CAP5415-Computer Vision
Lecture 7-Optical Flow

Ulas Bagci

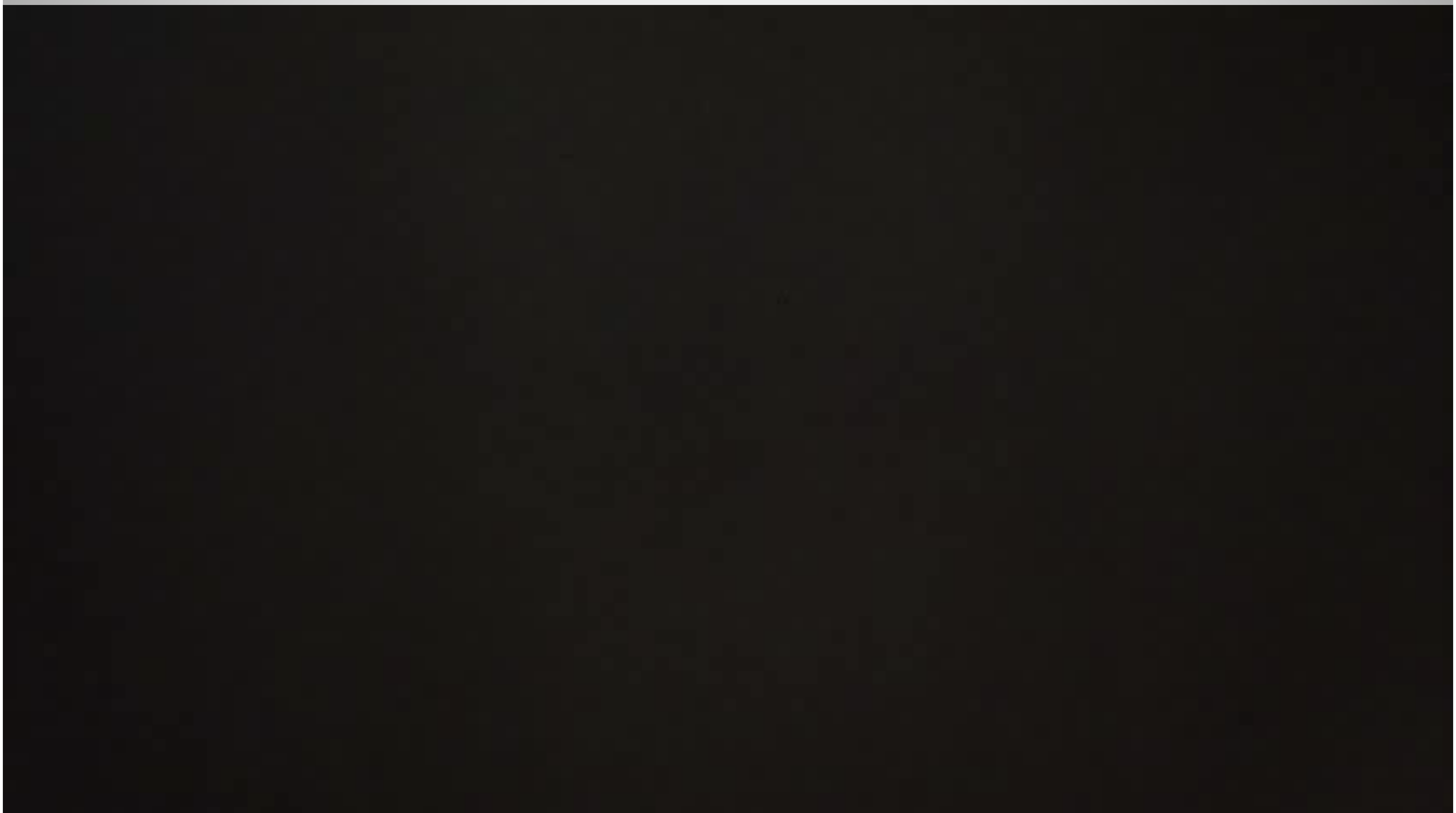
bagci@ucf.edu

Readings

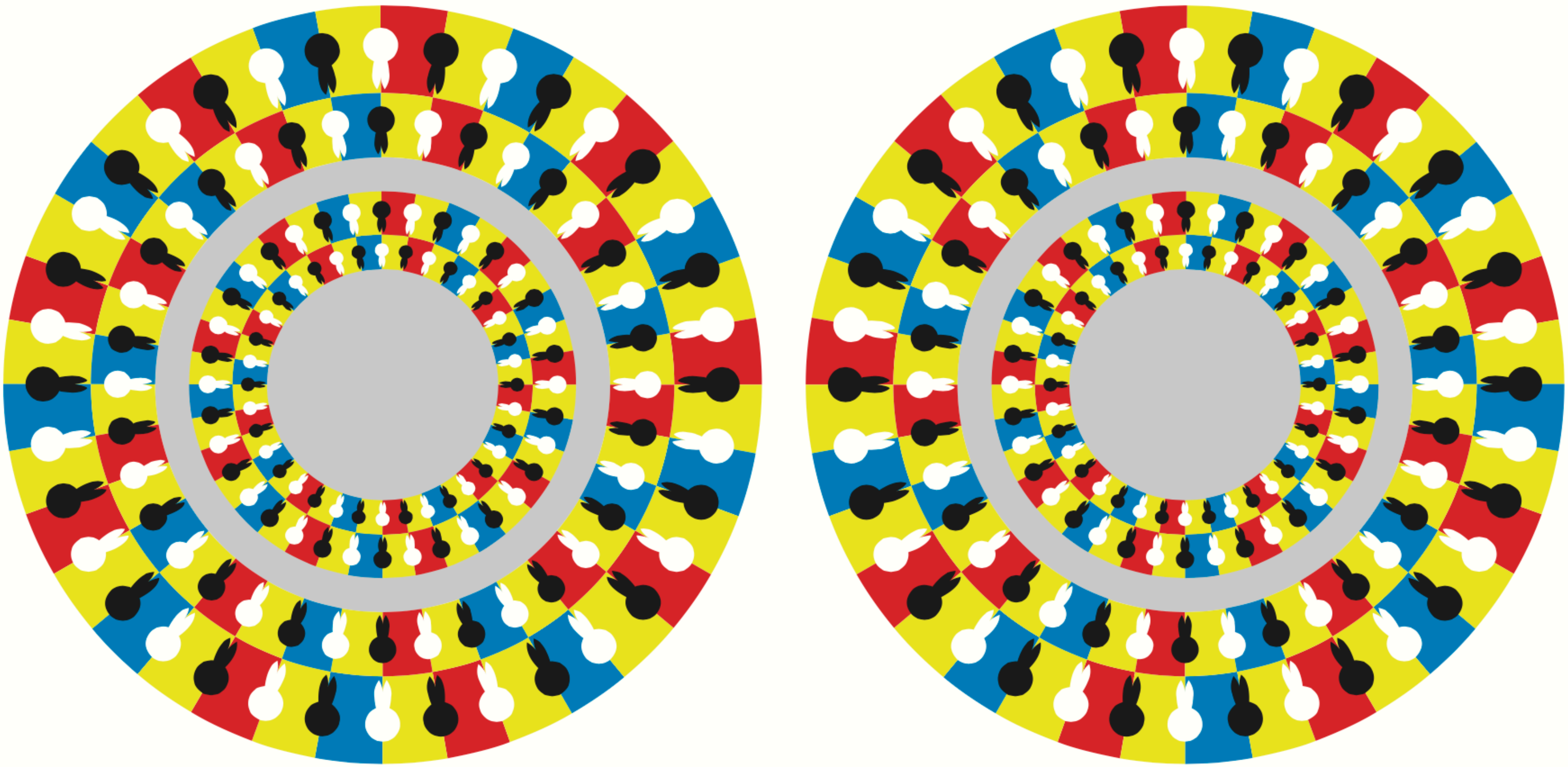
- Szeliski, R. Ch. 7
- Bergen et al. ECCV 92, pp. 237-252.
- Shi, J. and Tomasi, C. CVPR 94, pp.593-600.
- Baker, S. and Matthews, I. IJCV 2004, pp. 221-255.

- Slide Credits: Szeliski and Shah.
- New York Times
- Video from Michael Black

Seeing the visible in the invisible



Motion



U-zu-maki Glasses=> U: rabbits, zu: figure, maki: rotation

Reasons?

- the patterns only move when you blink or move your eyes
- the arrangement of the backgrounds of the ‘rabbits’ determines which way the patterns rotate.

Motion

Perception of change in the outside world is of the highest importance for all living beings.

A great majority of **changes** in sensory excitation in the image result from **motion**.

Images also change very markedly when a light source is turned on or off-but this change cannot be interpreted as motion.

Why Estimate Visual Motion?

- Visual Motion can be due to problems
  - Camera instabilities, jitter

Why Estimate Visual Motion?

- Visual Motion can be due to problems
  - Camera instabilities, jitter
- Visual Motion Indicates dynamics in the scene
  - Moving objects, behavior, tracking objects, analyze trajectories

Why Estimate Visual Motion?

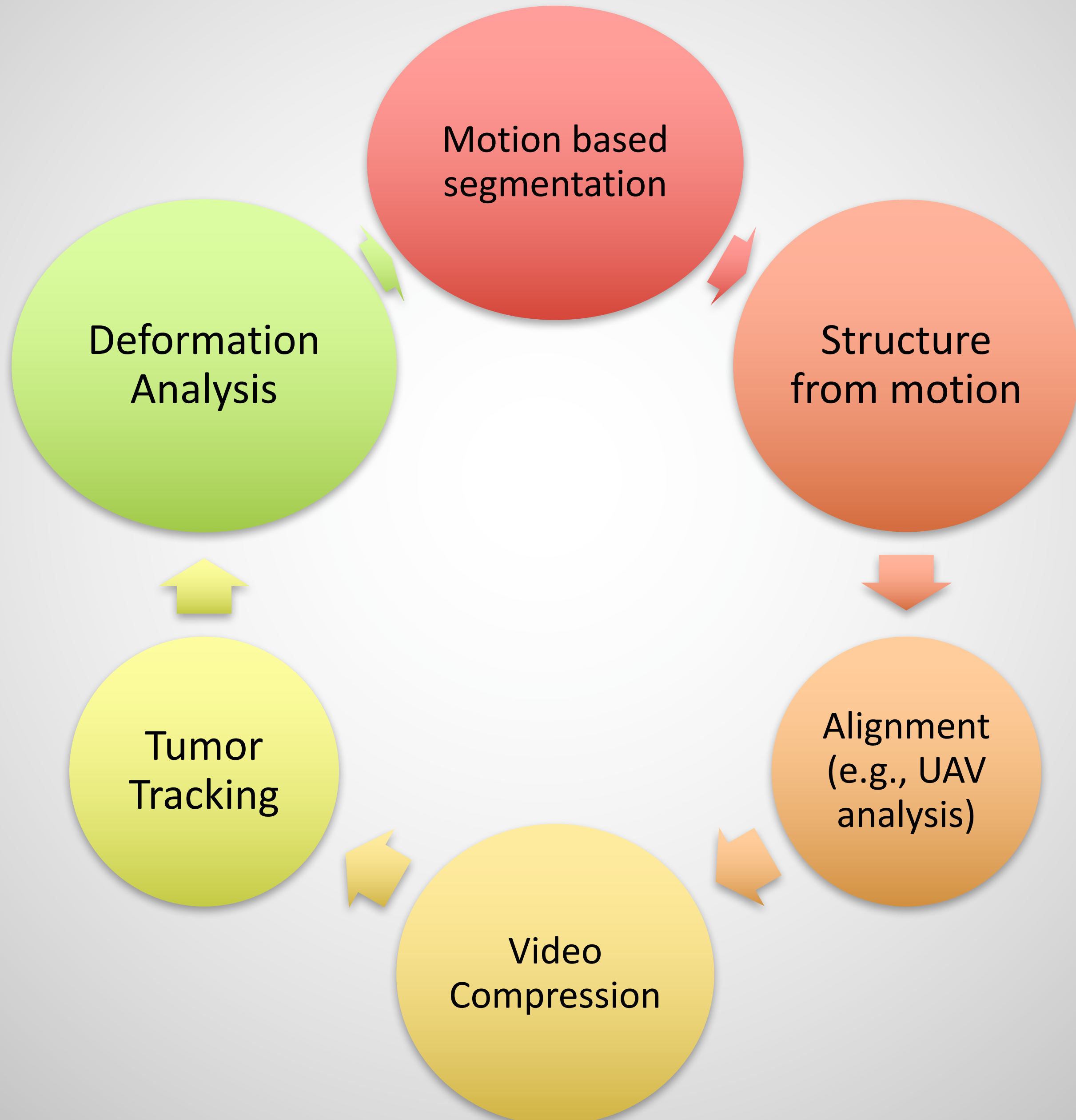
- Visual Motion can be due to problems
  - Camera instabilities, jitter
- Visual Motion Indicates dynamics in the scene
  - Moving objects, behavior, tracking objects, analyze trajectories
- Visual Motion reveals spatial layout
  - Motion parallax

