

The Optical Flow Field

Caltech, Oct. 2004

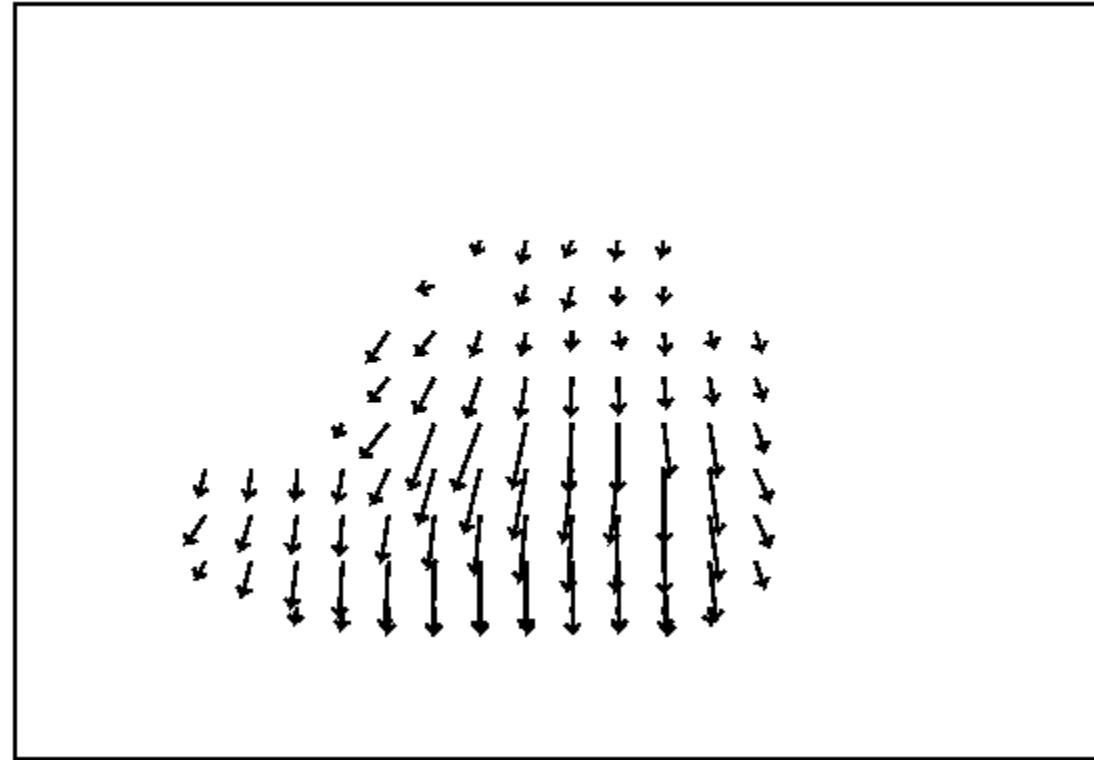
Lihi Zelnik-Manor

Optical Flow

Where did each pixel in image 1 go to in image 2



Optical Flow



Pierre Kornprobst's Demo

Introduction

- Given a video sequence with camera/objects moving we can better understand the scene if we find the motions of the camera/objects.

