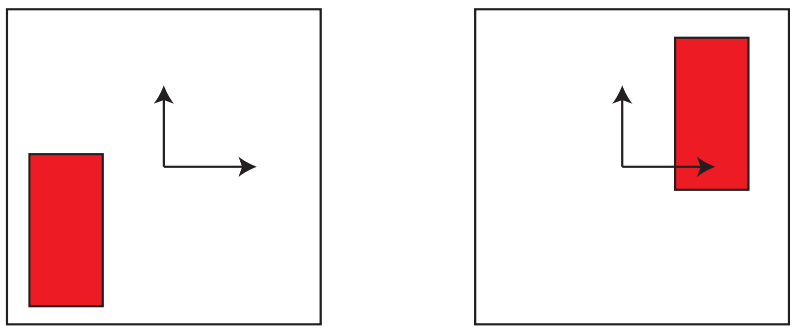
$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{bmatrix} a & b & c & d & e \\ f & g & h & i & j \end{bmatrix} \begin{bmatrix} 1 \\ x \\ y \\ x^2 \\ xy \\ y^2 \end{bmatrix}$$

Flow

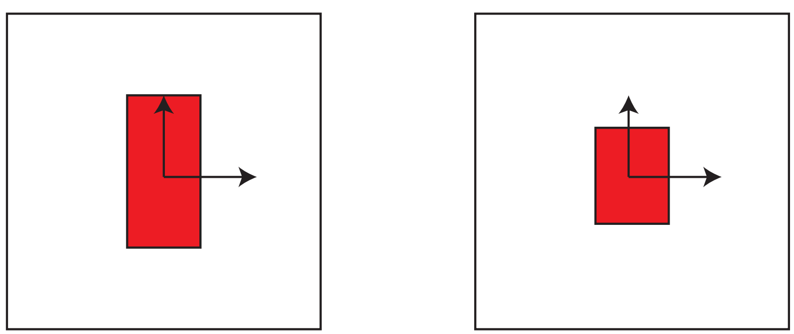
parameters

functions of pixel position

Examples

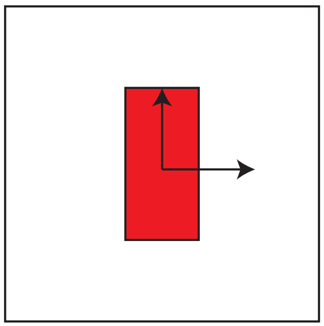


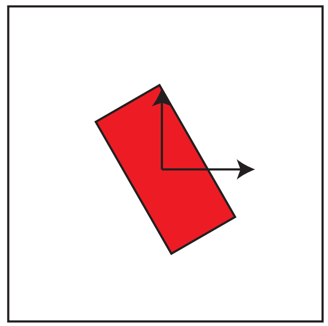
$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{bmatrix} 1 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ x \\ y \\ x^2 \\ xy \\ y^2 \end{bmatrix}$$



$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{bmatrix} 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ x \\ y \\ x^2 \\ xy \\ y^2 \end{bmatrix}$$

Examples





$$\begin{pmatrix} u \\ v \end{pmatrix} = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 \\ 0 & 1 & 0 & 0 & 0 \end{bmatrix} \begin{bmatrix} 1 \\ x \\ y \\ x^2 \\ xy \\ y^2 \end{bmatrix}$$

Estimating

This is a clustering problem:

- we have k regions, k flow models
  - If we knew the regions, we could estimate the flow
    - (eg compute a Lucas-Kanade soln, fit a flow model with least squares)
  - If we knew the flow models, we could estimate the regions
    - (eg, for each pixel, apply flow models, choose best)

Roughly

Initialize regions

  Repeat:

