Optical flow

Subhransu Maji

CMPSCI 670: Computer Vision

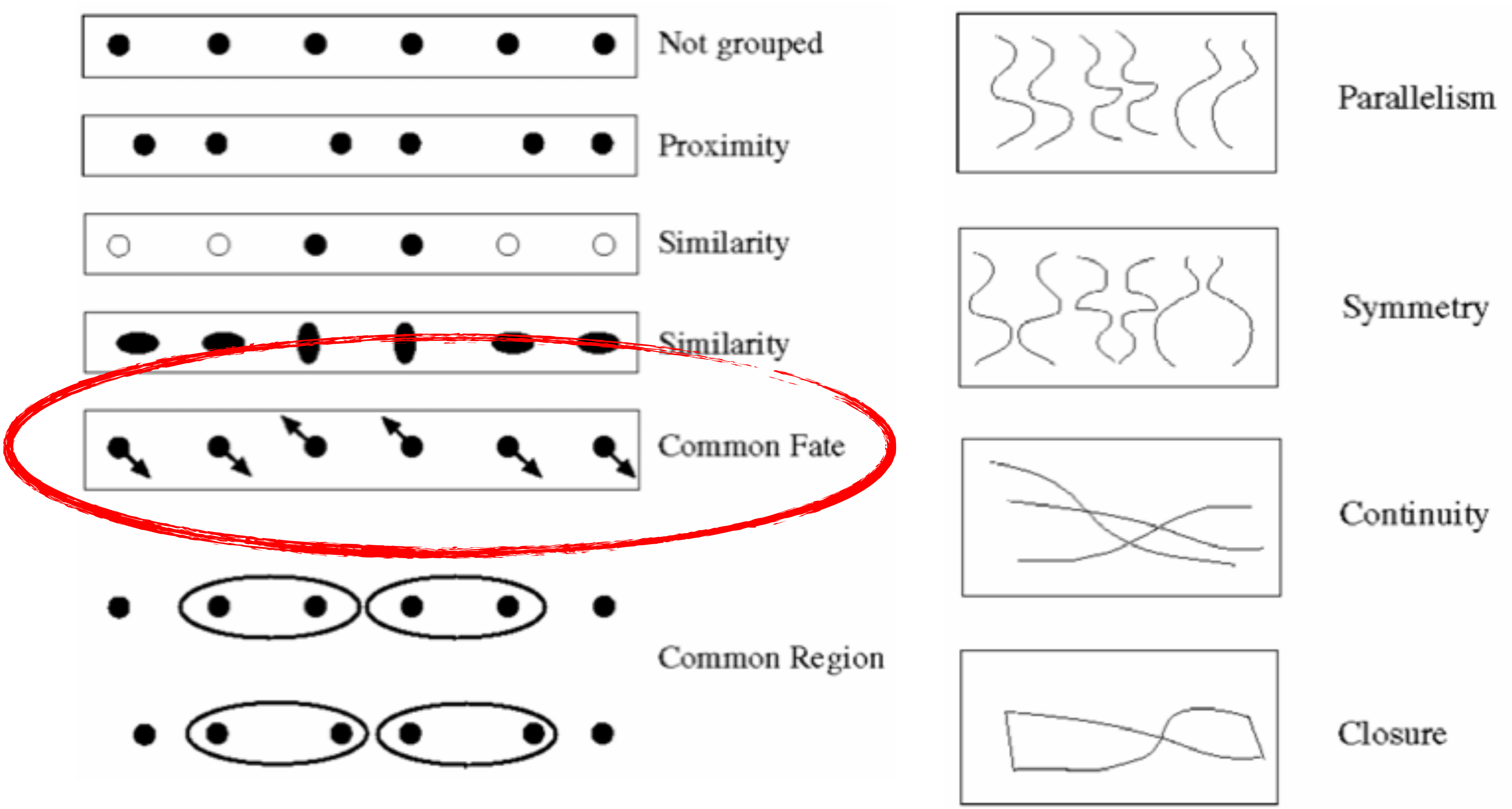
October 20, 2016

Visual motion

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

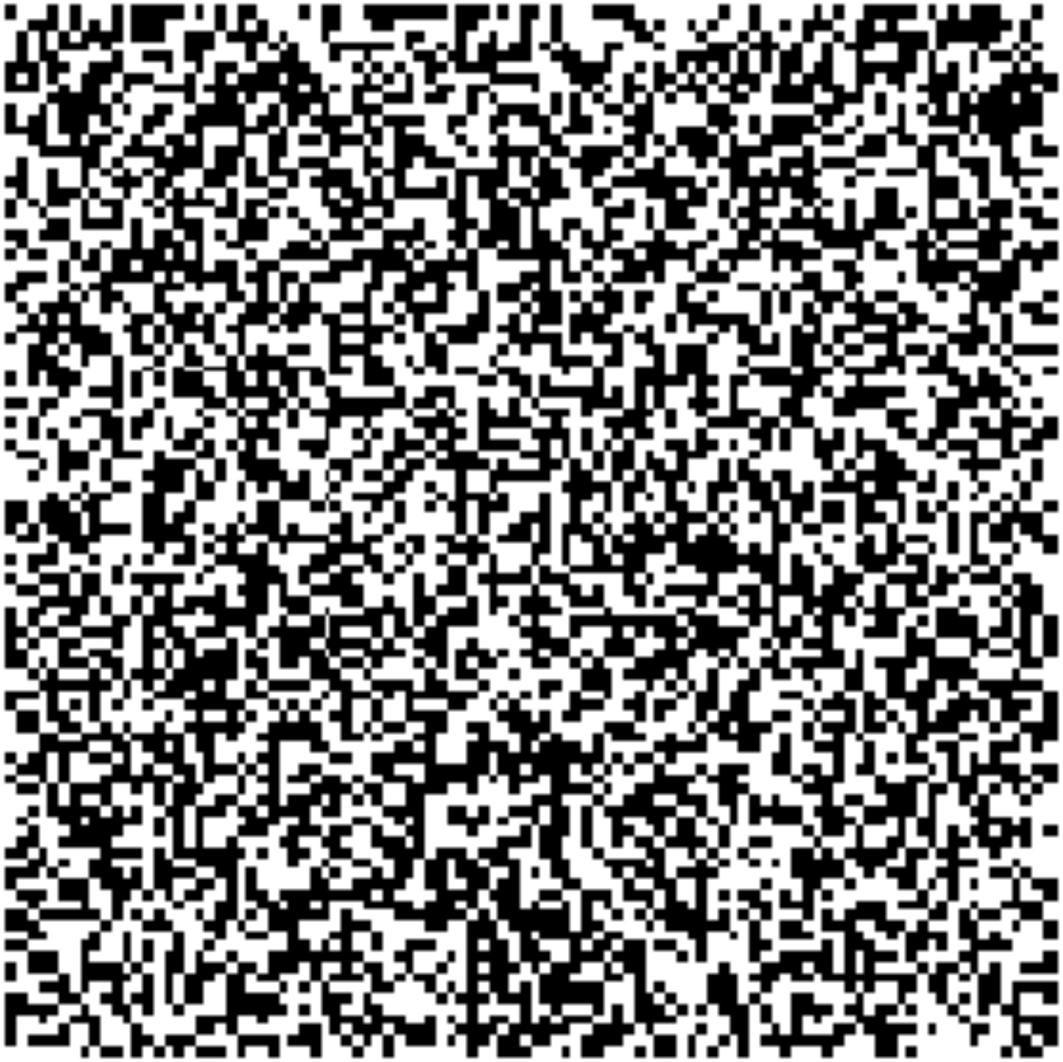
Motion and perceptual organization

- Sometimes, motion is the only cue



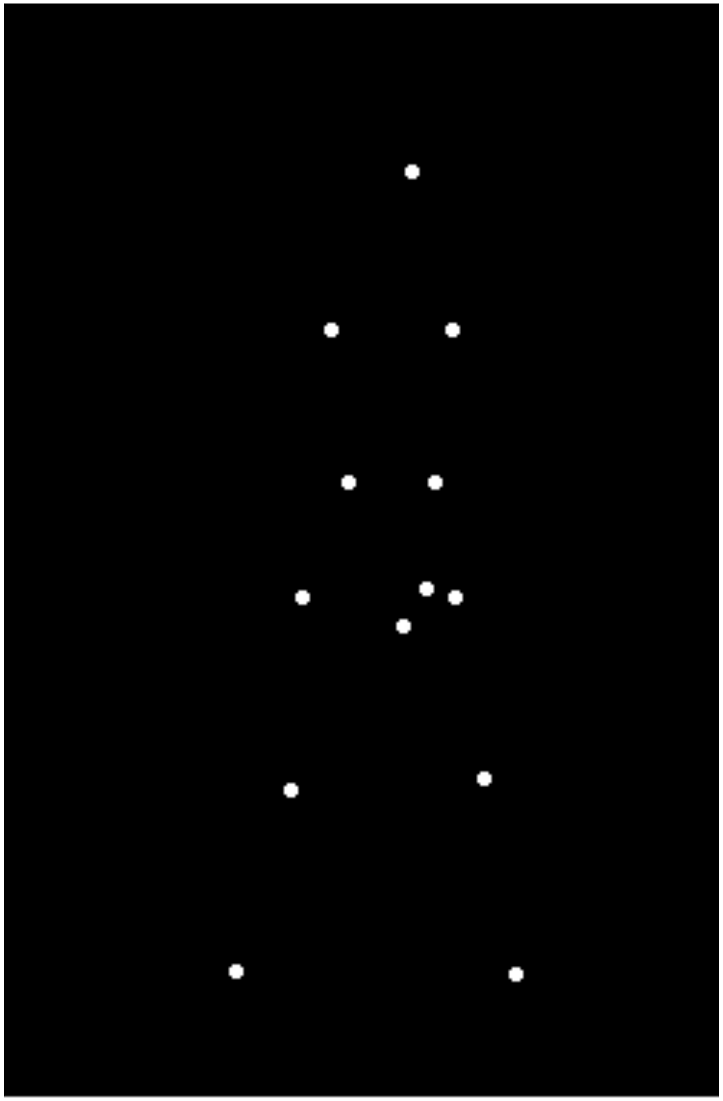
Motion and perceptual organization

- Sometimes, motion is the only cue



Motion and perceptual organization

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



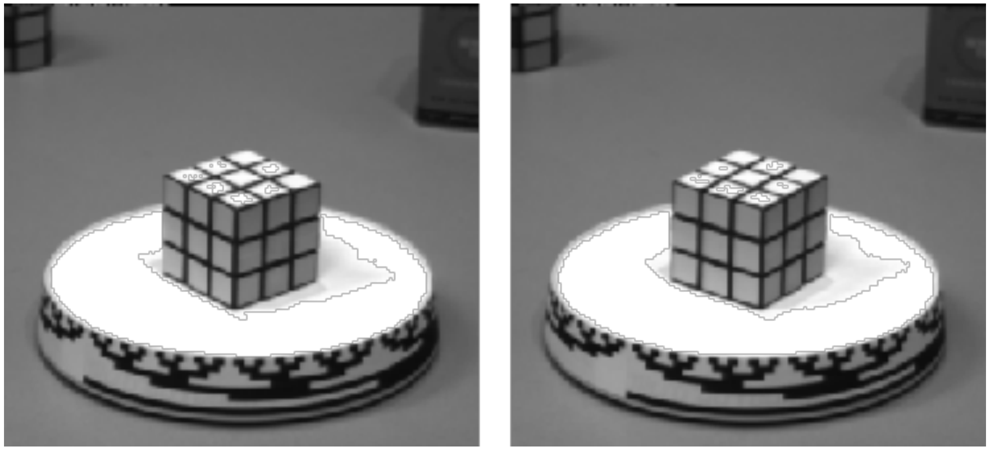
G. Johansson, “Visual Perception of Biological Motion and a Model For Its Analysis", Perception and Psychophysics 14, 201-211, 1973.

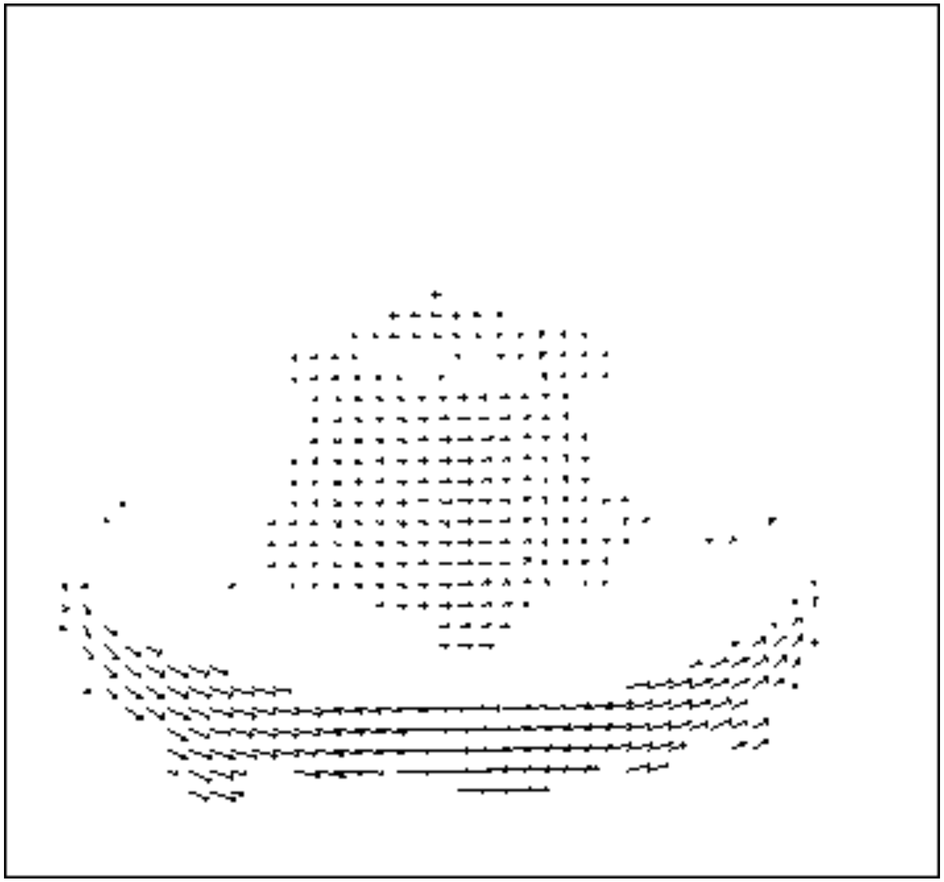
Uses of motion

- Segmenting objects based on motion cues