Today's Lecture

Visual Motion Estimation

- Patch-based motion
  - Optical Flow
    - Lucas-Kanade
    - Horn-Schunck
- Motion Models

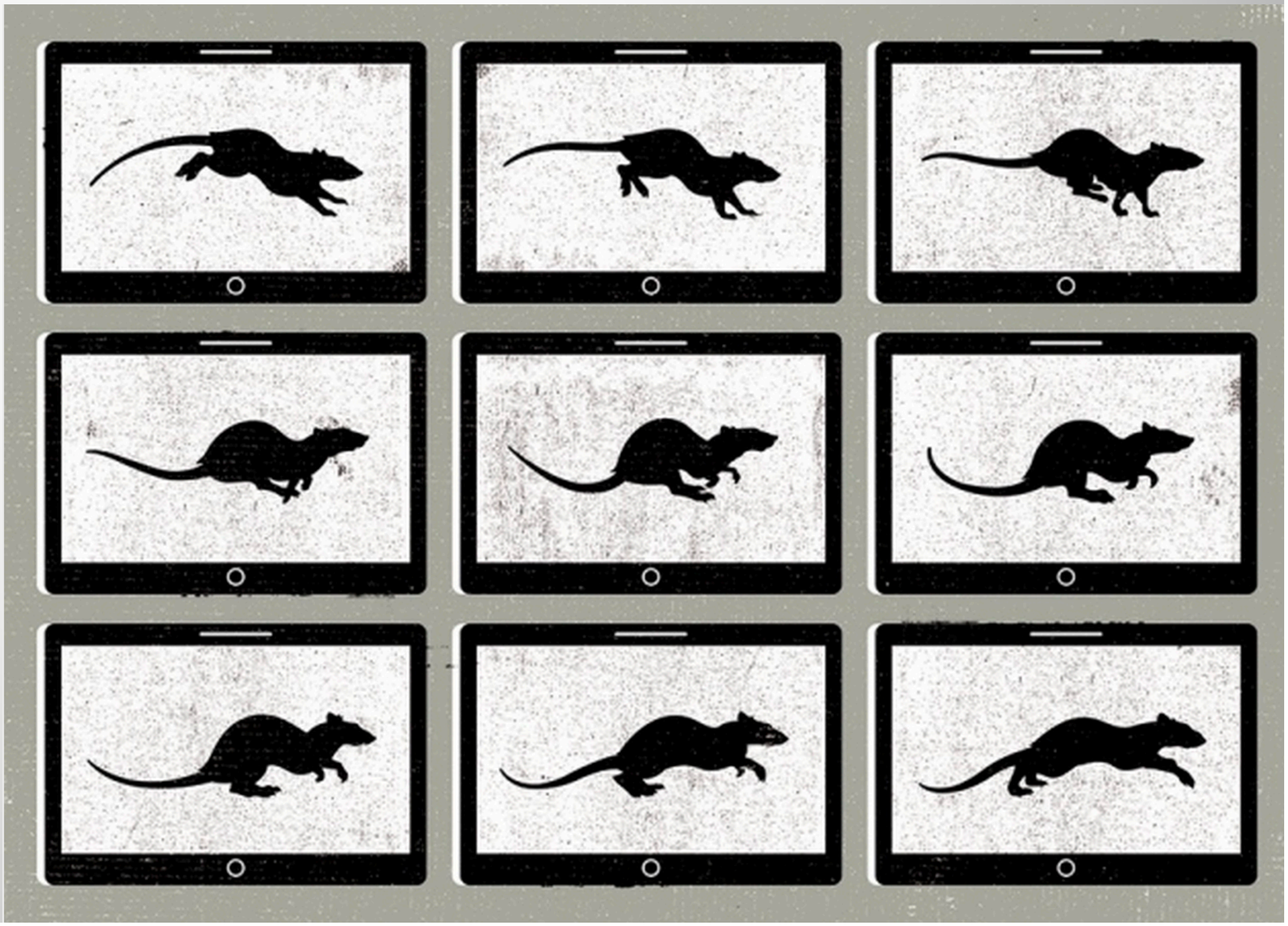
Applications

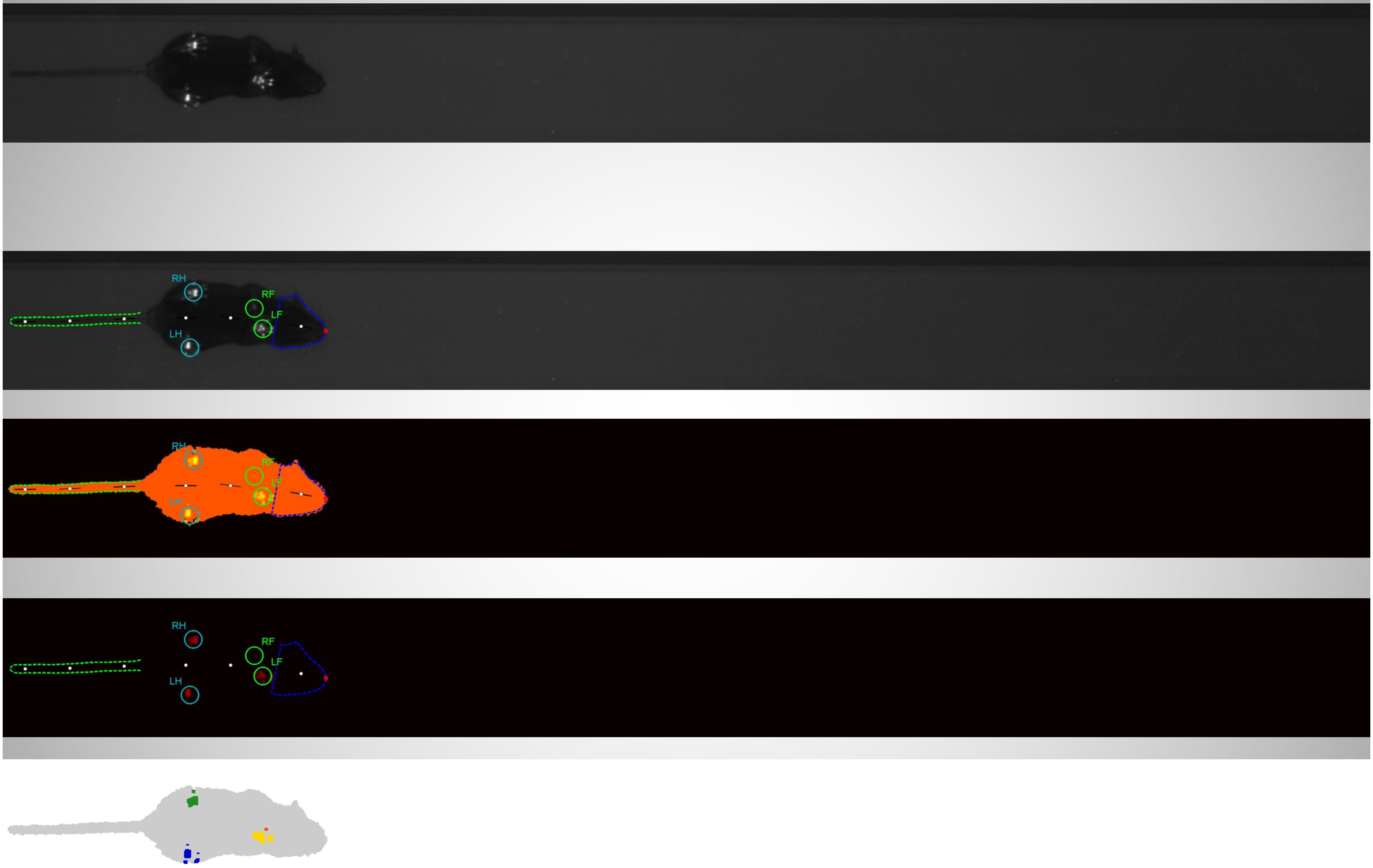


Nature News-Vol 525, Issue 7567

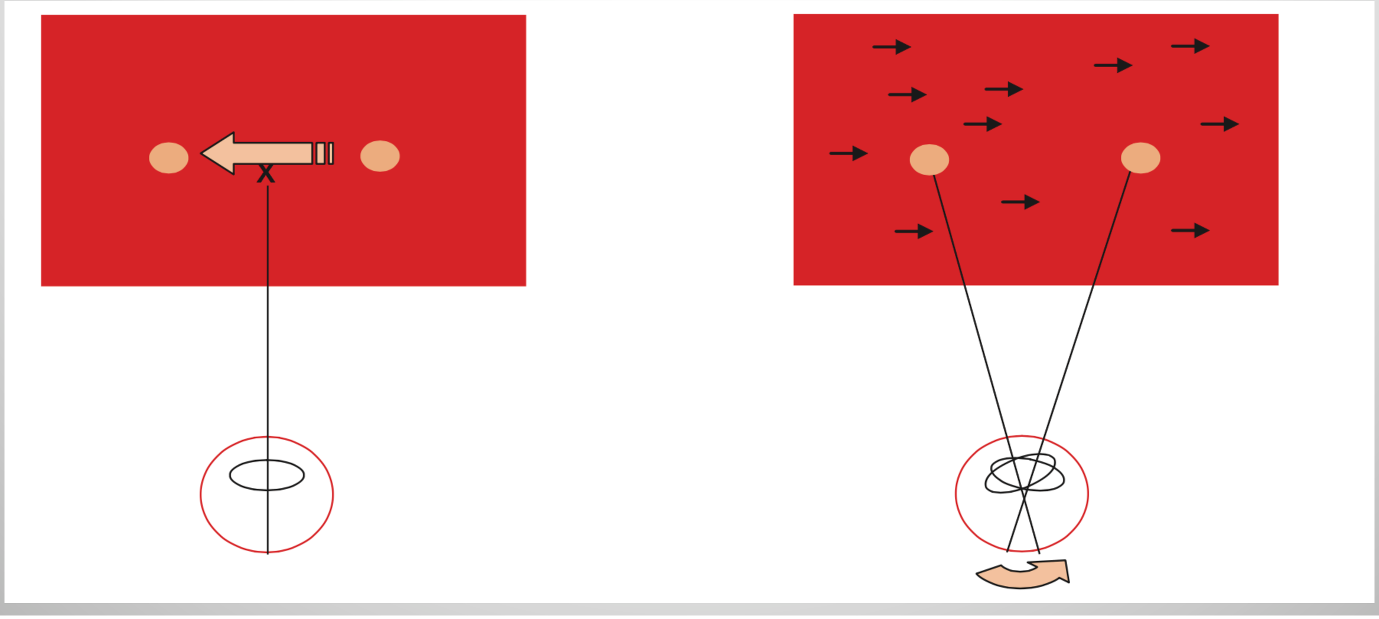
- CV Tools that track how animals move are helping researchers to do everything from diagnosing neurological conditions to illuminating evolution

Higher-speed cameras eventually improved what could be captured. But movement studies still needed a person to look through the results frame by frame, laboriously tracing the arc of each step, arm swing or wing flap to extract information about angles and forces.