    Estimate flow from regions

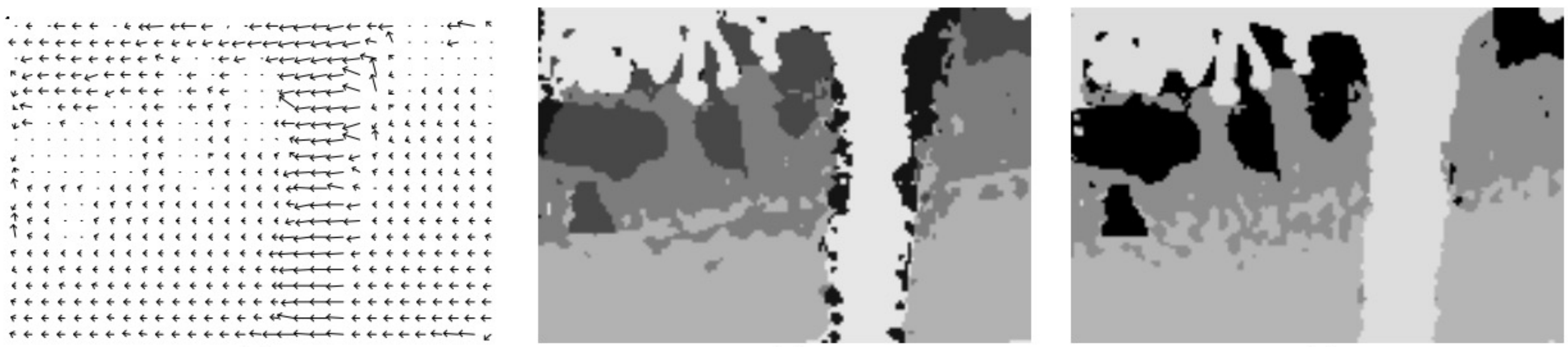
    Estimate regions from flow

Product:

  k flow fields AND k regions

Fixing the errors in Lucas-Kanade

- The motion is large (larger than a pixel)
- A point does not move like its neighbors
  - Motion segmentation



(a) (b) (c)

Figure 11: (a) The optic flow from multi-scale gradient method. (b) Segmentation obtained by clustering optic flow into affine motion regions. (c) Segmentation from consistency checking by image warping. Representing moving images with layers.

J. Wang and E. Adelson, Representing Moving Images with Layers, IEEE Transactions on Image Processing, 1994

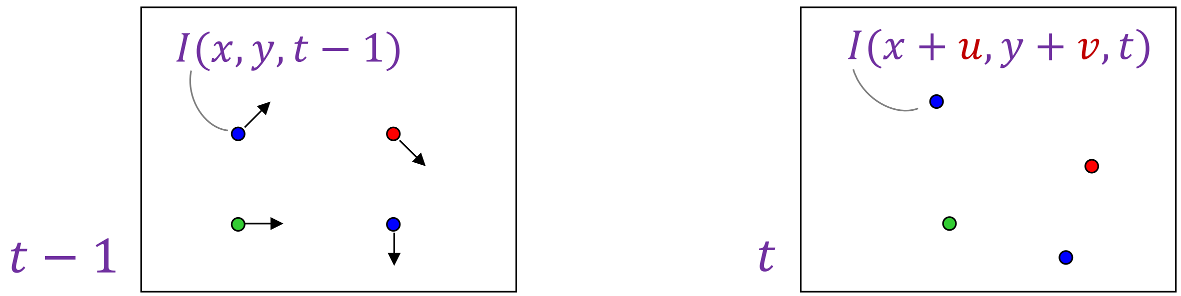
Fixing the errors in Lucas-Kanade

- The motion is large (larger than a pixel)
- A point does not move like its neighbors
- Brightness constancy does not hold
  - Feature matching (e.g., SIFT) or tracking

Last time: Optical flow

- Motion: motivation
- Motion field and optical flow definition
- Brightness constancy constraint, aperture problem
- Estimating optical flow (Lucas-Kanade)
- Beyond basic flow estimation

Review: Estimating optical flow

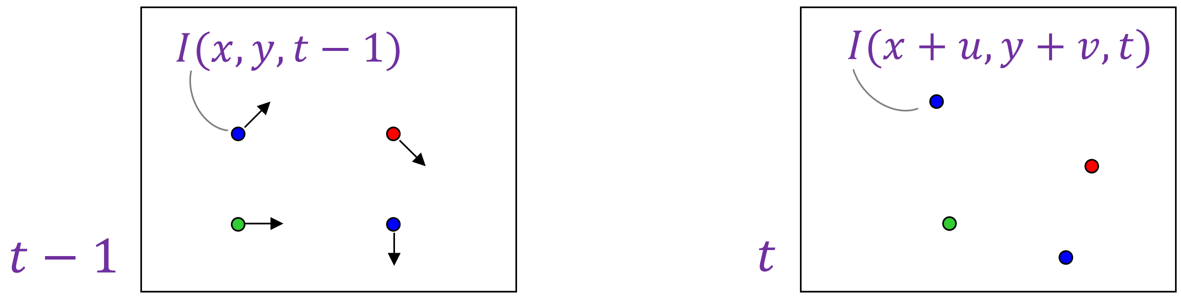


- Given frames at times $t-1$ and t, estimate the apparent motion field $u(x, y)$ and $v(x, y)$ between them
- **Brightness constancy constraint**: projection of the same point looks the same in every frame