- Estimating the 3D structure
- Learning and tracking dynamical models
- Recognizing events and activities

Motion field

- The motion field is the projection of the 3D scene motion into the image



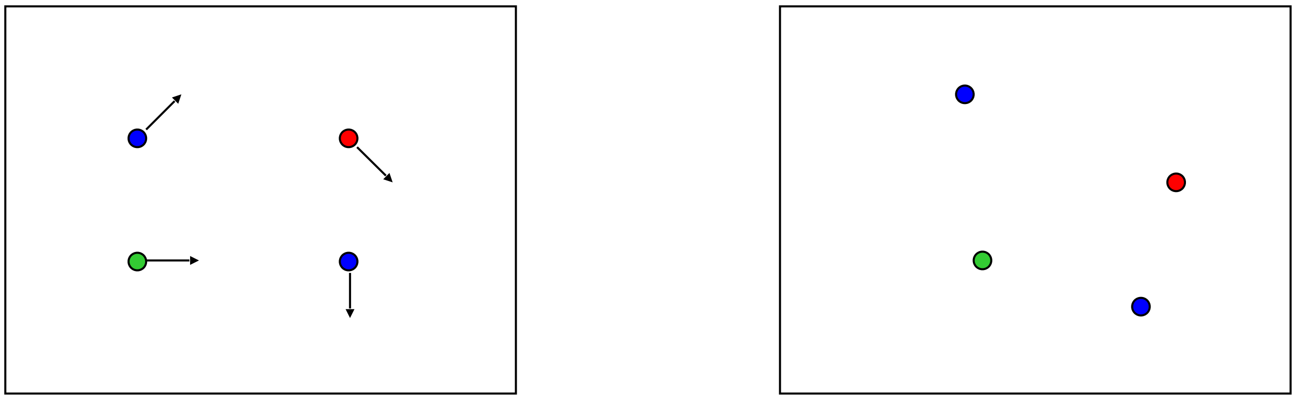


Optical flow

- **Definition**: optical flow is the apparent motion of brightness patterns in the image
- Ideally, optical flow would be the same as the motion field
- Have to be careful: apparent motion can be caused by lighting changes without any actual motion
  - Think of a uniform rotating sphere under fixed lighting vs. a stationary sphere under moving illumination

Estimating optical flow

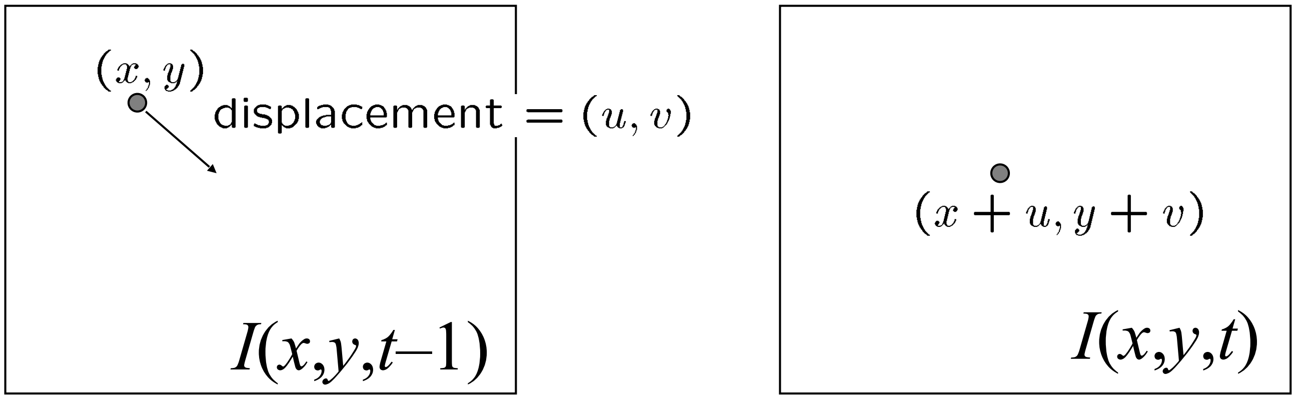
- Given two subsequent frames, estimate the apparent motion field *u*(*x*,*y*) and *v*(*x*,*y*) between them