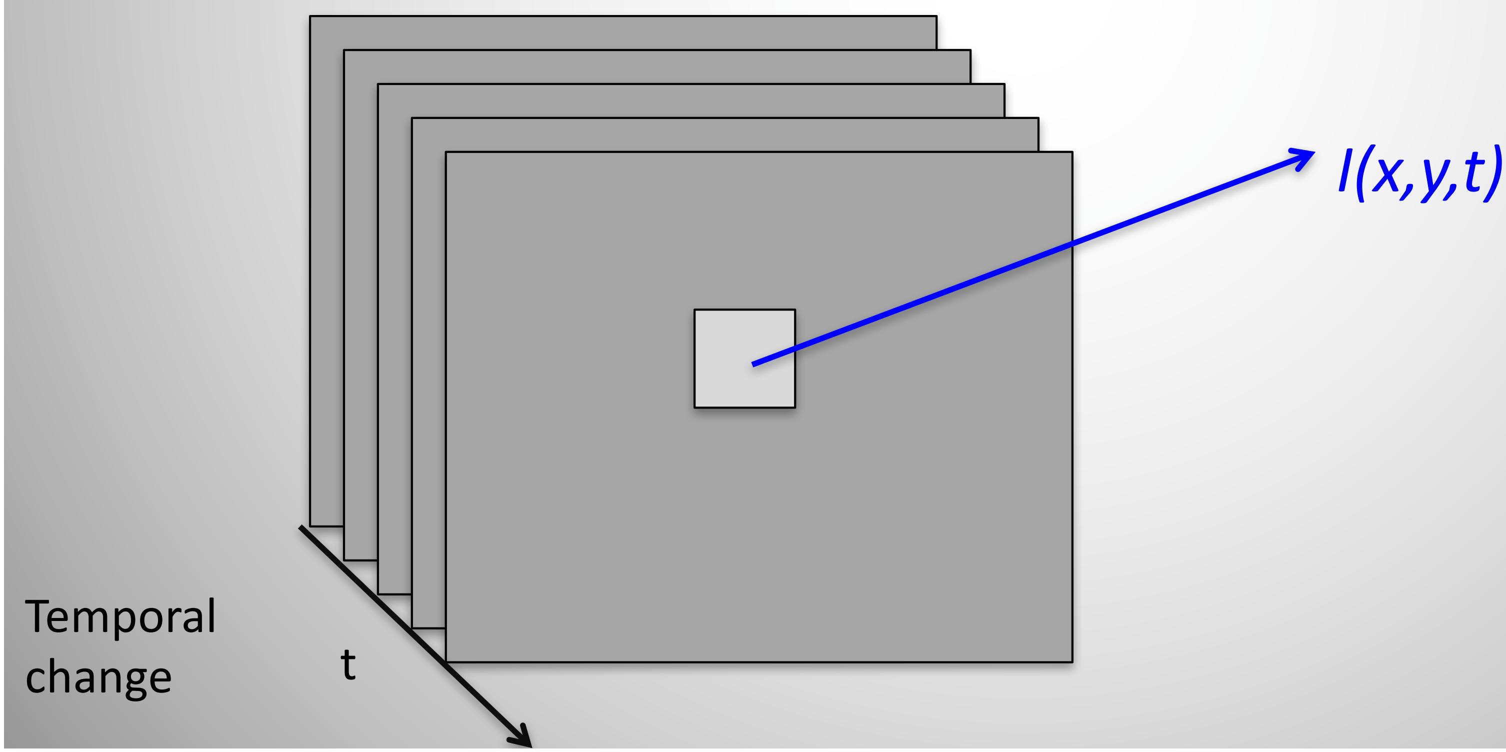

RH
RF
LF
LH
RH
RF
LF
LH
RH
RF
LF
LH

Perception of Motion

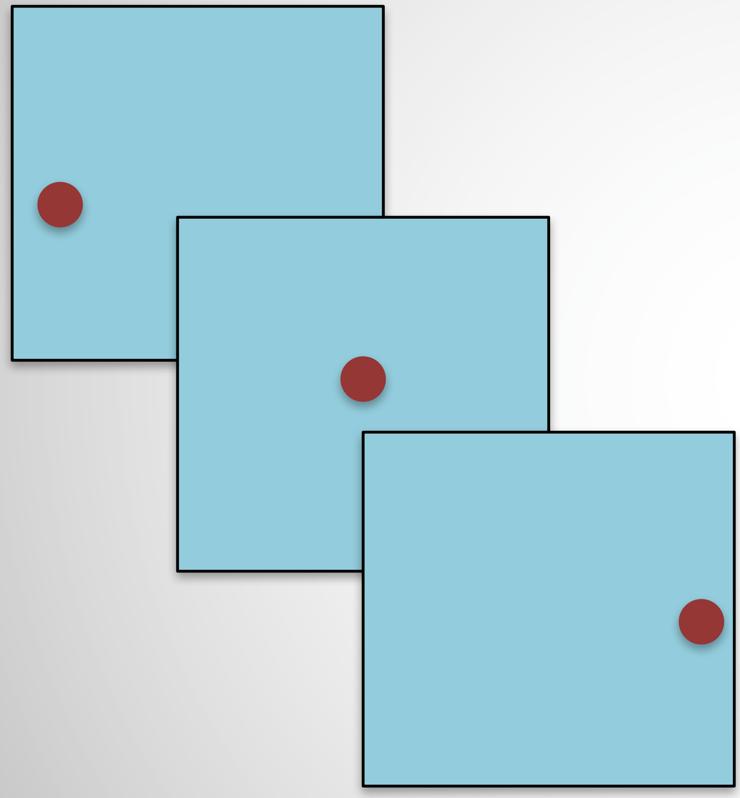


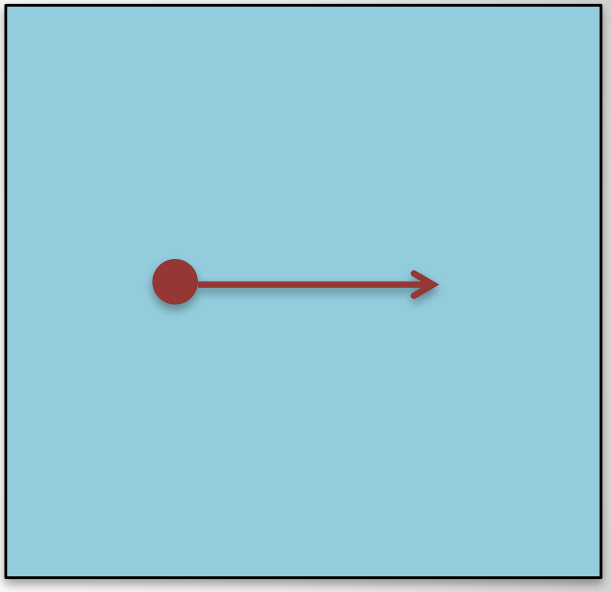
Video

- A video is a sequence of frames captured over time
  - Image data is a function of space (x,y) and time (t)



Apparent Motion





Present dots on three movie frames

See one moving dot

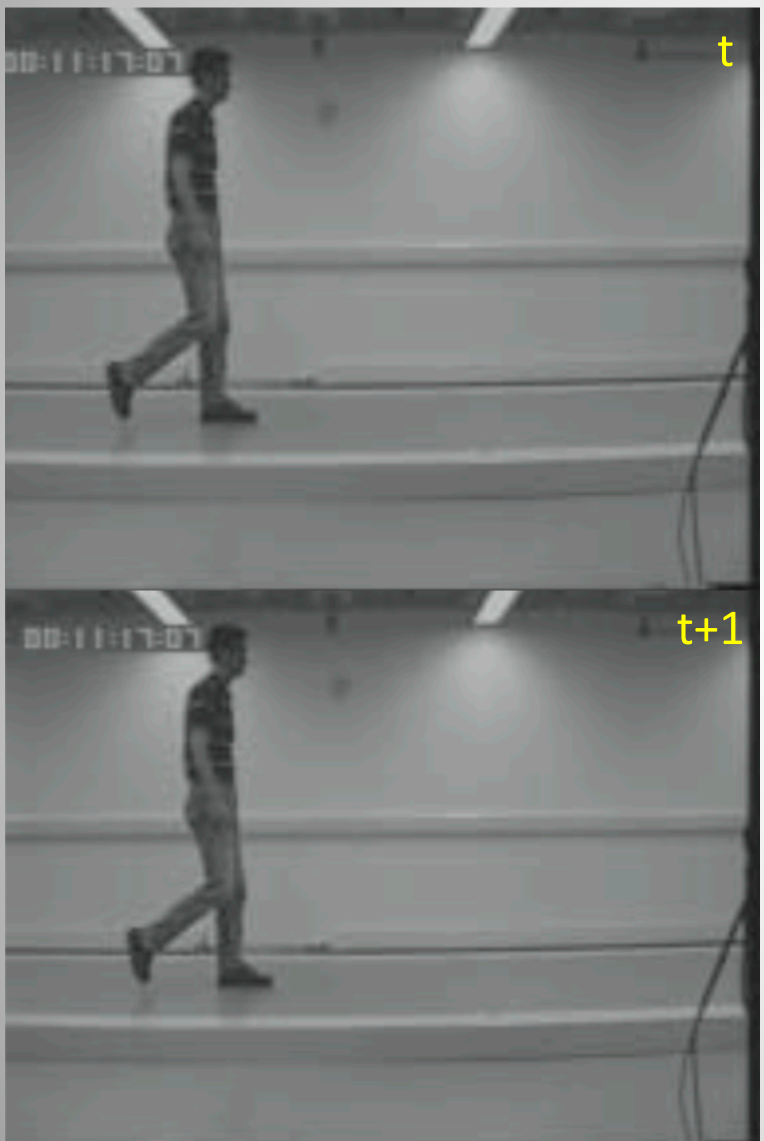
Describing Motion

Describing Motion

- Simplest way: Image Differencing (intensity values)

Describing Motion

- Simplest way: Image Differencing (intensity values)



Optical Flow

- Refers to the problem of estimating a vector field of local displacement in a sequence of images.