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

- Additional assumptions:
  - **Small motion**: points do not move very far
  - **Spatial coherence**: points move like their neighbors

Review: Estimating optical flow



- Lucas-Kanade flow: find (u, v) minimizing SSD between pixel values in corresponding locations in a patch at times $t-1$ and t:

$$\sum_i \left(I(x_i + u, y_i + v, t) - I(x_i, y_i, t-1) \right)^2$$

  - Use Taylor approximation of $I(x_i + u, y_i + v, t)$ for small shifts (u, v) to obtain a linear least squares problem

B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 674–679, 1981

Review: Lucas-Kanade flow

$$\begin{bmatrix} I_x(x_1, y_1) & I_y(x_1, y_1) \\ \vdots & \vdots \\ I_x(x_n, y_n) & I_y(x_n, y_n) \end{bmatrix} \begin{pmatrix} u \\ v \end{pmatrix} = -\begin{bmatrix} I_t(x_1, y_1) \\ \vdots \\ I_t(x_n, y_n) \end{bmatrix}$$

$$A \qquad\qquad d \qquad\qquad b$$

- Solution is given by $d = (A^T A)^{-1} A^T b$
- But when does the solution exist?

B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of the International Joint Conference on Artificial Intelligence*, pp. 674–679, 1981

Shi-Tomasi feature tracker

- Find good features using eigenvalues of second-moment matrix
  - Key idea: "good" features to track are the ones whose motion can be estimated reliably
- From frame to frame, track with Lucas-Kanade
  - This amounts to assuming a translation model for frame-to-frame feature movement
- Check consistency of tracks by *affine* registration to the first observed instance of the feature
  - Affine model is more accurate for larger displacements
  - Comparing to the first frame helps to minimize drift

J. Shi and C. Tomasi. Good Features to Track. CVPR 1994.

Tracking example

Figure 1: Three frame details from Woody Allen's *Manhattan*. The details are from the 1st, 11th, and 21st frames of a subsequence from the movie.

Figure 2: The traffic sign windows from frames 1,6,11,16,21 as tracked (top), and warped by the computed deformation matrices (bottom).

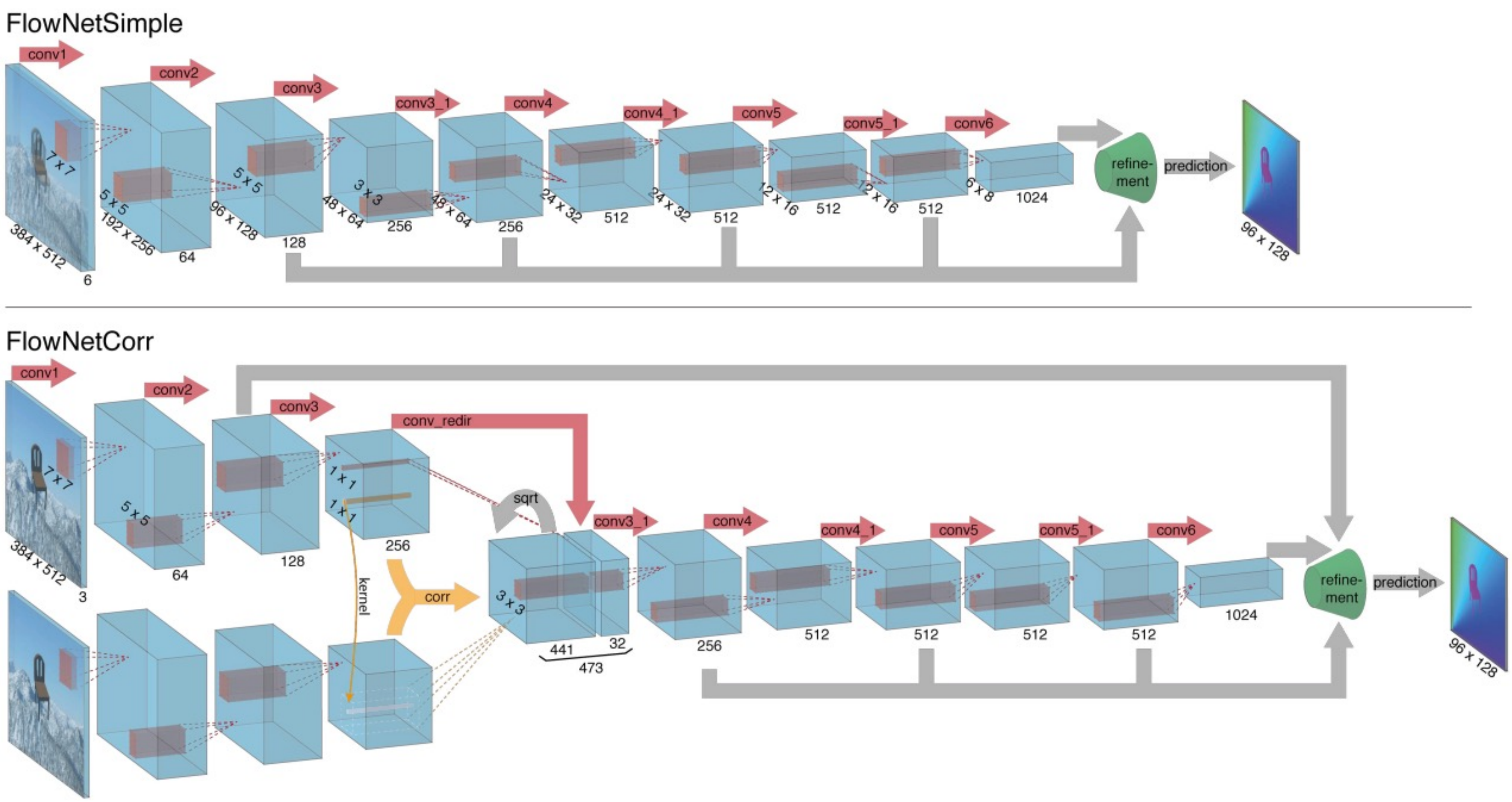
J. Shi and C. Tomasi. Good Features to Track. CVPR 1994.