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

- Key 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



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:

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

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

The brightness constancy constraint

- How many equations and unknowns per pixel?
  - One equation, two unknowns

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

- 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

The brightness constancy constraint

- How many equations and unknowns per pixel?
  - One equation, two unknowns

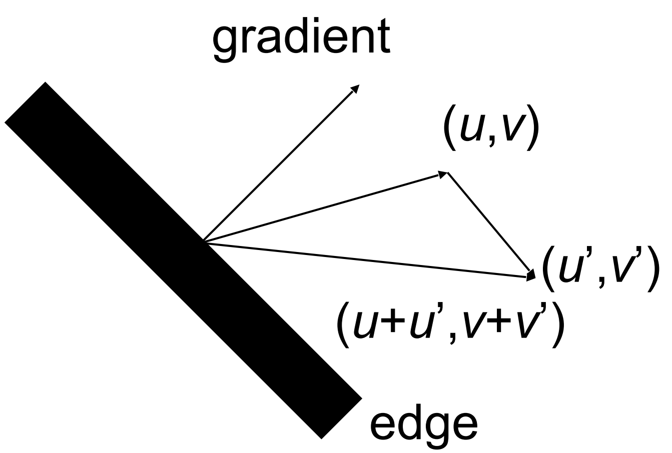
$$I_x u + I_y v + I_t = 0$$

- What does this constraint mean?

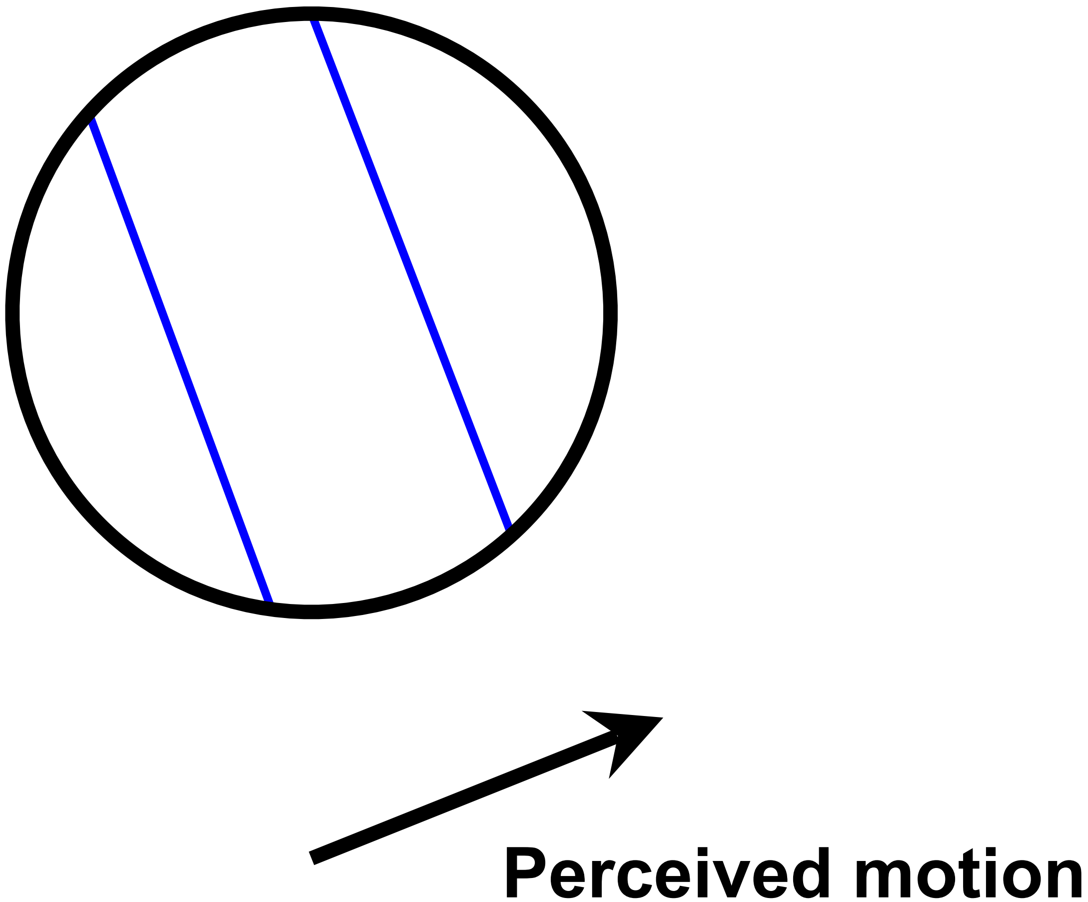
$$\nabla I \cdot (u, v) + I_t = 0$$

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

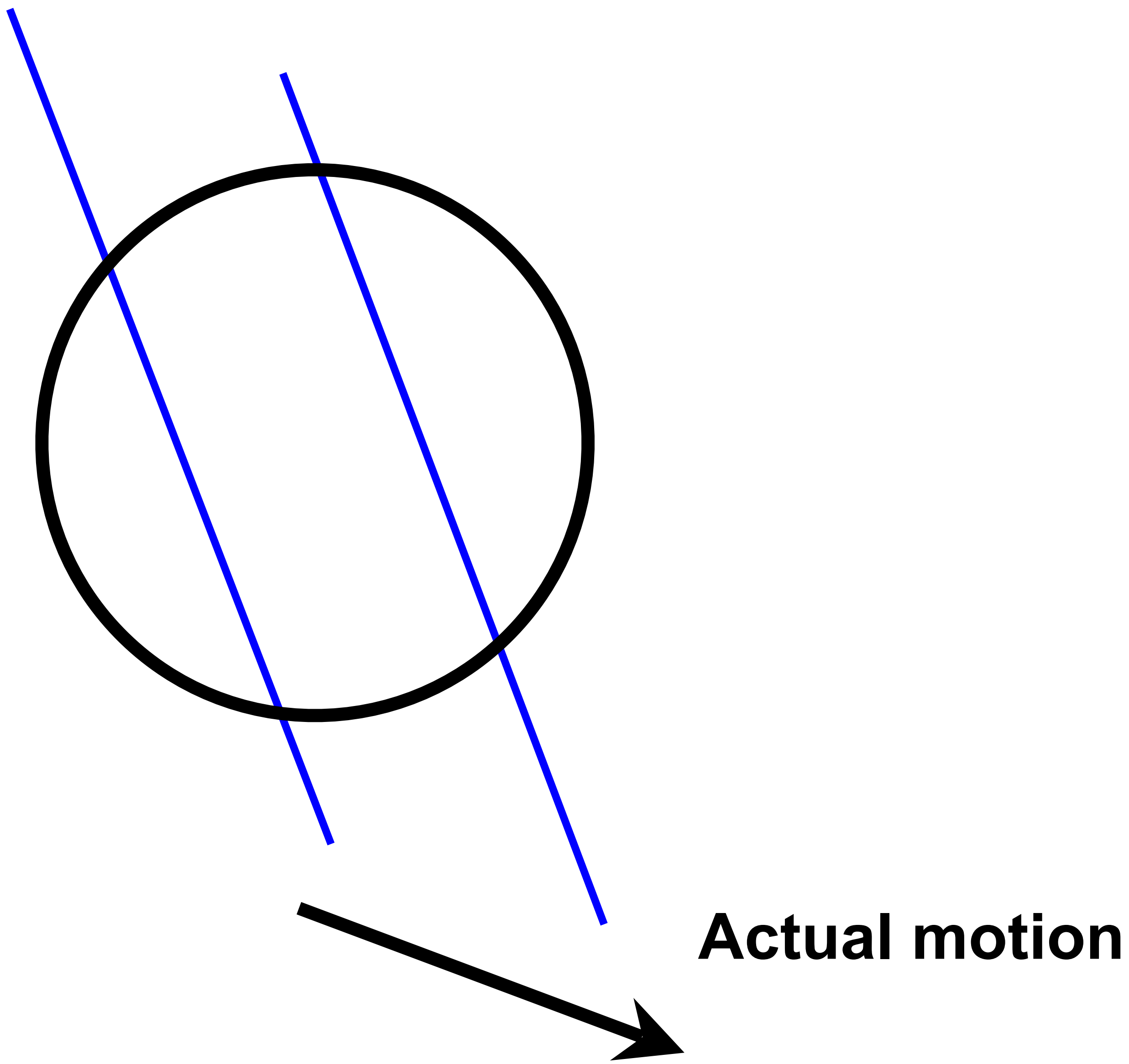
If (u, v) satisfies the equation,
so does ($u+u'$, $v+v'$) if $\nabla I \cdot (u', v') = 0$