Optical Flow

- Refers to the problem of estimating a vector field of local displacement in a sequence of images.
- When we fix our attention to a single point and measure velocities flowing through that location, then the problem is called *optical flow.*

Optical Flow

- Refers to the problem of estimating a vector field of local displacement in a sequence of images.
- When we fix our attention to a single point and measure velocities flowing through that location, then the problem is called *optical flow.*
  - *Stereo matching, image matching, tracking,…*

Estimating Optical Flow

- Assume the image intensity I is constant

Time = *t*

Time = *t+dt*

$$I(x,y,t) = I(x+dx,y+dy,t+dt)$$

Estimating Optical Flow

$$I(x, y, t) \simeq I(x + dx, y + dy, t + dt)$$

Estimating Optical Flow

$$I(x, y, t) \simeq I(x + dx, y + dy, t + dt)$$

$$I(x(t) + u.\Delta t, y(t) + v.\Delta t) - I(x(t), y(t), t) \approx 0$$

Estimating Optical Flow

$$I(x,y,t) \simeq I(x+dx, y+dy, t+dt)$$

$$I(x(t)+u.\Delta t, y(t)+v.\Delta t) - I(x(t), y(t), t) \approx 0$$

Assuming I is differentiable function, and expand the first term using Taylor's series:

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

Estimating Optical Flow

$$I(x,y,t) \simeq I(x+dx, y+dy, t+dt)$$

$$I(x(t) + u.\Delta t, y(t) + v.\Delta t) - I(x(t), y(t), t) \approx$$

Assuming I is differentiable function, and expand the first term using Taylor's series:

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

Compact representation

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

Brightness constancy constraint

Assumption: The Brightness Constraint

- Expresses the idea of similar brightness for the same objects observed in a sequence

Assumption: The Brightness Constraint

- Expresses the idea of similar brightness for the same objects observed in a sequence
- When we follow with a given location and trace their position in consecutive images of a sequence, then the problem is called "feature tracking"

Assumption: The Brightness Constraint

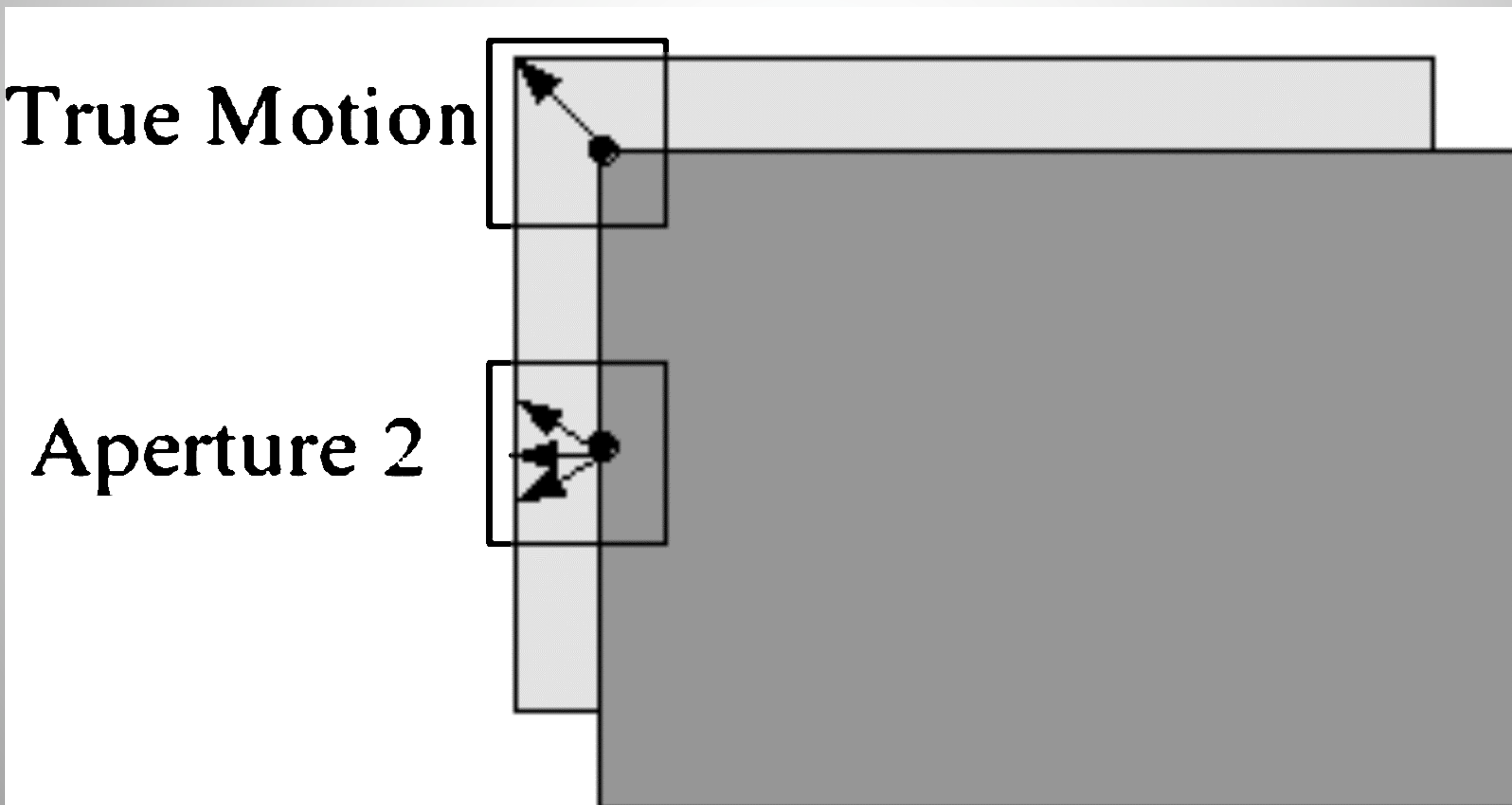
- Expresses the idea of similar brightness for the same objects observed in a sequence
- When we follow with a given location and trace their position in consecutive images of a sequence, then the problem is called "feature tracking"
- Trying to solve a single equation for two variables (u and v) →ill-posed

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

Aperture Problem

Aperture Problem

- How certain are the motion estimates?



Second Assumption: Gradient Constraint

Velocity vector is constant within a small neighborhood **(LUCAS AND KANADE)**

Second Assumption: Gradient Constraint

Velocity vector is constant within a small neighborhood **(LUCAS AND KANADE)**

$$E(u,v) = \int_{x,y} (I_x u + I_y v + I_t)^2 dxdy$$

Second Assumption: Gradient Constraint

Velocity vector is constant within a small neighborhood **(LUCAS AND KANADE)**

$$E(u,v) = \int_{x,y} (I_x u + I_y v + I_t)^2 dxdy$$

$$\frac{\partial E(u,v)}{\partial u} = \frac{\partial E(u,v)}{\partial v} = 0$$

Second Assumption: Gradient Constraint

Velocity vector is constant within a small neighborhood **(LUCAS AND KANADE)**

$$E(u,v) = \int_{x,y} (I_x u + I_y v + I_t)^2 dxdy$$

$$\frac{\partial E(u,v)}{\partial u} = \frac{\partial E(u,v)}{\partial v} = 0$$

$$2(I_x u + I_y v + I_t) I_x = 0$$

$$2(I_x u + I_y v + I_t) I_y = 0$$

Lucas-Kanade

$$\begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} \sum I_x I_t \\ \sum I_y I_t \end{bmatrix}$$

Lucas-Kanade

$$\begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} \sum I_x I_t \\ \sum I_y I_t \end{bmatrix}$$

Structural Tensor representation

$$\begin{bmatrix} T_{xx} & T_{xy} \\ T_{xy} & T_{yy} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} T_{xt} \\ T_{yt} \end{bmatrix}$$

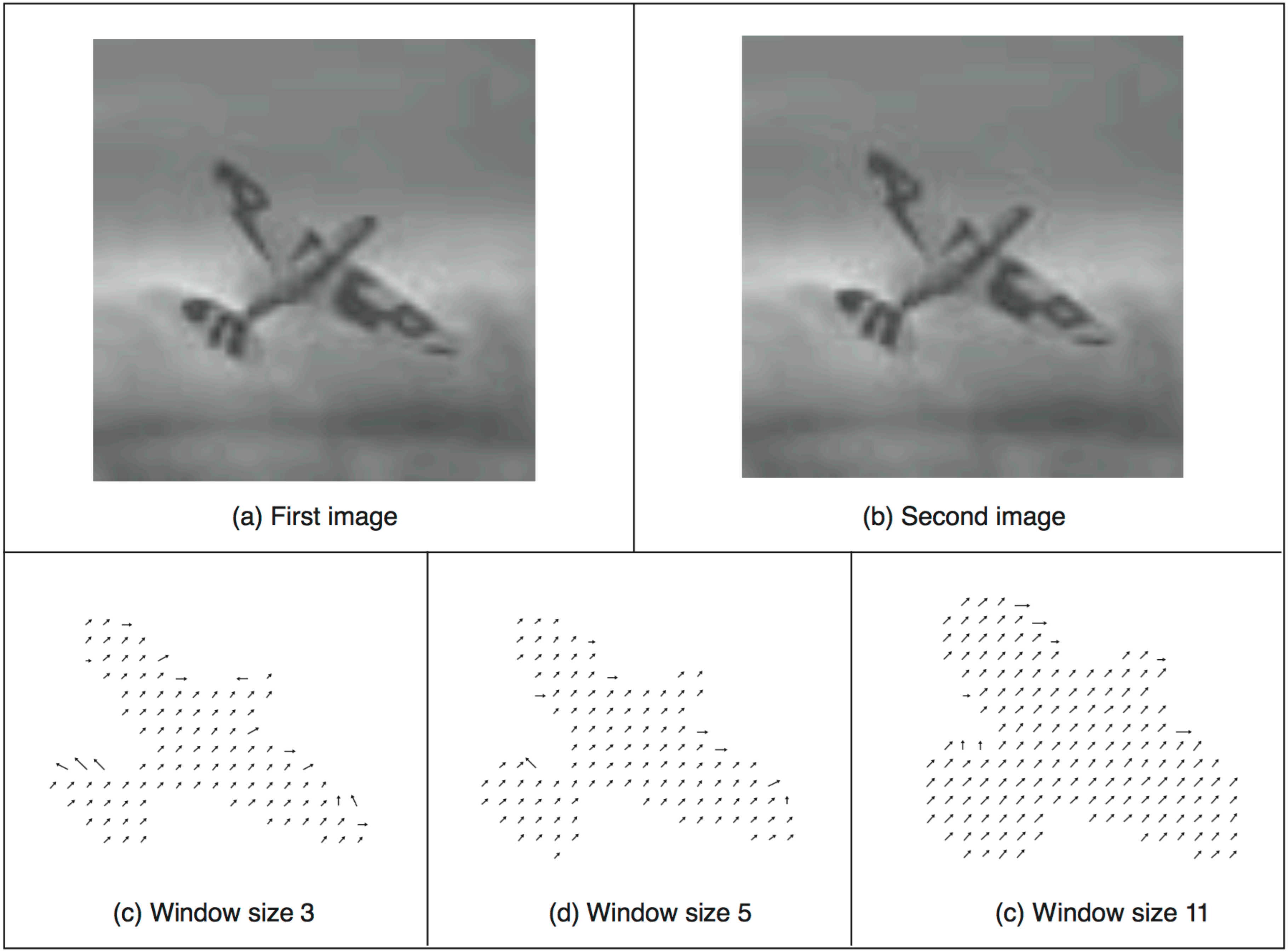
Lucas-Kanade

$$\begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} \sum I_x I_t \\ \sum I_y I_t \end{bmatrix}$$