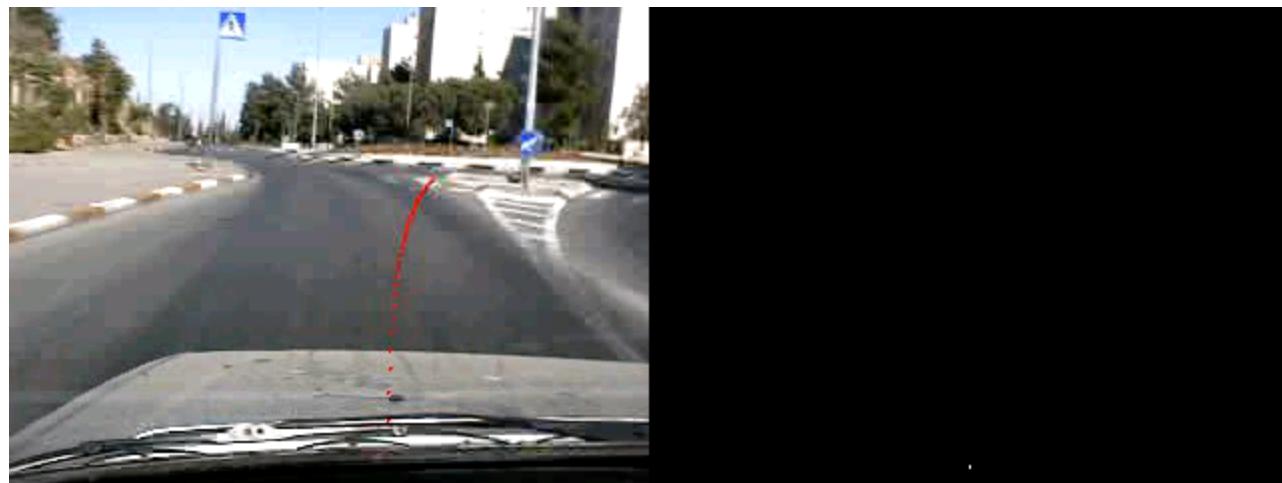


Scene Interpretation

- How is the camera moving?
- How many moving objects are there?
- Which directions are they moving in?
- How fast are they moving?
- Can we recognize their type of motion (e.g. walking, running, etc.)?

Applications

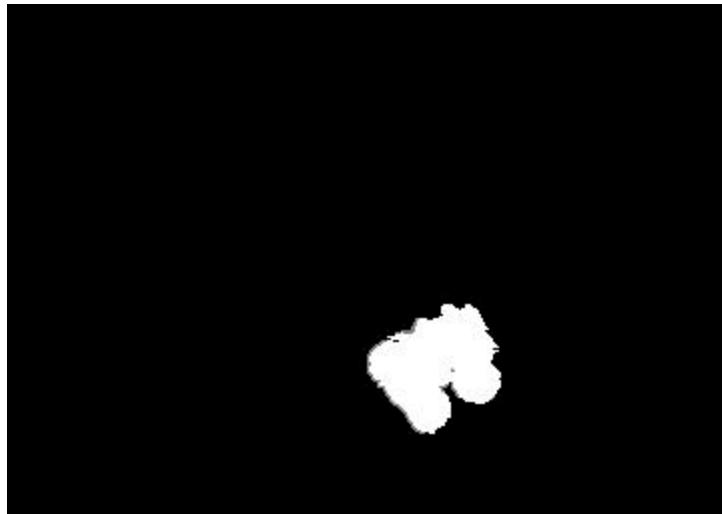
- Recover camera ego-motion.



Result by MobilEye (www.mobileye.com)

Applications

- Motion segmentation



Result by: L.Zelnik-Manor, M.Machline, M.Irani
“Multi-body Segmentation: Revisiting Motion Consistency” To appear, IJCV

Applications

- Structure from Motion



Input

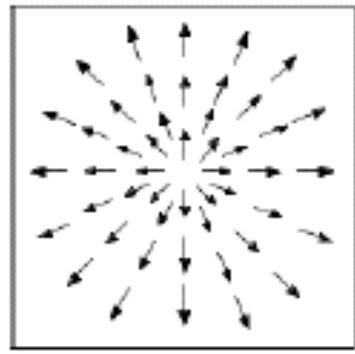


Reconstructed shape

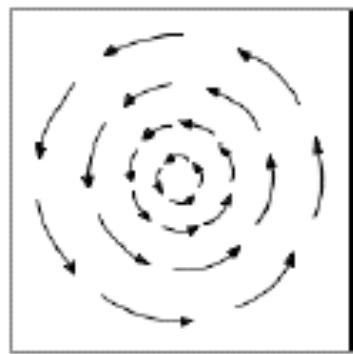
Result by: L. Zhang, B. Curless, A. Hertzmann, S.M. Seitz

“Shape and motion under varying illumination: Unifying structure from motion, photometric stereo, and multi-view stereo” ICCV’03