The aperture problem

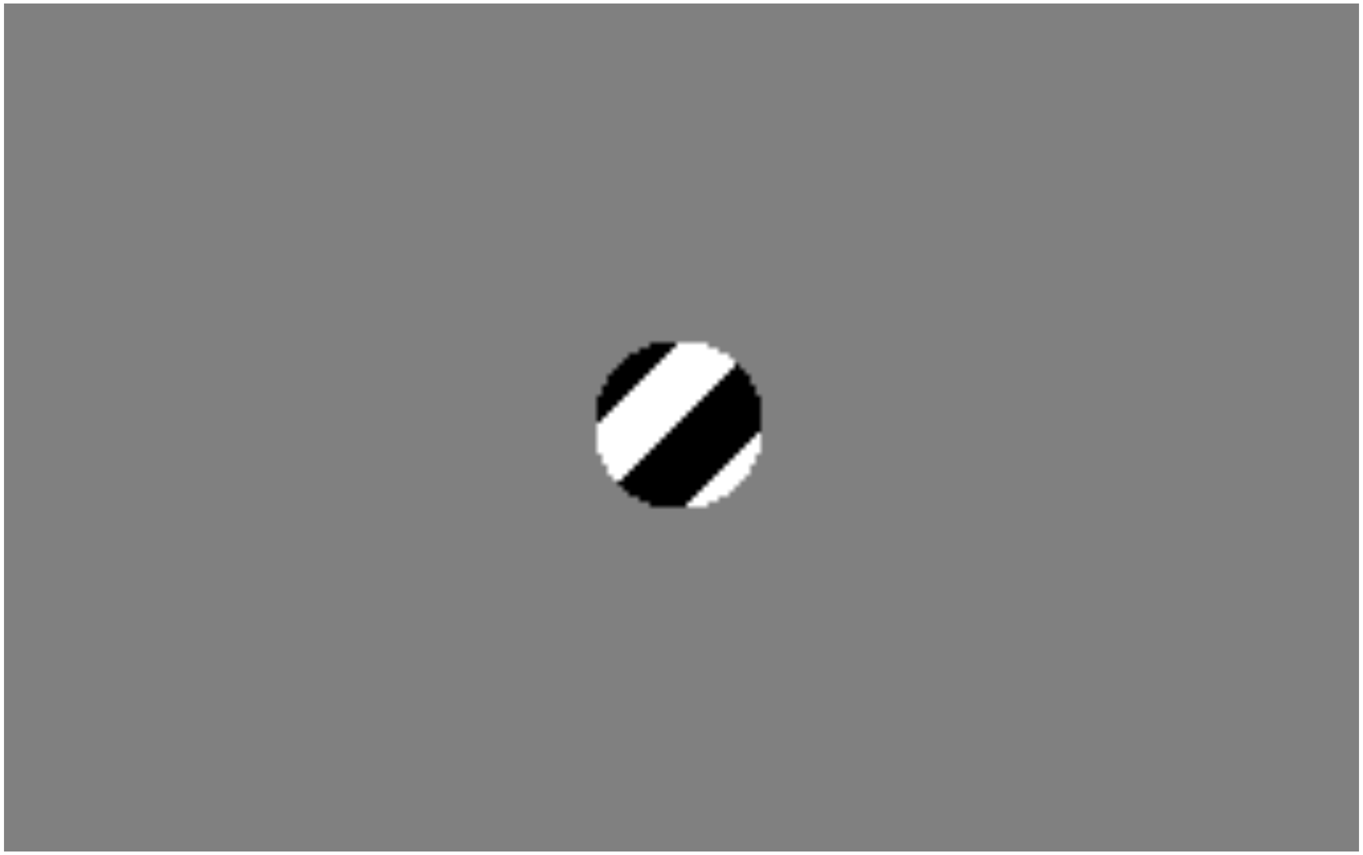


The aperture problem



The aperture problem

What direction is the motion?



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:** pretend the pixel's neighbors have the same (u,v)
  - E.g., if we use a 5x5 window, that gives us 25 equations per pixel

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

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

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.

Solving the aperture problem

- Least squares problem:

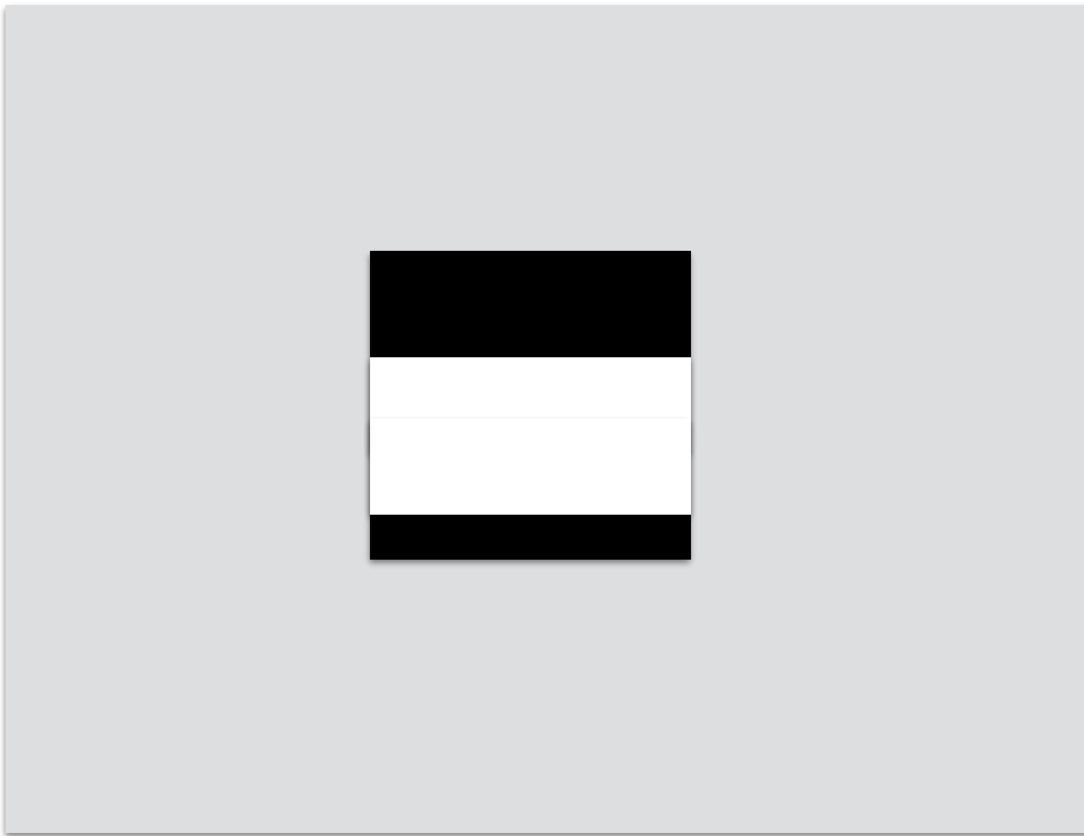
$$\begin{bmatrix} I_x(\mathbf{x}_1) & I_y(\mathbf{x}_1) \\ I_x(\mathbf{x}_2) & I_y(\mathbf{x}_2) \\ \vdots & \vdots \\ I_x(\mathbf{x}_n) & I_y(\mathbf{x}_n) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(\mathbf{x}_1) \\ I_t(\mathbf{x}_2) \\ \vdots \\ I_t(\mathbf{x}_n) \end{bmatrix}$$

- When is this system solvable?
  - What if the window contains just a single straight edge?

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.

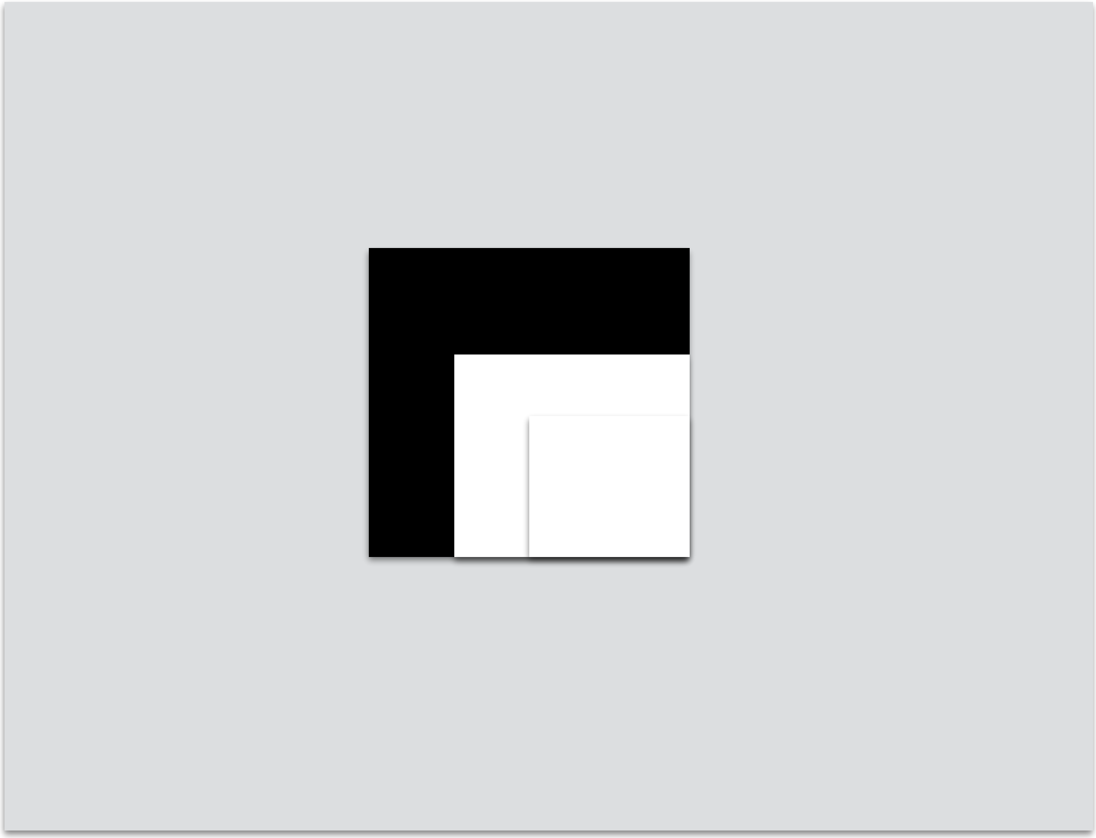
Conditions for solvability

- “Bad” case: single straight edge



Conditions for solvability

- "Good" case: corner



Lucas-Kanade flow

Linear least squares problem

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

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

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

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

The summations are over all pixels in the window

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.