Structural Tensor representation

$$\begin{bmatrix} T_{xx} & T_{xy} \\ T_{xy} & T_{yy} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} T_{xt} \\ T_{yt} \end{bmatrix}$$

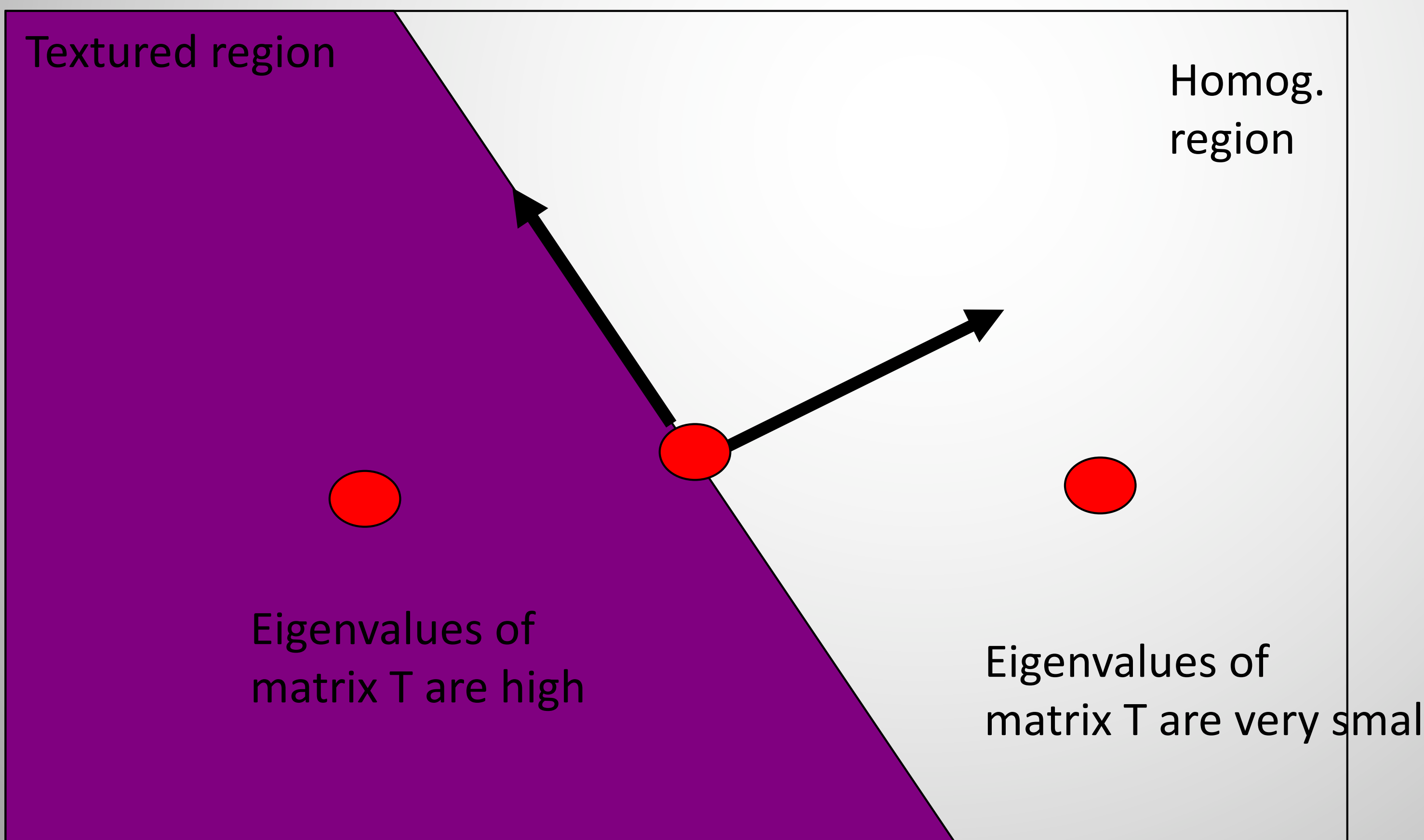
$$u = \frac{T_{yt}T_{xy} - T_{xt}T_{yy}}{T_{xx}T_{yy} - T_{xy}^2} \text{ and } v = \frac{T_{xt}T_{xy} - T_{yt}T_{xx}}{T_{xx}T_{yy} - T_{xy}^2}$$


(a) First image
(b) Second image
(c) Window size 3
(d) Window size 5
(c) Window size 11

How does Lucas-Kanade behave?

$$\begin{bmatrix} T_{xx} & T_{xy} \\ T_{xy} & T_{yy} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} T_{xt} \\ T_{yt} \end{bmatrix}$$

At edges, matrix T becomes singular!



Pitfalls & Alternatives

- Brightness constancy is not satisfied
  - Correlation based method could be used

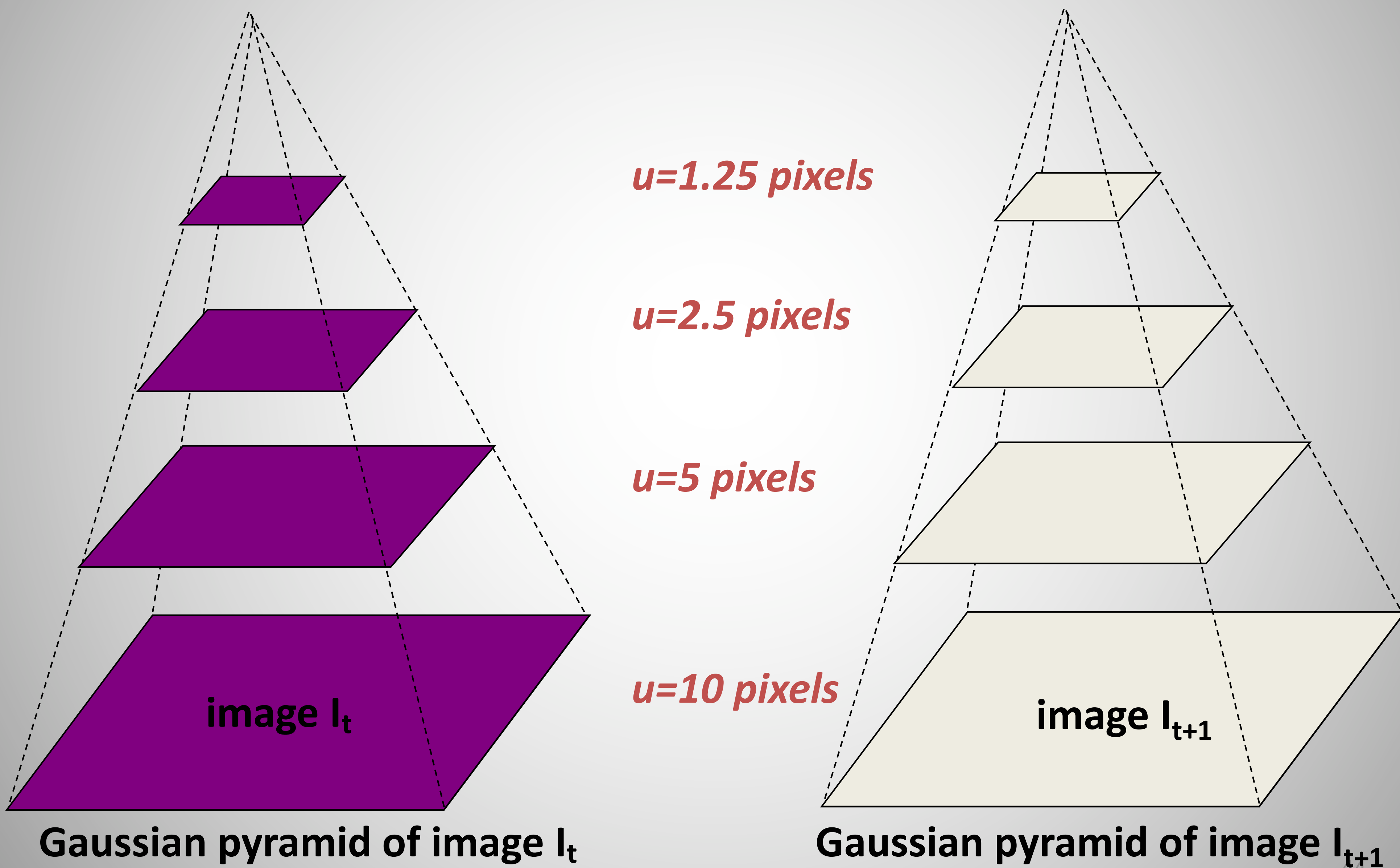
Pitfalls & Alternatives

- Brightness constancy is not satisfied
  - Correlation based method could be used
- A point may not move like its neighbors
  - Regularization based methods