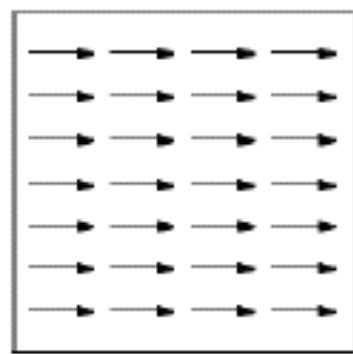
Examples of Motion fields



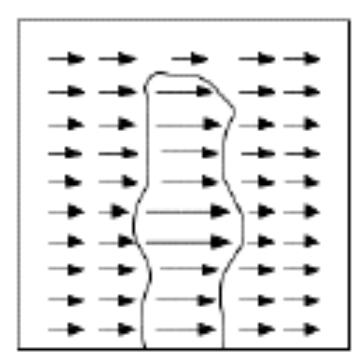
Forward
motion



Rotation



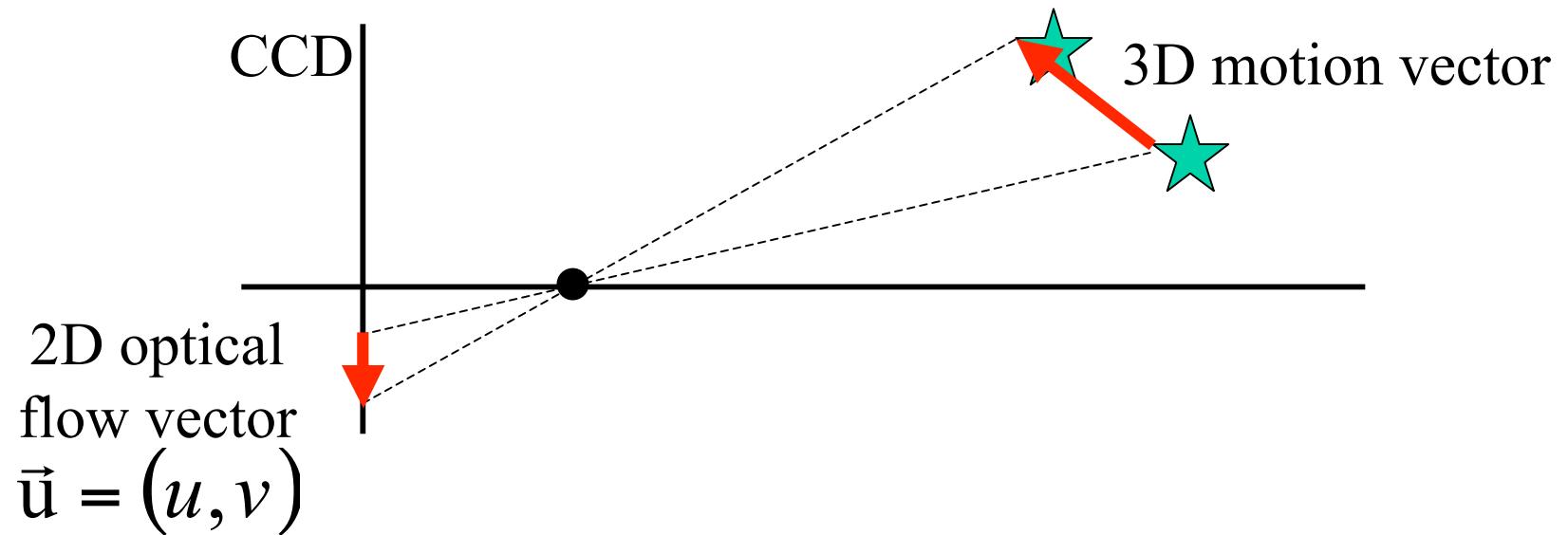
Horizontal
translation



Closer
objects
appear to
move faster!!

Motion Field & Optical Flow Field

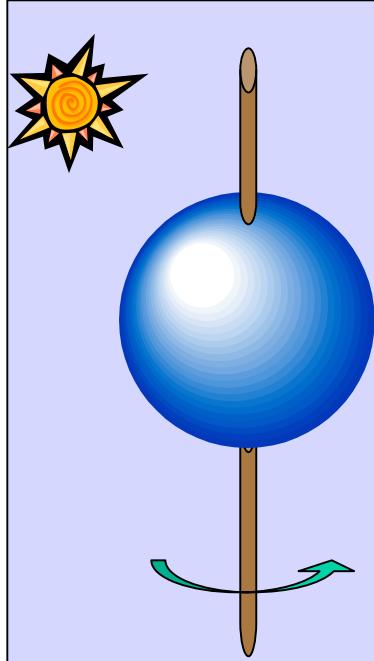
- Motion Field = Real world 3D motion
- Optical Flow Field = Projection of the motion field onto the 2d image



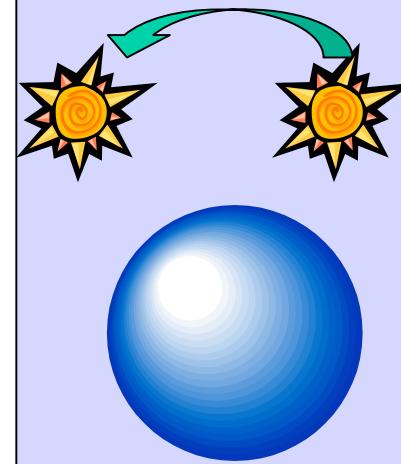
When does it break?