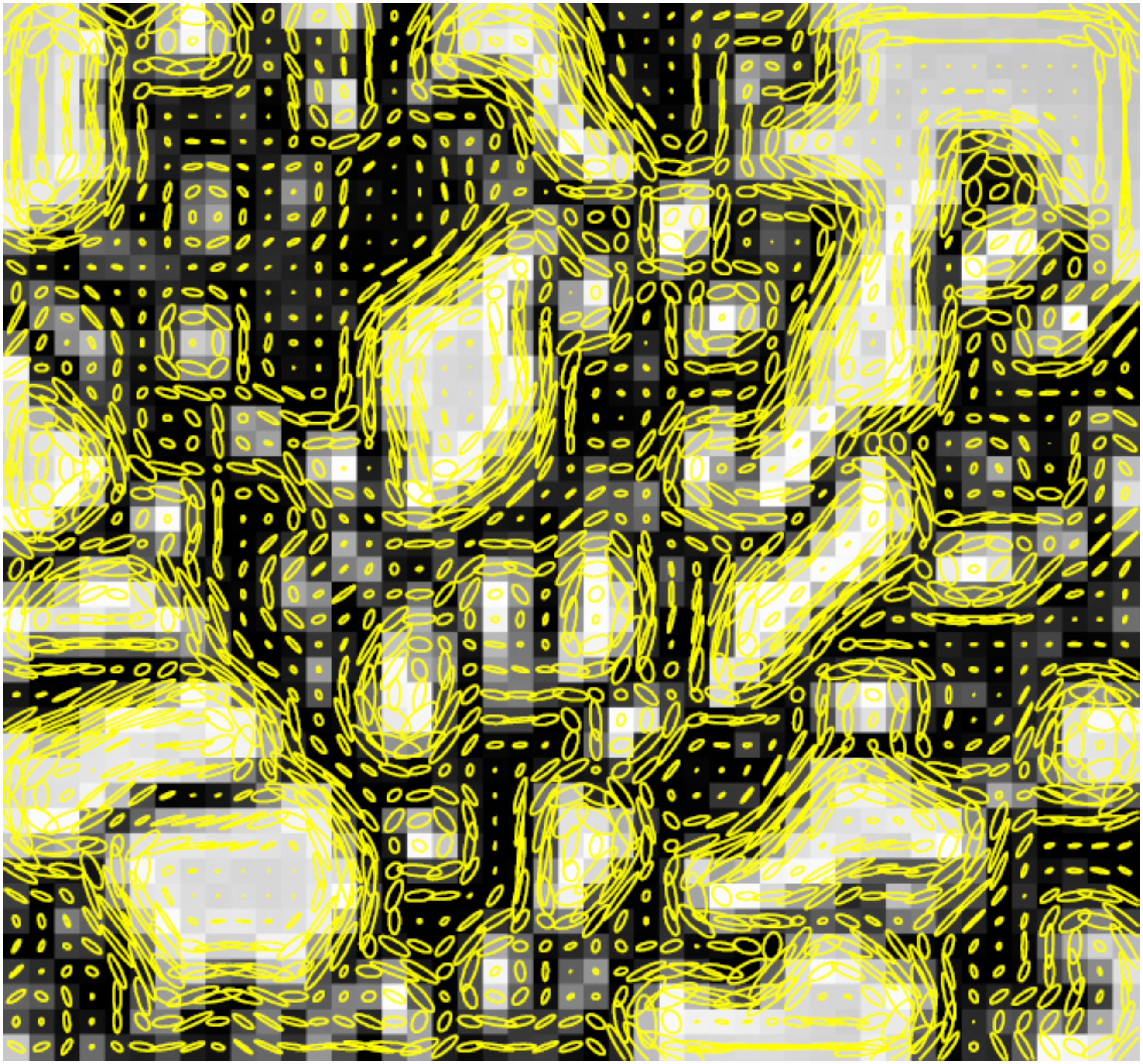
Lucas-Kanade flow

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

- Recall the Harris corner detector: $\mathbf{M} = \mathbf{A}^T\mathbf{A}$ is the *second moment matrix*
- We can figure out whether the system is solvable by looking at the eigenvalues of the second moment matrix
  - The eigenvectors and eigenvalues of **M** relate to edge direction and magnitude
  - The eigenvector associated with the larger eigenvalue points in the direction of fastest intensity change, and the other eigenvector is orthogonal to it

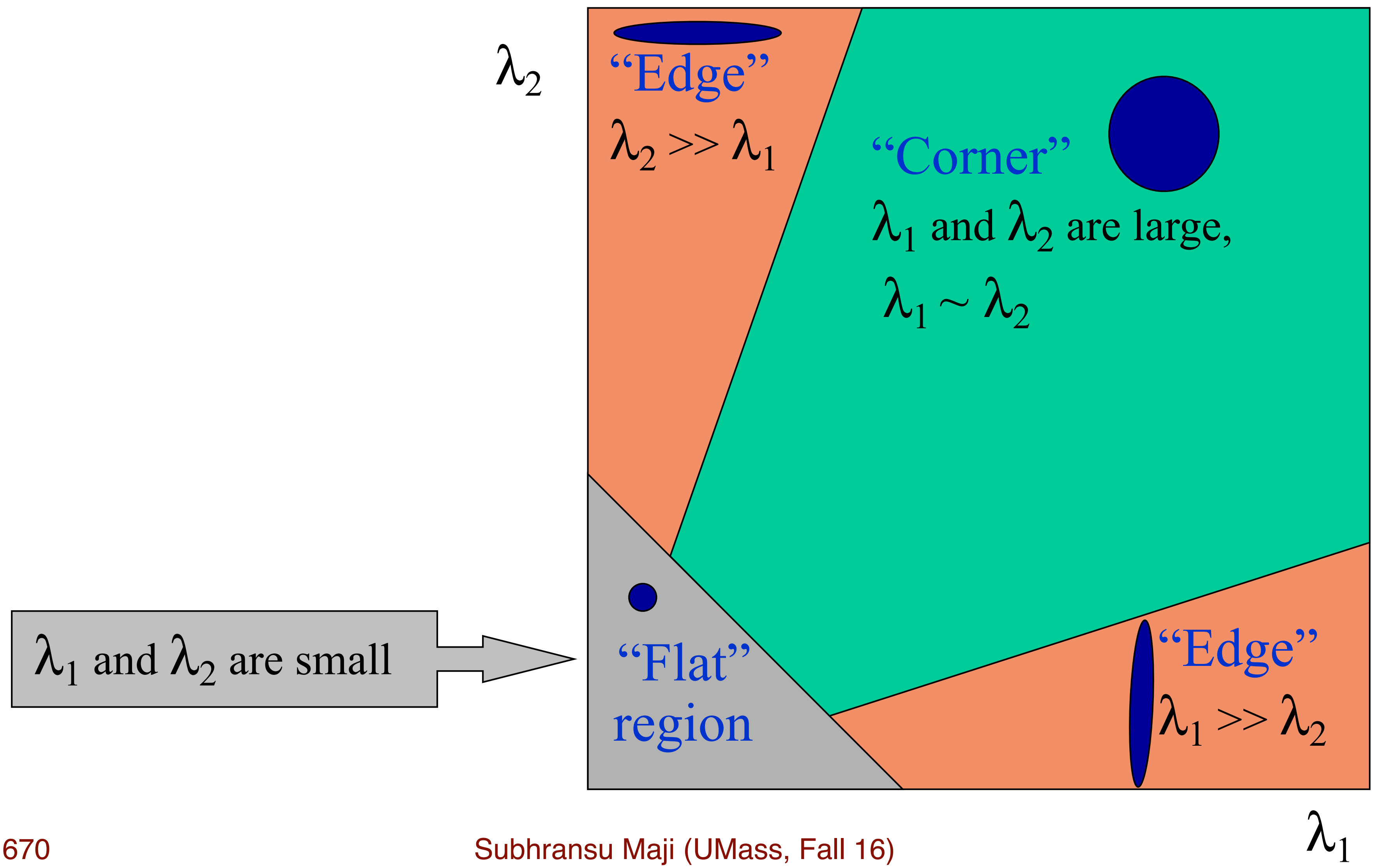
Visualization of second moment matrices

Visualization of second moment matrices



Interpreting the eigenvalues

Classification of image points using eigenvalues of the second moment matrix:



Example

*From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Uniform region

– gradients have small magnitude
– small λ_1, small λ_2
– system is ill-conditioned

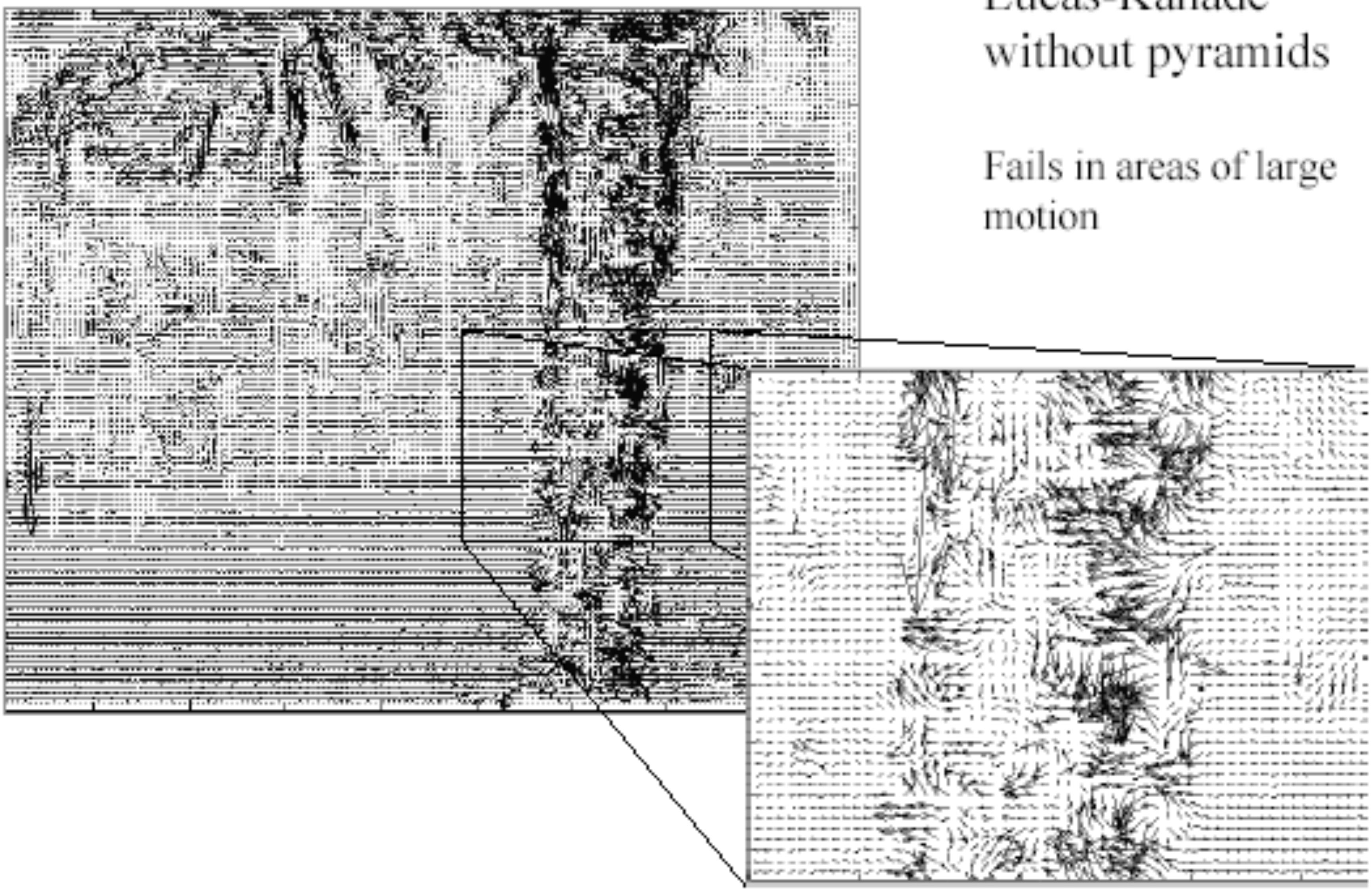
Edge

- gradients have one dominant direction
- large λ_1, small λ_2
- system is ill-conditioned

High-texture or corner region

– gradients have different directions, large magnitudes
– large λ_1, large λ_2
– system is well-conditioned

Optical Flow Results



Lucas-Kanade without pyramids

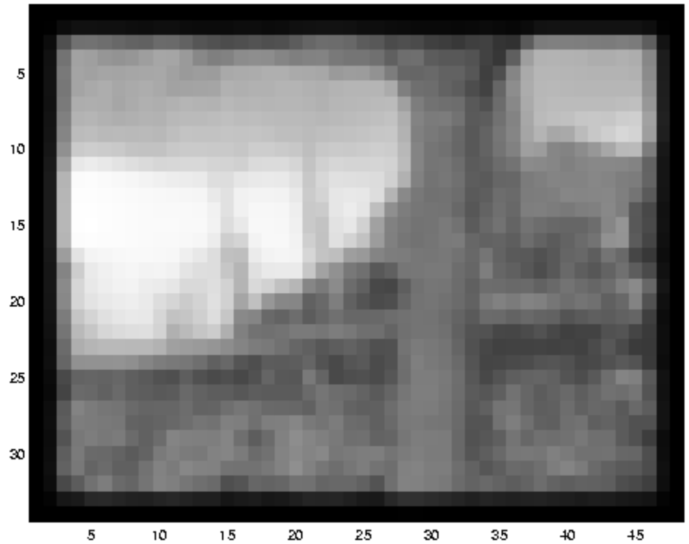
Fails in areas of large motion

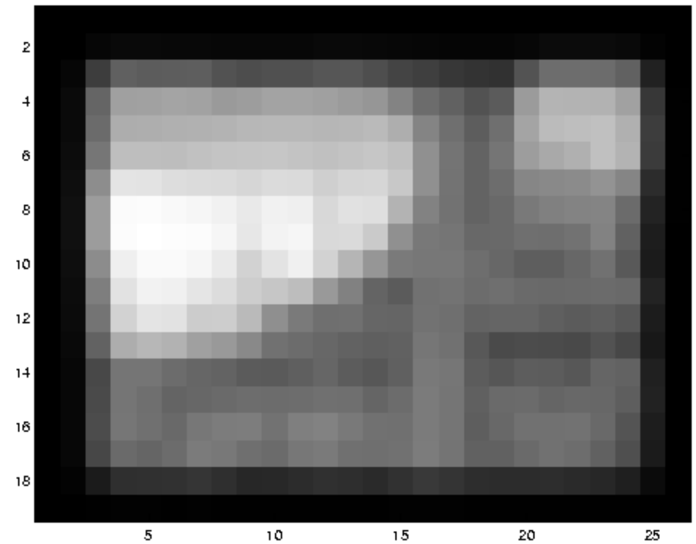
* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Errors in Lucas-Kanade

- The motion is large (larger than a pixel)
  - Iterative refinement
  - Coarse-to-fine estimation
  - Exhaustive neighborhood search (feature matching)
- A point does not move like its neighbors
  - Motion segmentation
- Brightness constancy does not hold
  - Exhaustive neighborhood search with normalized correlation