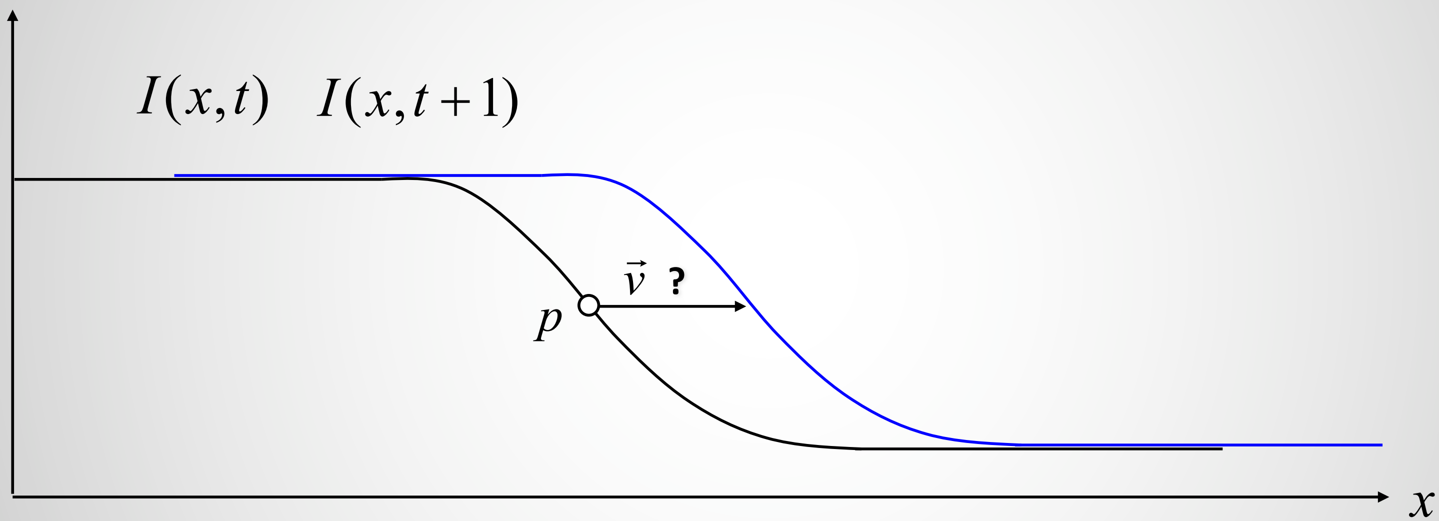
Pitfalls & Alternatives

- Brightness constancy is not satisfied
  - Correlation based method could be used
- A point may not move like its neighbors
  - Regularization based methods
- The motion may not be small (Taylor does not hold!)
  - Multi-scale estimation could be used

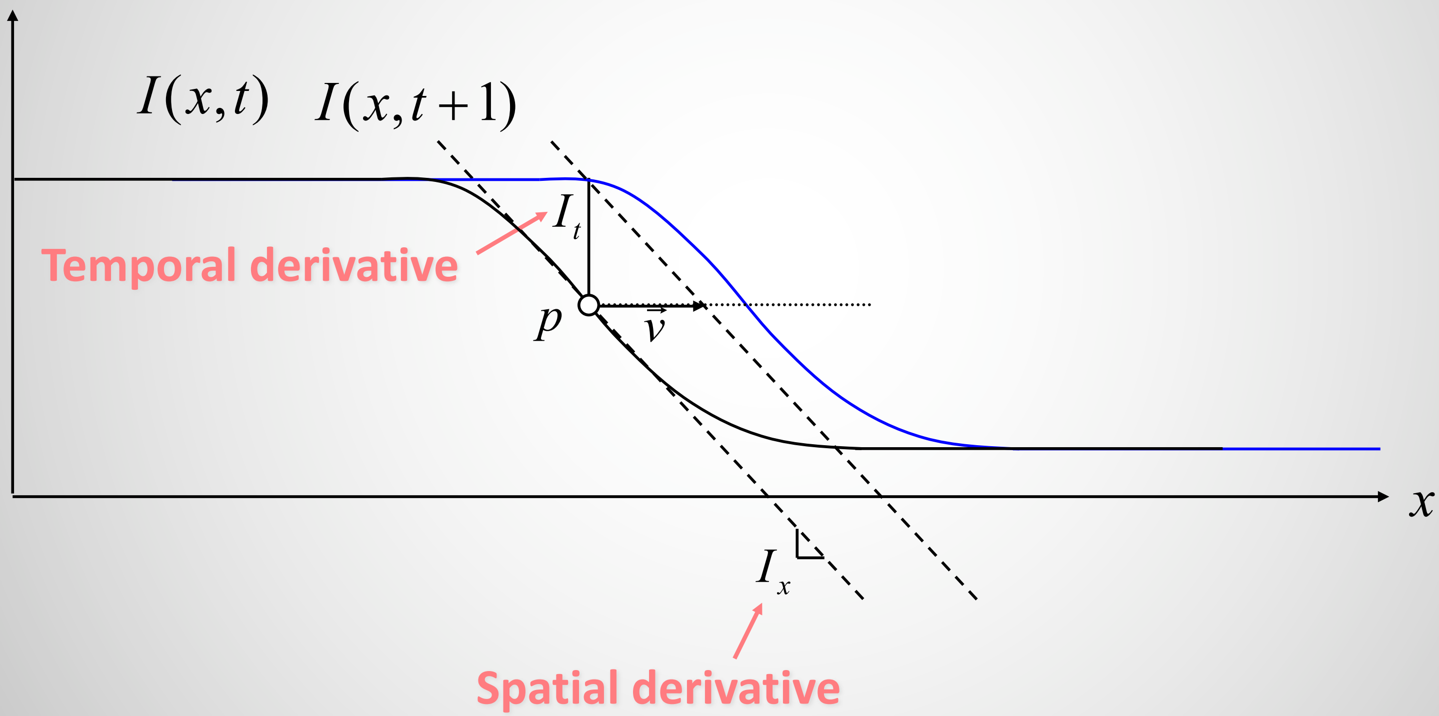
Multi-Scale Flow Estimation



Example: Optical Flow in 1D



Example: Optical Flow in 1D



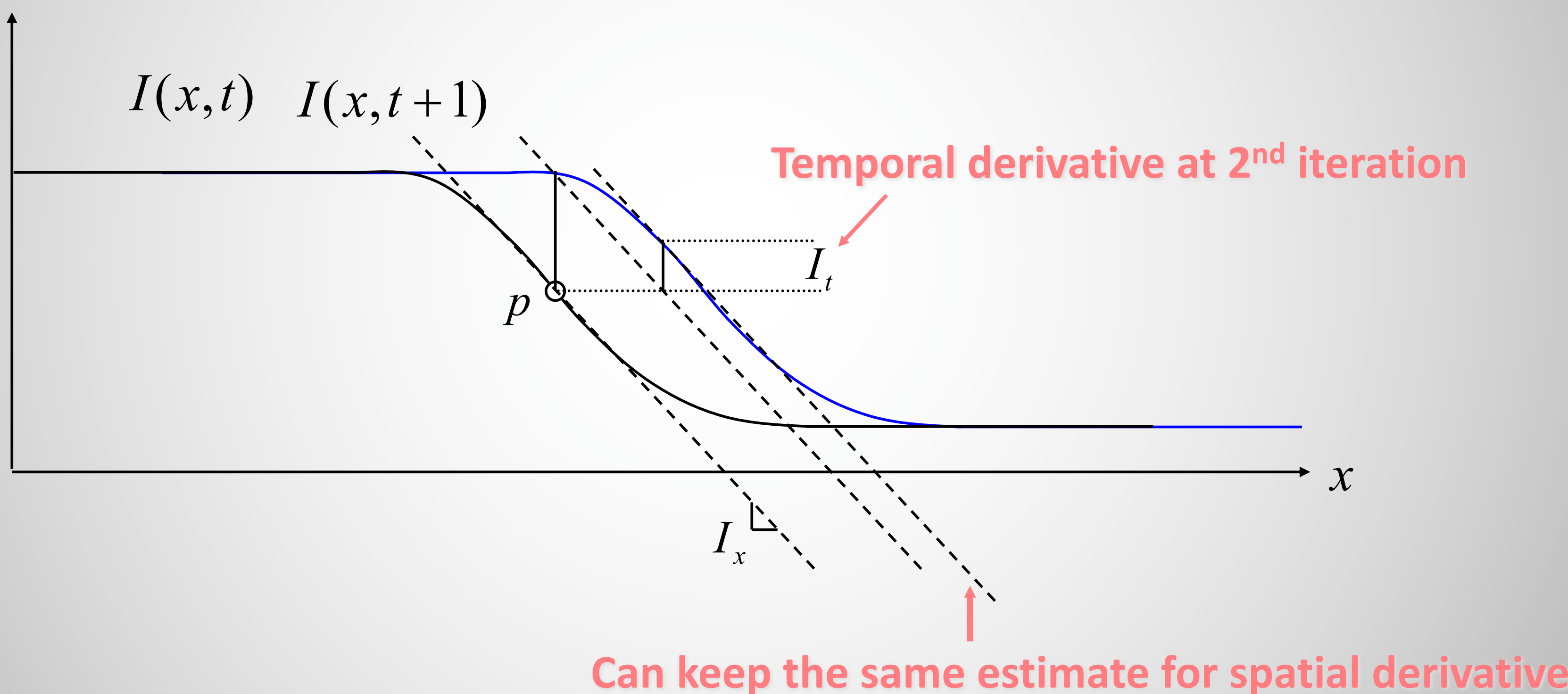
$$I_x = \left.\frac{\partial I}{\partial x}\right|_t \qquad I_t = \left.\frac{\partial I}{\partial t}\right|_{x=p} \quad \Rightarrow \quad \vec{v} \approx -\frac{I_t}{I_x}$$

Assumptions:

- **Brightness constancy**
- **Small motion**

Example: Optical Flow in 1D

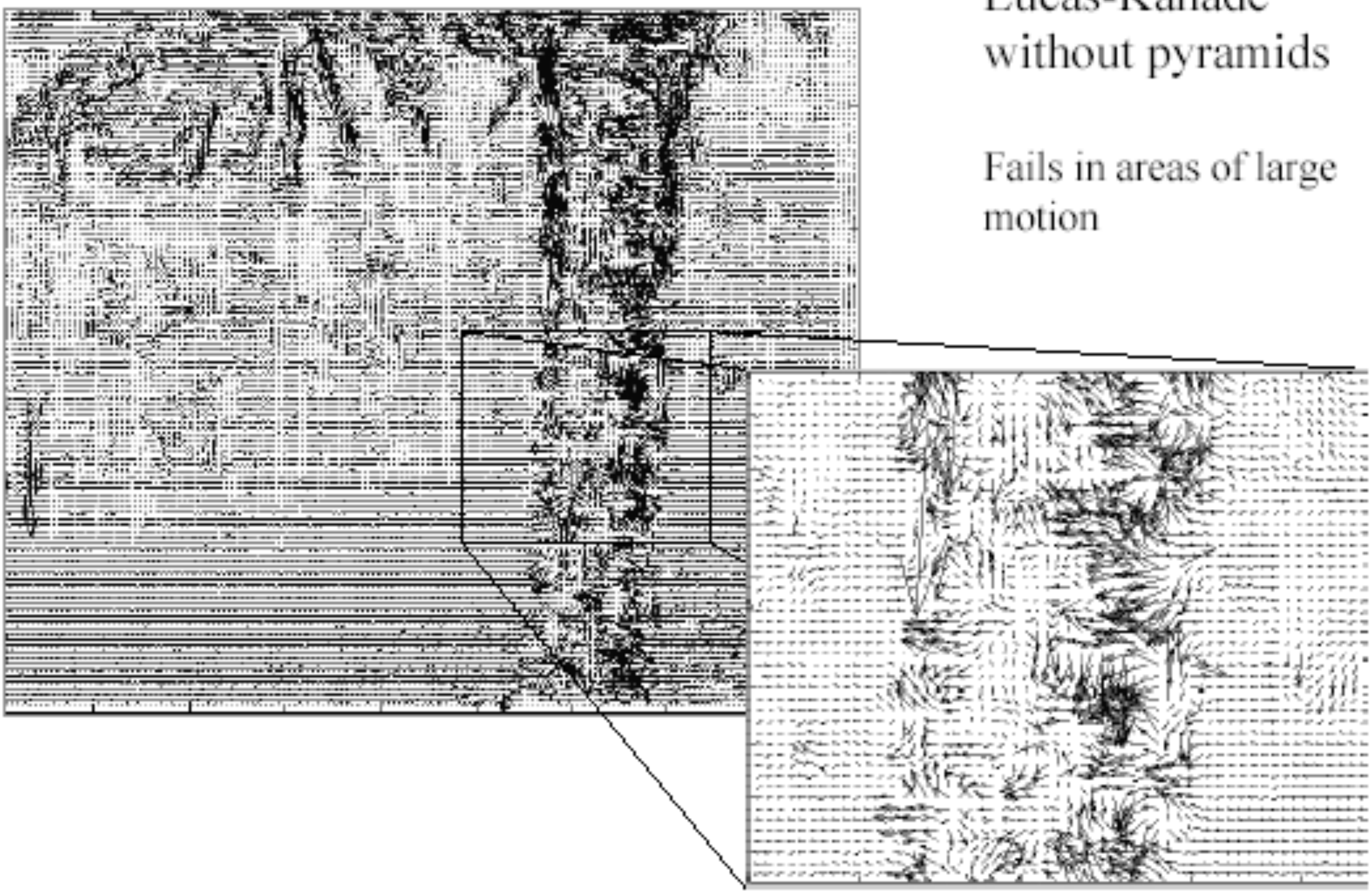
Iterating helps refining the velocity vector