The screen is
stationary yet
displays motion



Homogeneous
objects
generate zero
optical flow.



Fixed sphere.
Changing light
source.



Non-rigid
texture motion

The Optical Flow Field

Still, in many cases it does work....

- Goal:
Find for each pixel a velocity vector $\vec{u} = (u, v)$
which says:
 - How quickly is the pixel moving across the image
 - In which direction it is moving

How do we actually do that?

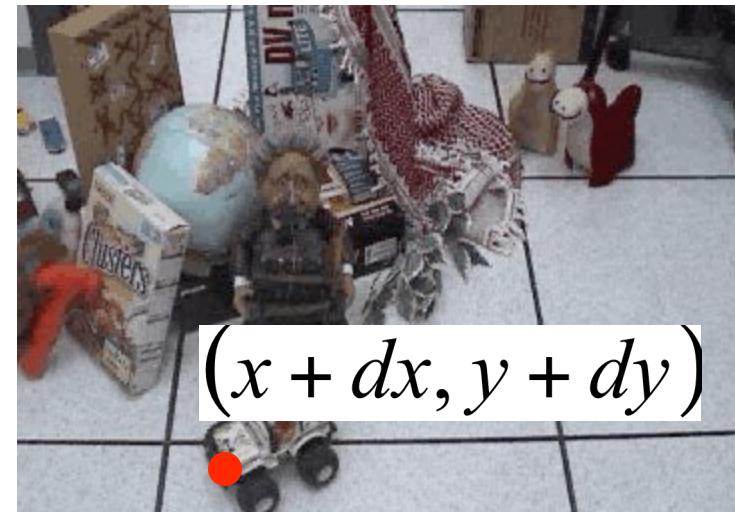
Estimating Optical Flow

- Assume the image intensity I is constant

Time = t



Time = $t+dt$



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

Brightness Constancy Equation

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

First order Taylor Expansion

$$= I(x, y, t) + \frac{\partial I}{\partial x} dx + \frac{\partial I}{\partial y} dy + \frac{\partial I}{\partial t} dt$$

Simplify notations:

$$I_x dx + I_y dy + I_t dt = 0$$

Divide by dt and denote:

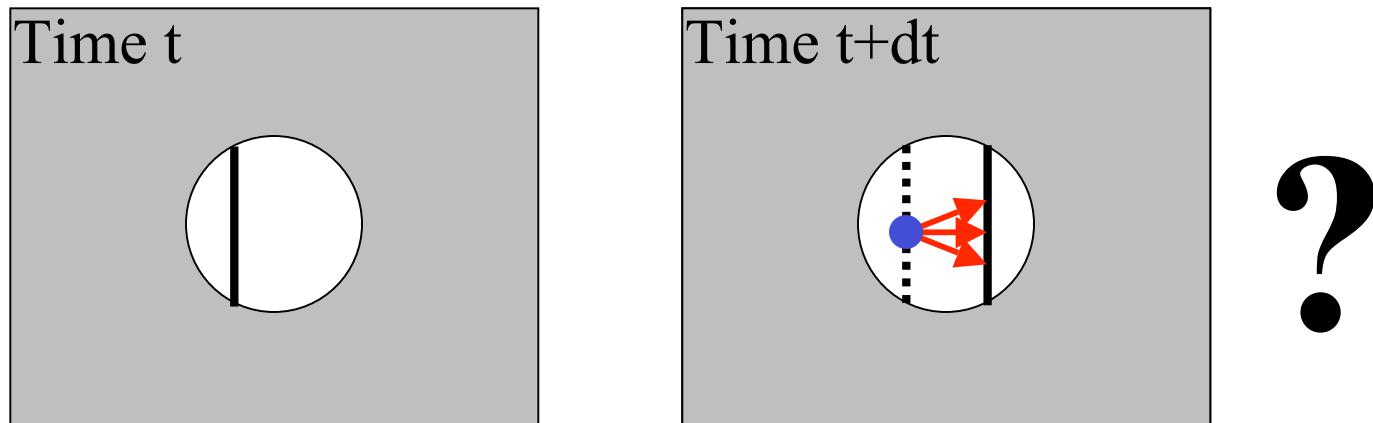
$$u = \frac{dx}{dt} \quad v = \frac{dy}{dt}$$