Outline

- Motion: motivation
- Motion field and optical flow definition
- Brightness constancy constraint, aperture problem
- Lucas-Kanade flow estimation
- Beyond basic flow estimation
- Current approaches, applications

State-of-the-art optical flow estimation

- Current best methods are learned (often on synthetic data)



A. Dosovitskiy et al., [FlowNet: Learning Optical Flow with Convolutional Networks](), ICCV 2015

State-of-the-art optical flow estimation

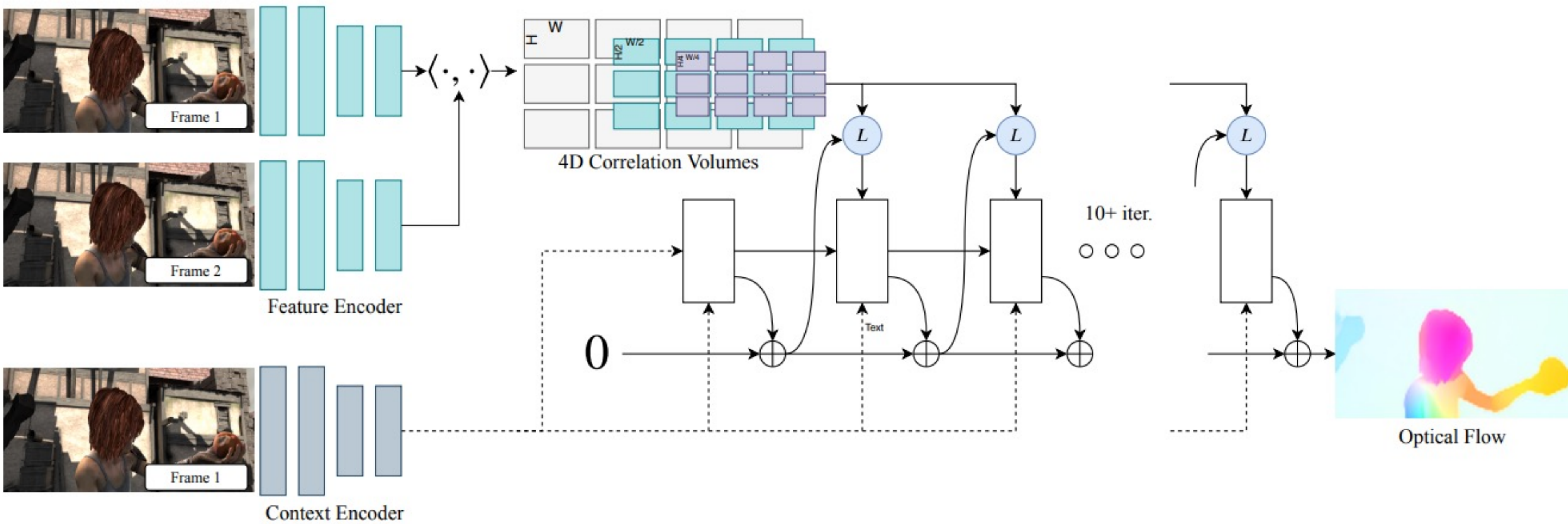
- Current best methods are learned (often on synthetic data)