$$\vec{v} \leftarrow \vec{v}_{previous} - \frac{I_t}{I_x}$$

Converges in about 5 iterations

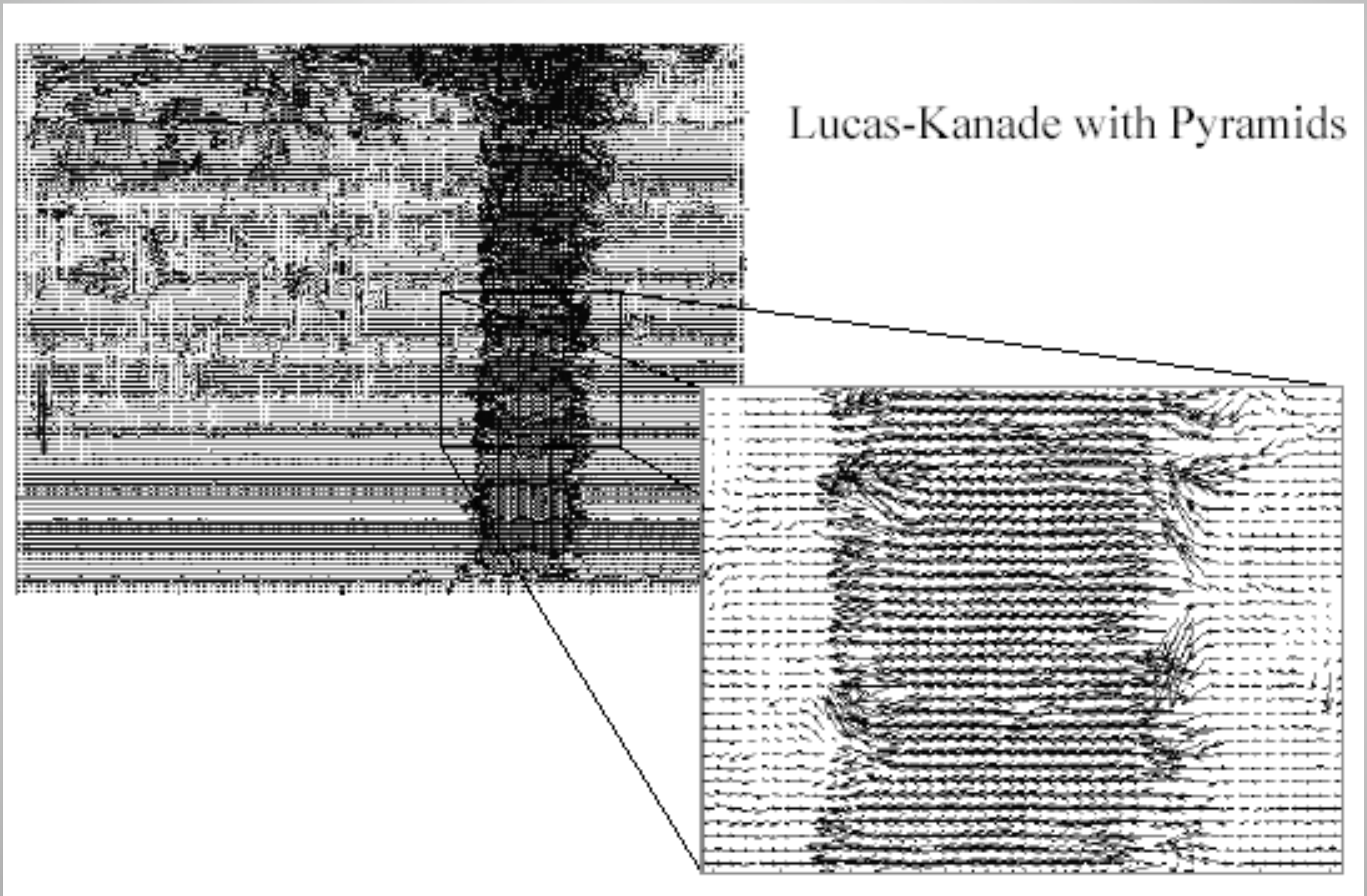
Lukas-Kanade without Pyramid



Lucas-Kanade without pyramids

Fails in areas of large motion

Lukas-Kanade with Pyramid



Horn & Schunck

- Global method with smoothness constraint to solve aperture problem

Horn & Schunck

- Global method with smoothness constraint to solve aperture problem
- Minimize a global energy function

Horn & Schunck

- Global method with smoothness constraint to solve aperture problem
- Minimize a global energy function

$$E(u,v) = \int_{x,y} [(I_x u + I_y v + I_t)^2 + \alpha^2(|\nabla u|^2 + |\nabla v|^2)]dxdy$$

Horn & Schunck

- Global method with smoothness constraint to solve aperture problem
- Minimize a global energy function

$$E(u,v) = \int_{x,y} [(I_x u + I_y v + I_t)^2 + \alpha^2(|\nabla u|^2 + |\nabla v|^2)] dx dy$$

- Take partial derivatives w.r.t. u and v:

Horn & Schunck

- Global method with smoothness constraint to solve aperture problem
- Minimize a global energy function

$$E(u,v) = \int_{x,y} [(I_x u + I_y v + I_t)^2 + \alpha^2(|\nabla u|^2 + |\nabla v|^2)] dxdy$$

- Take partial derivatives w.r.t. u and v:

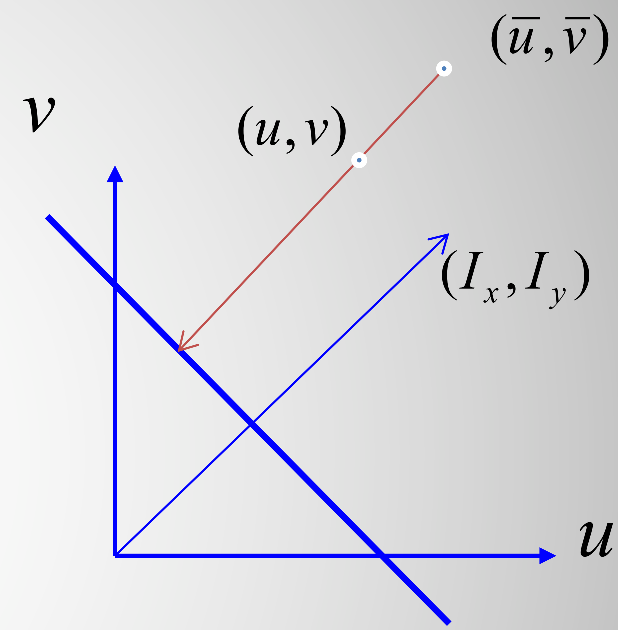
$$(I_x u + I_y v + I_t)I_x - \alpha^2 \nabla u = 0$$
$$(I_x u + I_y v + I_t)I_y - \alpha^2 \nabla v = 0$$

Horn & Schunck

- Iterative scheme

$$u^{k+1} = \overline{u}^k - \frac{I_x(I_x\overline{u}^k + I_y\overline{v}^k + I_t)}{\alpha^2 + I_x^2 + I_y^2}$$

$$v^{k+1} = \overline{v}^k - \frac{I_y(I_x\overline{u}^k + I_y\overline{v}^k + I_t)}{\alpha^2 + I_x^2 + I_y^2}$$