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

Problem I: One equation, two unknowns

Problem II: “The Aperture Problem”

- For points on a line of fixed intensity we can only recover the normal flow

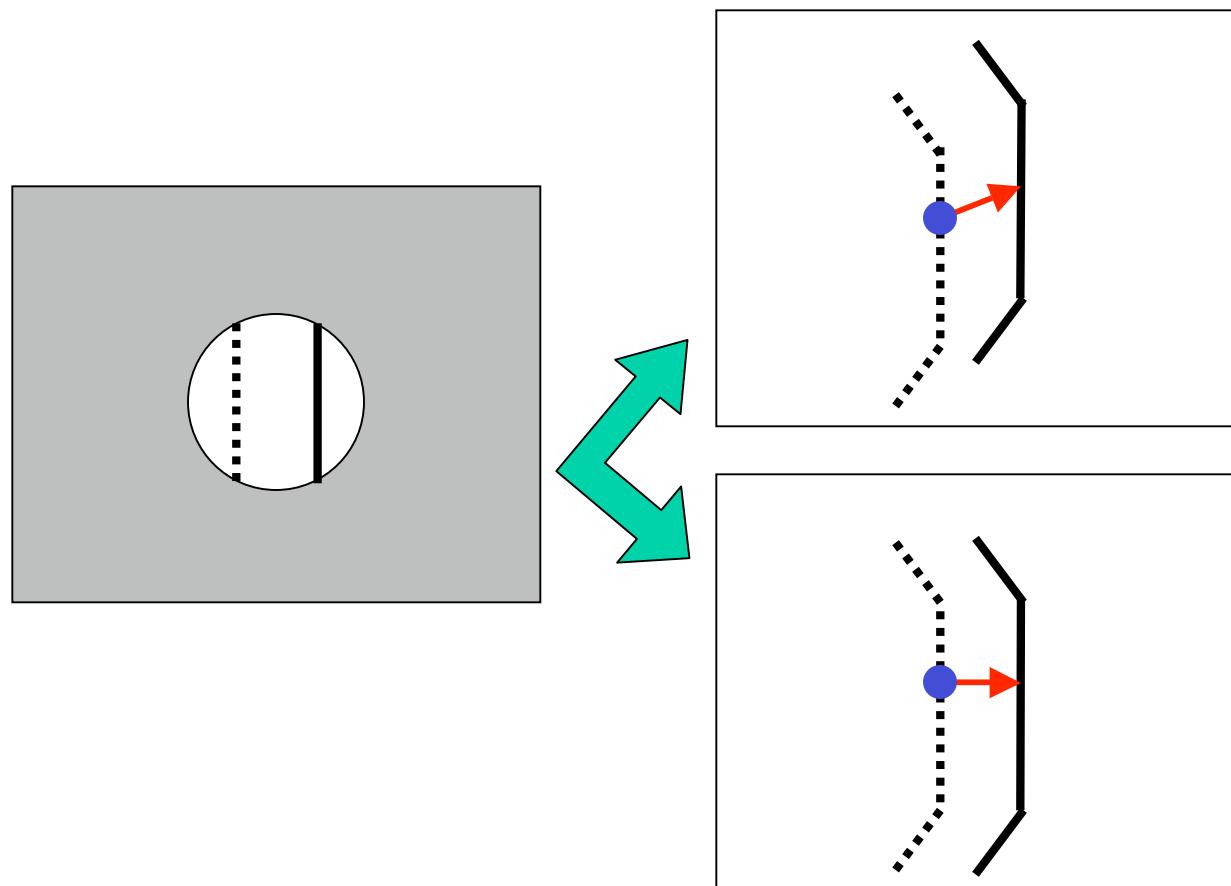


Where did the blue point move to?