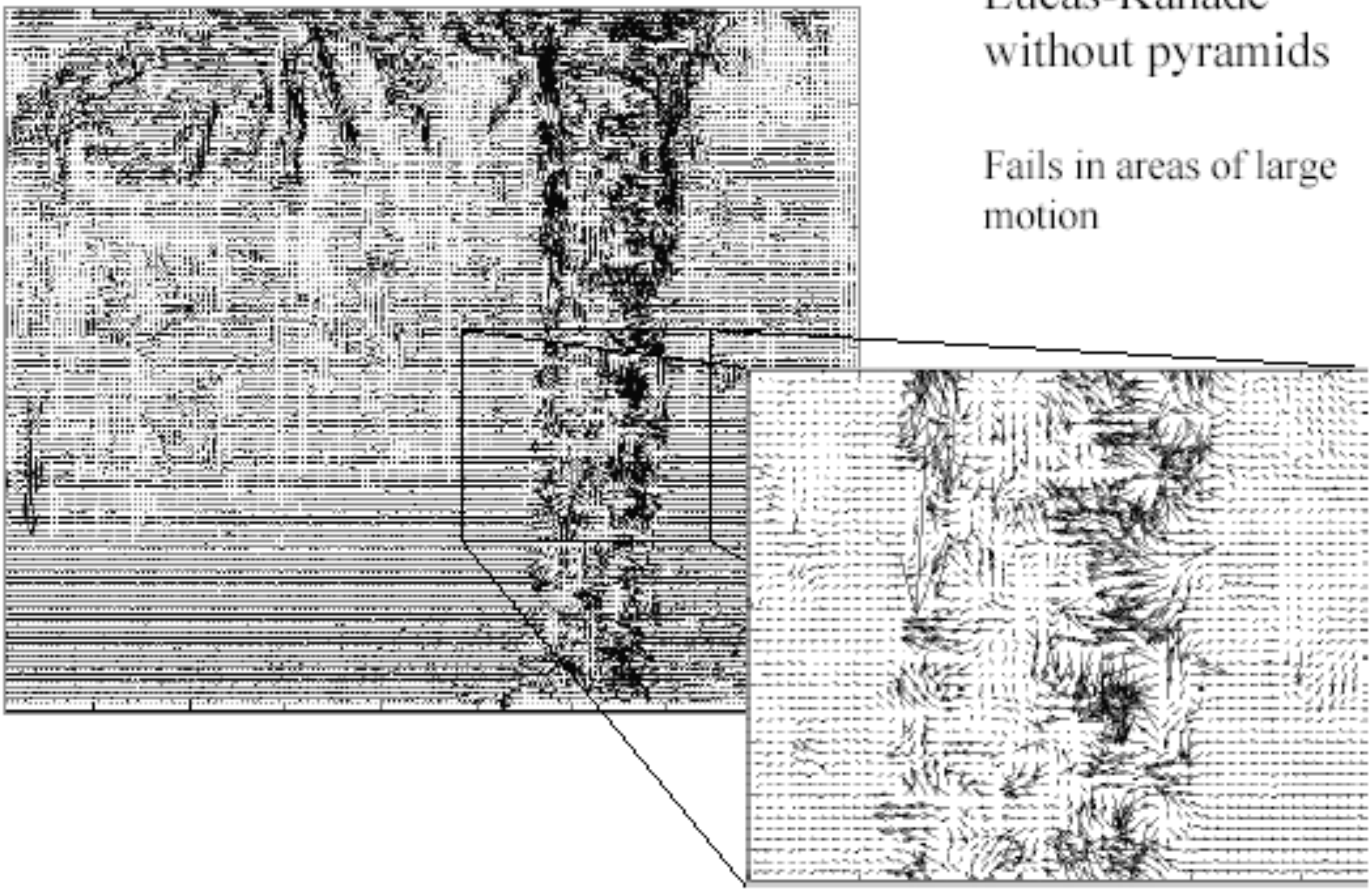
Multi-resolution registration





*** From Khurram Hassan-Shafique CAP5415 Computer Vision 2003**

Optical flow results

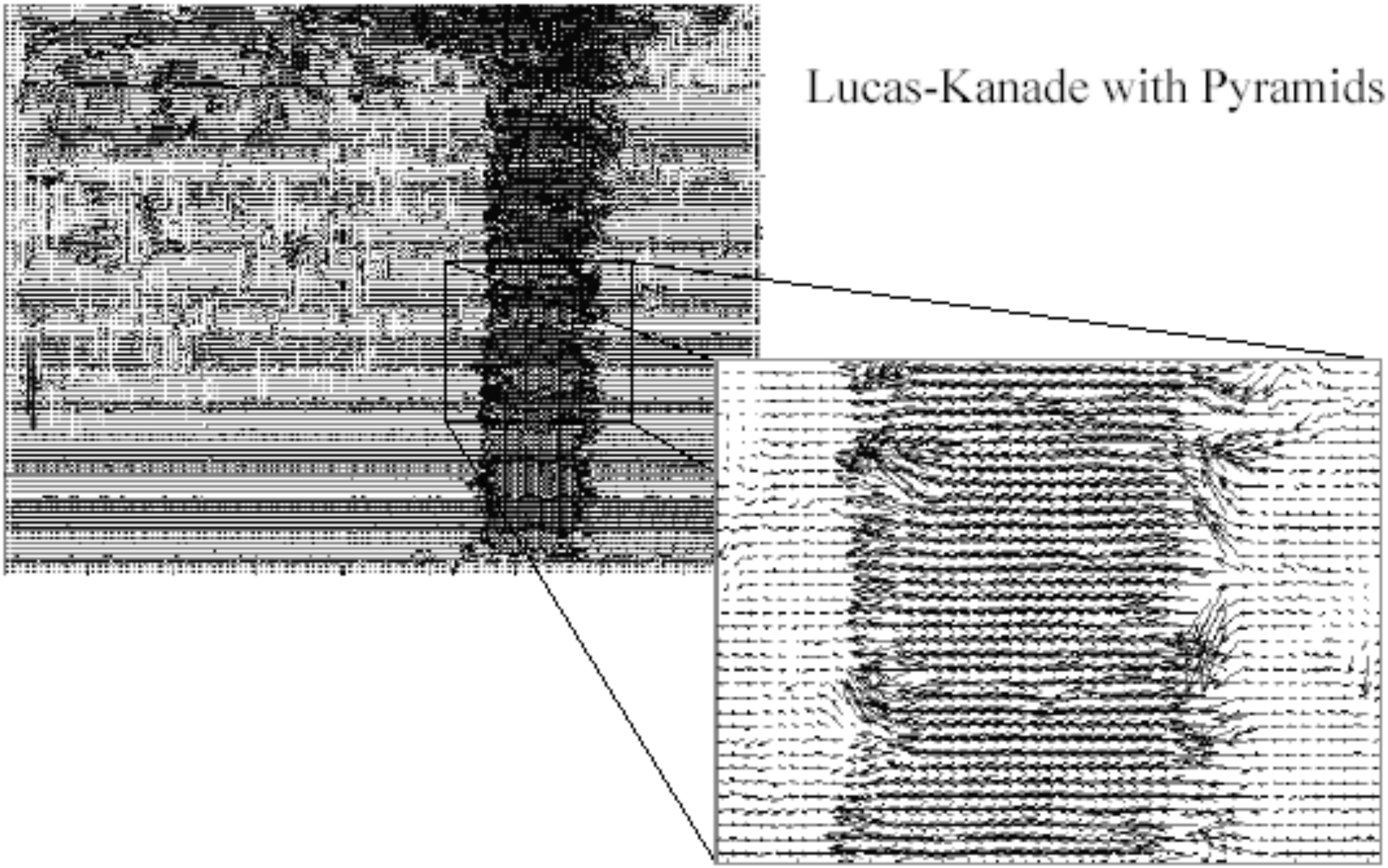


Lucas-Kanade without pyramids

Fails in areas of large motion

*** From Khurram Hassan-Shafique CAP5415 Computer Vision 2003**

Optical flow results



* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

State-of-the-art optical flow

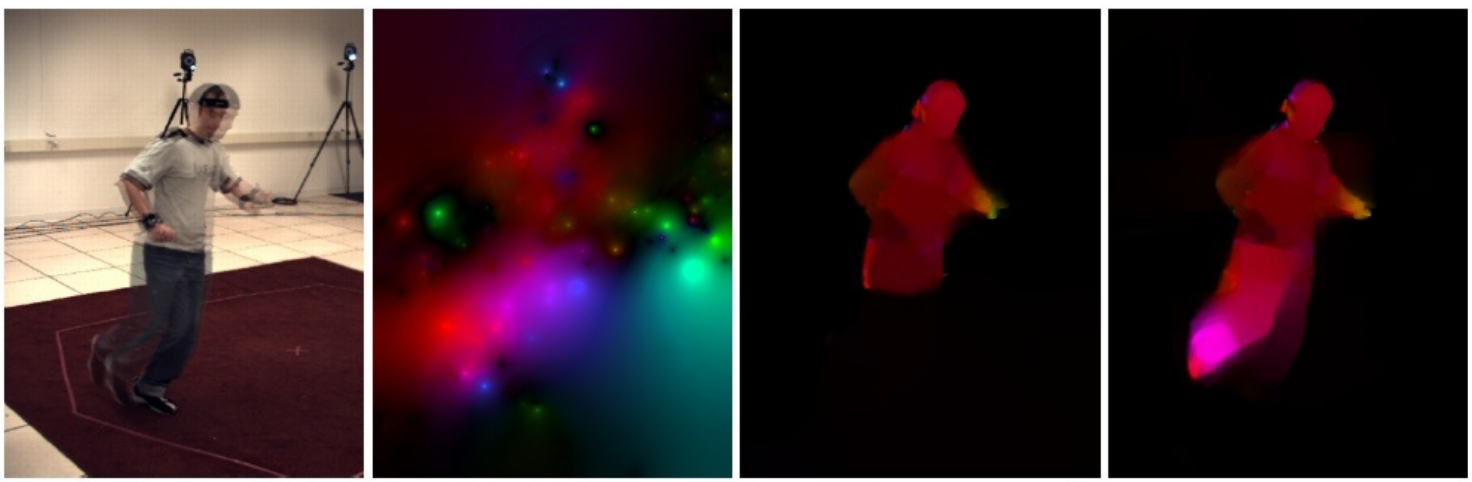
Start with something similar to Lucas-Kanade

+ gradient constancy

+ energy minimization with smoothing term

+ region matching

+ keypoint matching (long-range)