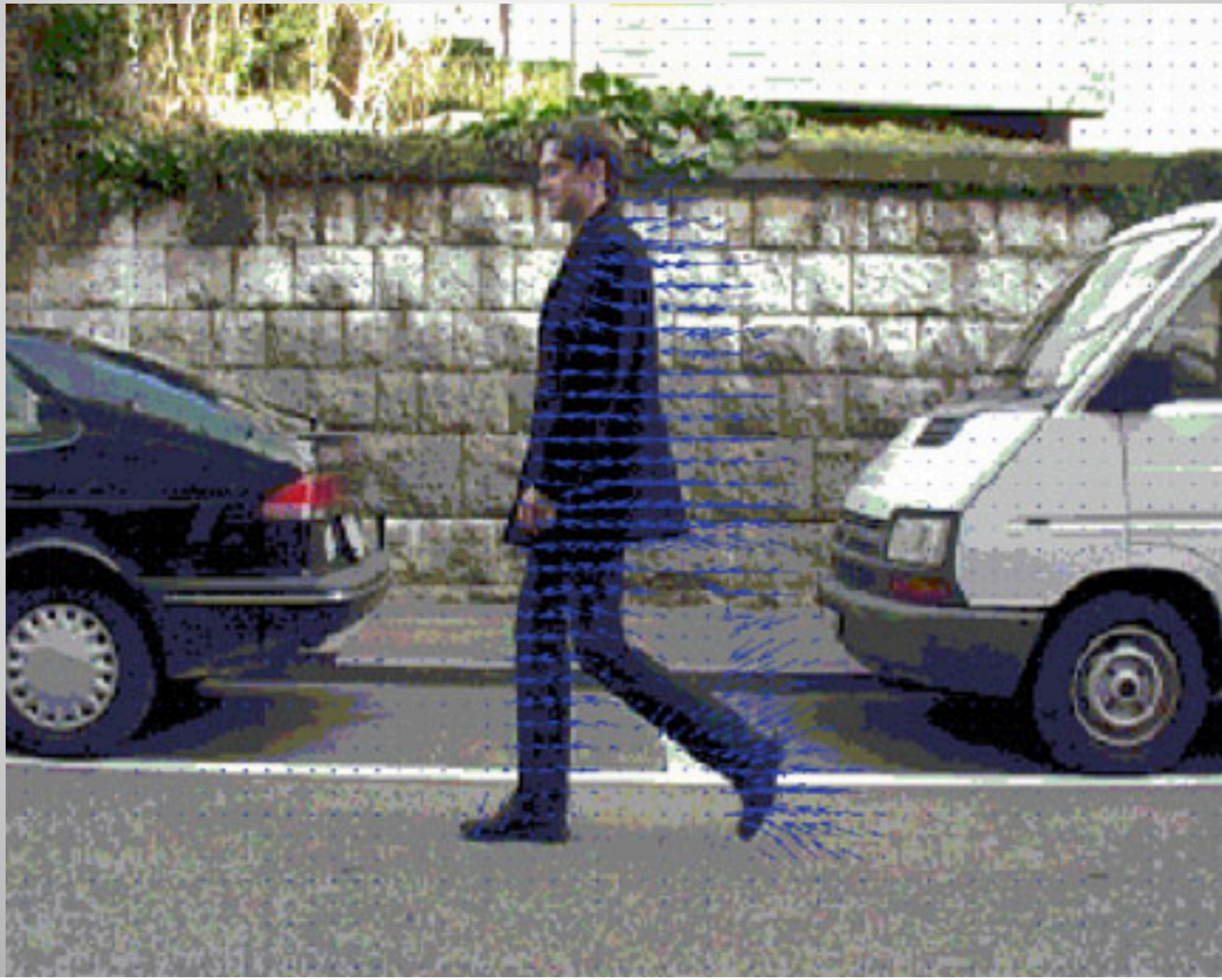


- Yields high density flow
- Fill in missing information in the homogenous regions
- More sensitive to noise than local methods

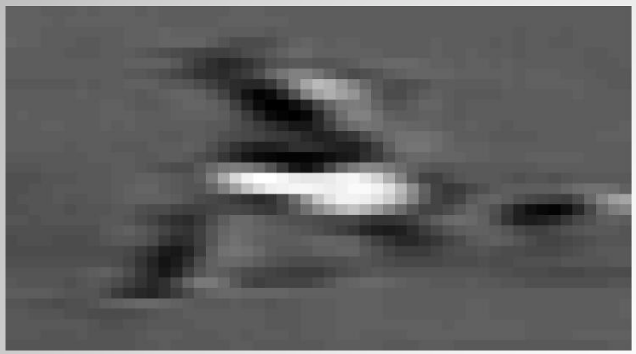
Optical Flow Matlab/C++/Python Code

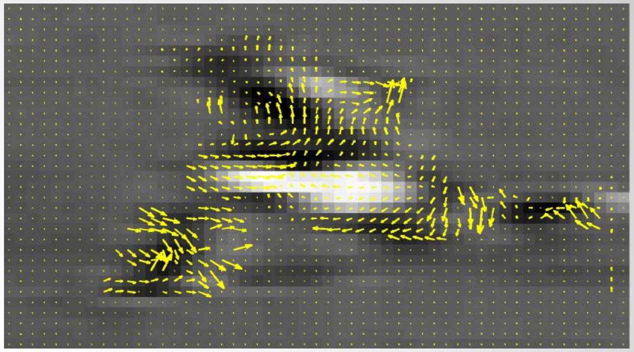
- http://people.csail.mit.edu/celiu/OpticalFlow/
- http://opencv-python-tutroals.readthedocs.org/en/latest/py_tutorials/py_video/py_lucas_kanade/py_lucas_kanade.html
- https://github.com/Itseez/opencv_attic/blob/a6078cc8477ff055427b67048a95547b3efe92a5/opencv/samples/python2/lk_track.py

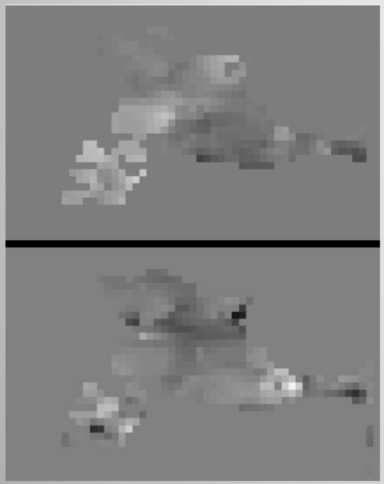
Applications: Target Tracking

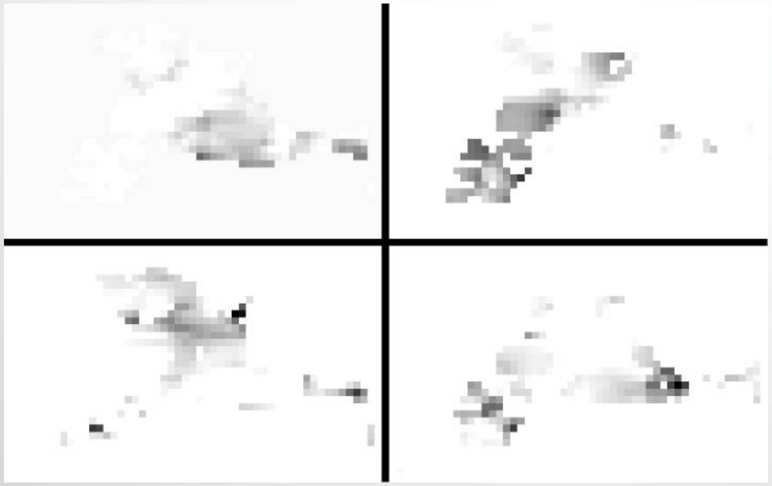


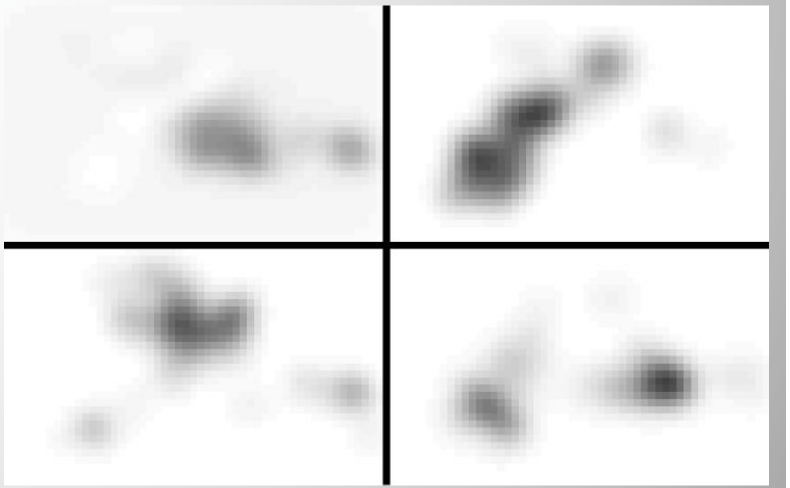
Applications: Action Recognition











F_x, F_y $F_x^-, F_x^+, F_y^-, F_y^+$ blurred $F_x^-, F_x^+, F_y^-, F_y^+$

Recognition actions at a distance, Efros et al.

Applications: Motion Modeling

Flipping between image 1 and 2.

Estimated flow field
(hue indicates orientation
and saturation indicates magnitude)

Second image is warped!

Slide credit to C.Liu

Applications: Motion Segmentation

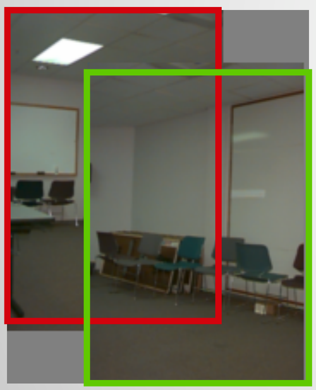
Global Motion Models (Parametric)

All pixels are considered to summarize global motion!

- 2D Models
  - Affine
  - Quadratic
  - Planar projective (homography)
- 3D Models
  - Inst. Camera motion models
  - Homography+epipole
  - Plane+parallalx

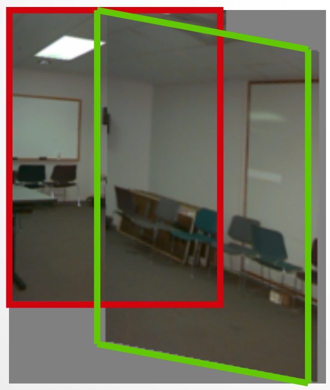
Motion Models

Translation



2 unknowns

Affine

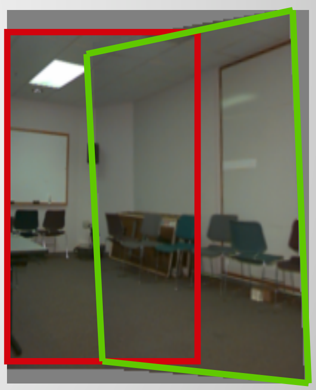


6 unknowns

Perspective

8 unknowns

3D rotation



3 unknowns

Example: Affine Motion

$$u(x,y) = a_1 + a_2 x + a_3 y$$

$$u(x,y) = a_4 + a_5 x + a_6 y$$

Example: Affine Motion

$$u(x,y) = a_1 + a_2 x + a_3 y$$

$$u(x,y) = a_4 + a_5 x + a_6 y$$

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

Example: Affine Motion

$$u(x,y) = a_1 + a_2 x + a_3 y$$

$$u(x,y) = a_4 + a_5 x + a_6 y$$

$$I_x(a_1 + a_2 x + a_3 y) + I_y(a_4 + a_5 x + a_6 y) + I_t \approx 0$$

Example: Affine Motion

$$u(x, y) = a_1 + a_2 x + a_3 y$$

$$u(x, y) = a_4 + a_5 x + a_6 y$$

$$I_x(a_1 + a_2 x + a_3 y) + I_y(a_4 + a_5 x + a_6 y) + I_t \approx 0$$

Each pixel provides 1 linear constraint in 6 global unknowns (a1,..,a6)

Example: Affine Motion

$$u(x,y) = a_1 + a_2 x + a_3 y$$

$$u(x,y) = a_4 + a_5 x + a_6 y$$

$$I_x(a_1 + a_2 x + a_3 y) + I_y(a_4 + a_5 x + a_6 y) + I_t \approx 0$$

Each pixel provides 1 linear constraint in 6 global unknowns (a1,..,a6)

Over all pixels, minimize the least square to find unknowns!

$$Err_{a_1,\ldots,a_6} = \sum [I_x(a_1 + a_2 x + a_3 y) + I_y(a_4 + a_5 x + a_6 y)]^2$$

Other 2D Motion Models

- Quadratic

$$u = q_1 + q_2 x + q_3 y + q_7 x^2 + q_8 xy$$
$$u = q_4 + q_5 x + q_6 y + q_7 xy + q_8 y^2$$

- Projective

$$u = \frac{h_1 + h_2 x + h_3 y}{h_7 + h_8 x + h_9 y} - x$$
$$v = \frac{h_4 + h_5 x + h_6 y}{h_7 + h_8 x + h_9 y} - y$$

FlowCap: 2D Human Pose from Optical FLow