We need additional constraints

Use Local Information

Sometimes enlarging the aperture can help

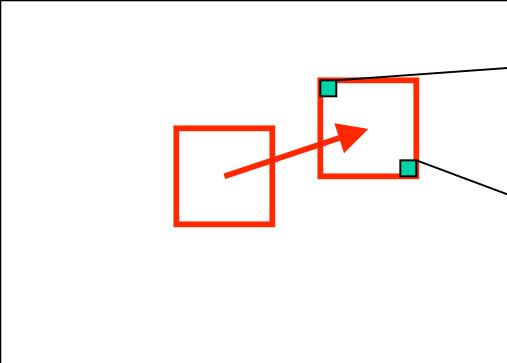


Local smoothness

Lucas Kanade (1984)

$$I_x u + I_y v = -I_t \quad \rightarrow \quad \begin{bmatrix} I_x & I_y \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = -I_t$$

Assume constant (u, v) in small neighborhood


$$\begin{bmatrix} I_{x1} & I_{y1} \\ I_{x2} & I_{y2} \\ \vdots & \vdots \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_{t1} \\ I_{t2} \\ \vdots \end{bmatrix}$$

$A\vec{u} = b$

Lucas Kanade (1984)

Goal: Minimize $\|A\vec{u} - b\|^2$

Method: Least-Squares

$$A\vec{u} = b$$



$$\underbrace{A^T A}_{2 \times 2} \underbrace{\vec{u}}_{2 \times 1} = \underbrace{A^T b}_{2 \times 1}$$



$$\vec{u} = (A^T A)^{-1} A^T b$$

How does Lucas-Kanade behave?

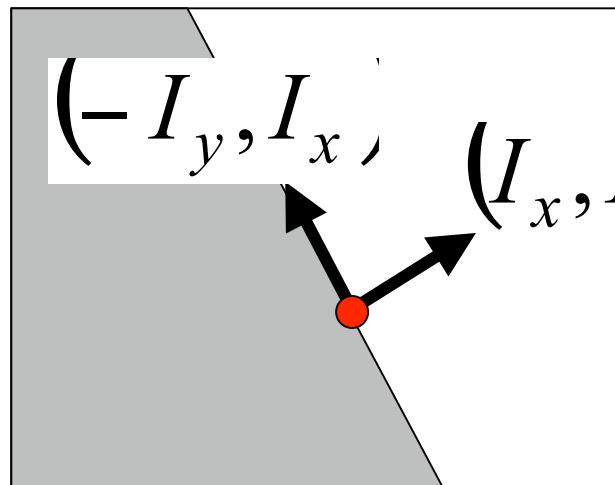
$$\vec{u} = (A^T A)^{-1} A^T b$$

$$A^T A = \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix}$$

We want this matrix to be invertible.
i.e., no zero eigenvalues

How does Lucas-Kanade behave?

- Edge $\rightarrow A^T A$ becomes singular



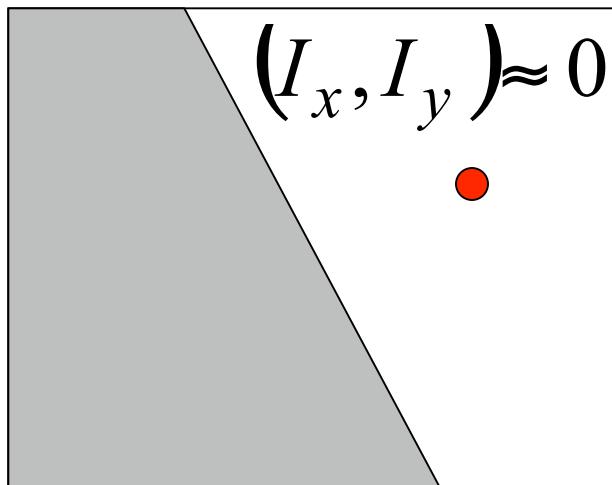
$$\begin{bmatrix} -I_y & I_x \\ I_x & I_y \end{bmatrix} \begin{bmatrix} \sum I_x^2 & \sum I_x I_y \\ \sum I_x I_y & \sum I_y^2 \end{bmatrix} \begin{bmatrix} -I_y \\ I_x \end{bmatrix} = \begin{bmatrix} 0 \\ 0 \end{bmatrix}$$

\downarrow

$$\begin{bmatrix} -I_y \\ I_x \end{bmatrix} \text{ is eigenvector with eigenvalue 0}$$