Synthetic dataset: Flying Chairs

A. Dosovitskiy et al., FlowNet: Learning Optical Flow with Convolutional Networks, ICCV 2015

State-of-the-art optical flow estimation

- Current best methods are learned (often on synthetic data)



Z. Teed and J. Deng. [RAFT: Recurrent All-Pairs Field Transforms for Optical Flow](). ECCV 2020 (Best Paper Award)

Motion magnification

Idea: take flow, magnify it

C. Liu et al., Motion Magnification, SIGGRAPH 2005

Source: D. Fouhey and J. Johnson

Motion magnification

C. Liu et al., Motion Magnification, SIGGRAPH 2005

Source: D. Fouhey and J. Johnson

Motion magnification

C. Liu et al., Motion Magnification, SIGGRAPH 2005

Source: D. Fouhey and J. Johnson

Motion magnification

(a) Registered input frame (b) Clustered trajectories of tracked features (c) Layers of related motion and appearance

(d) Motion magnified, showing holes (e) After texture in-painting to fill holes (f) After user's modification to segmentation map in (c)

C. Liu et al., Motion Magnification, SIGGRAPH 2005

Source: D. Fouhey and J. Johnson

Motion magnification

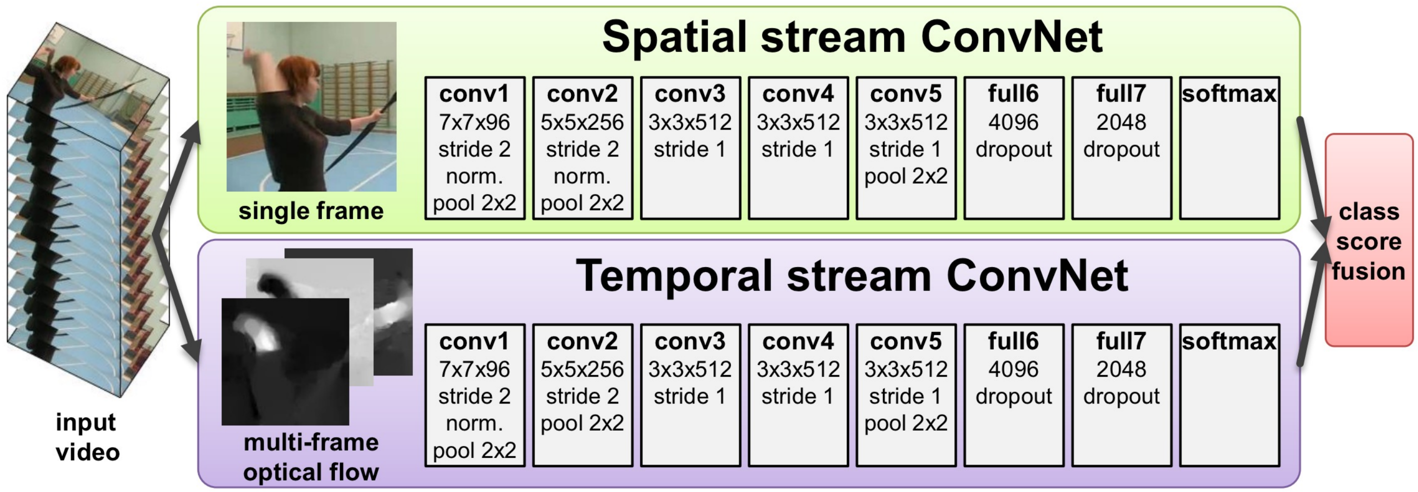
(a) Input

(b) Magnified

H. Wu et al. Eulerian Video Magnification for Revealing Subtle Changes in the World. SIGGRAPH 2012

Activity recognition

- Optical Flow is commonly used as an input feature for video classification with CNNs



K. Simonyan and A. Zisserman. [Two-Stream Convolutional Networks for Action Recognition in Videos](). NeurIPS 2014

Source: [D. Fouhey and J. Johnson]()