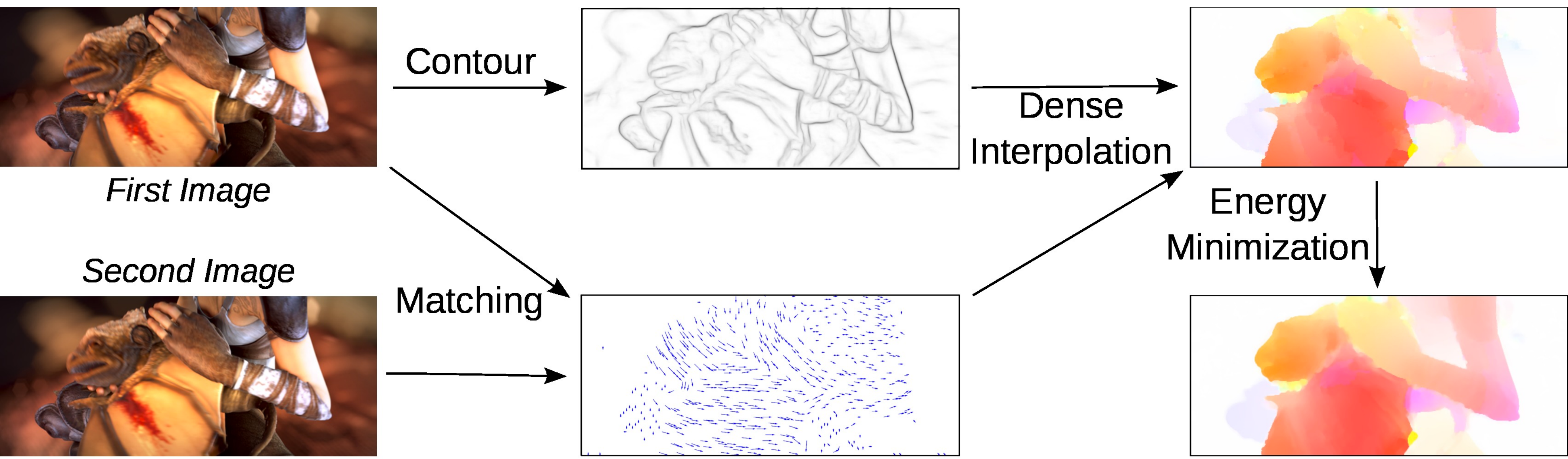
Region-based **+Pixel-based** **+Keypoint-based**

Large displacement optical flow, Brox et al., CVPR 2009

Source: J. Hays

State-of-the-art optical flow

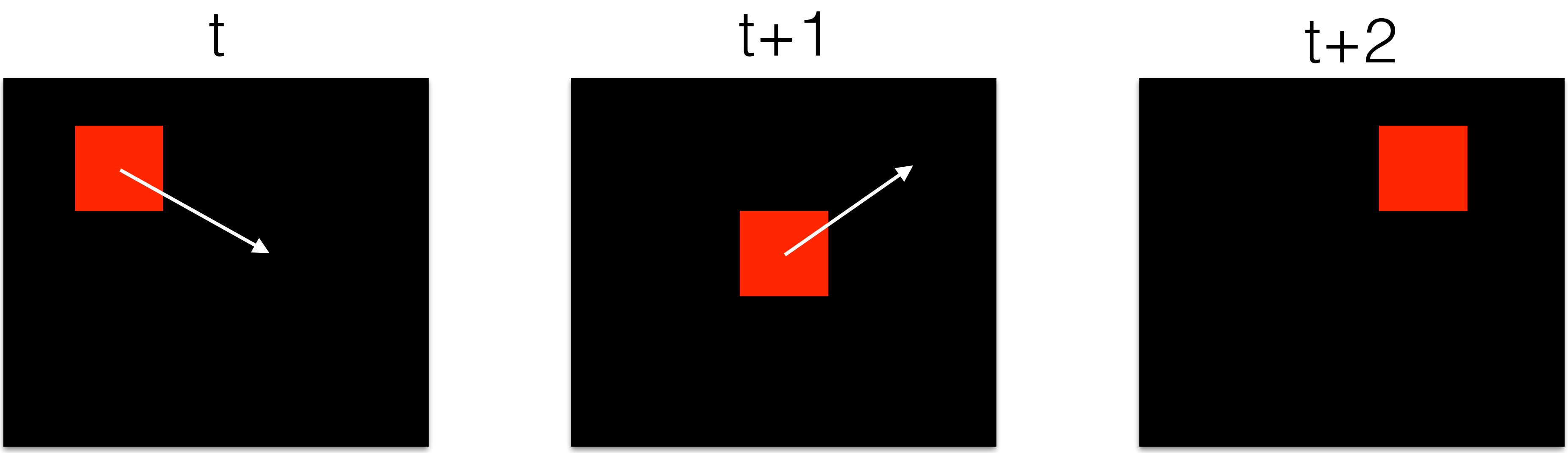
- Epic Flow: Feature matching + edge preserving flow interpolation



EpicFlow: Edge-Preserving Interpolation of Correspondences for Optical Flow, Jerome Revaud, Philippe Weinzaepfel, Zaid Harchaoui and Cordelia Schmid, CVPR 2015.

Feature tracking

- So far, we have only considered optical flow estimation in a pair of images
- If we have more than two images, we can compute the optical flow from each frame to the next
- Given a point in the first image, we can in principle reconstruct its path by simply “following the arrows”



Tracking challenges

- Ambiguity of optical flow
  - Need to find good features to track
- Large motions, changes in appearance, occlusions, disocclusions
  - Need mechanism for deleting, adding new features
- Drift – errors may accumulate over time
  - Need to know when to terminate a track

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.

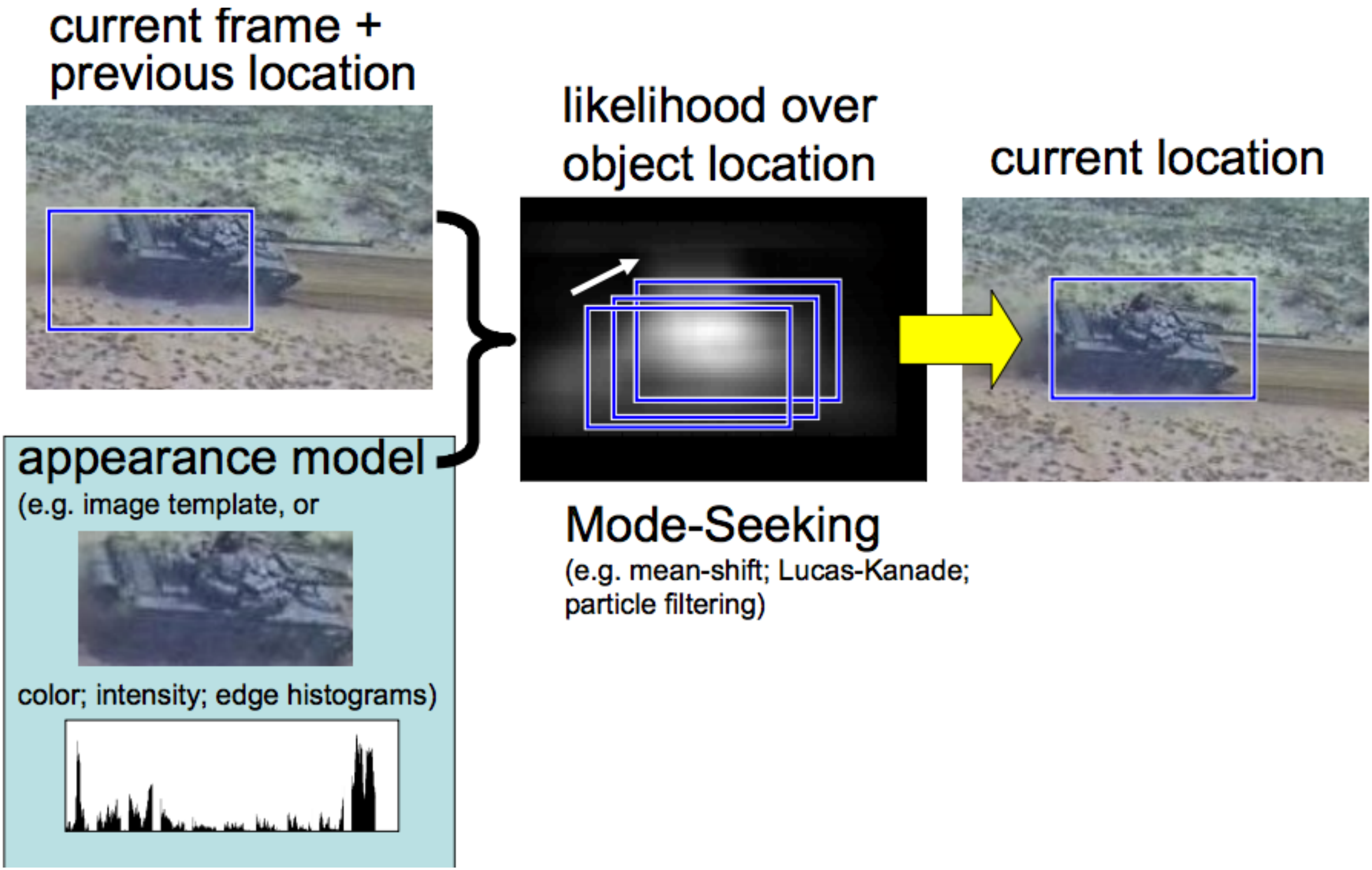
Tracking example

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.

Tracking by matching



Tracking example

https://www.youtube.com/watch?v=RG5uV_h50b0