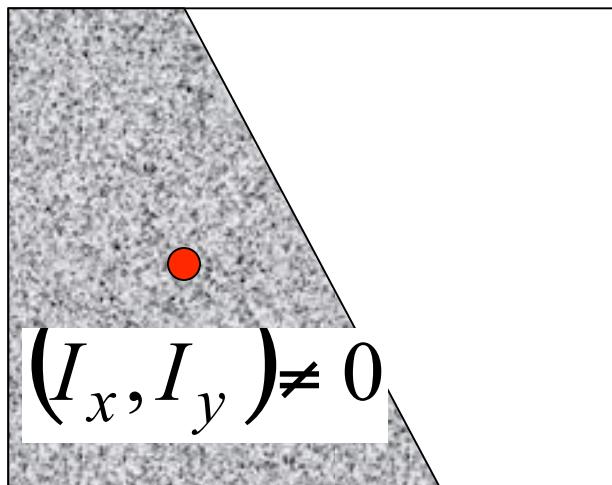
How does Lucas-Kanade behave?

- Homogeneous $\rightarrow A^T A \approx 0 \rightarrow$ 0 eigenvalues



How does Lucas-Kanade behave?

- Textured regions → two high eigenvalues



How does Lucas-Kanade behave?

- Edge → $A^T A$ becomes singular

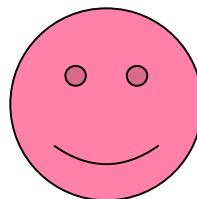


- Homogeneous regions → low gradients

$$A^T A \approx 0$$



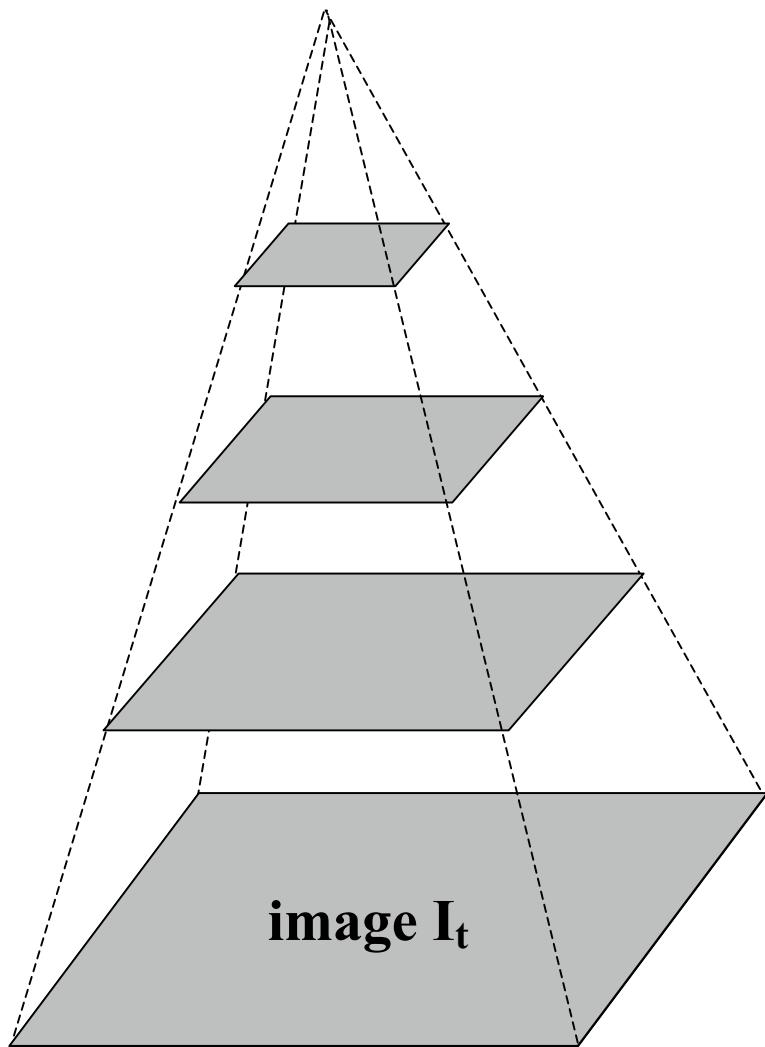
- High texture →



Other break-downs

- Brightness constancy is **not** satisfied
 - Correlation based methods
- A point does **not** move like its neighbors
 - what is the ideal window size?
 - Regularization based methods
- The motion is **not** small (Taylor expansion doesn't hold)
 - Use multi-scale estimation

Multi-Scale Flow Estimation



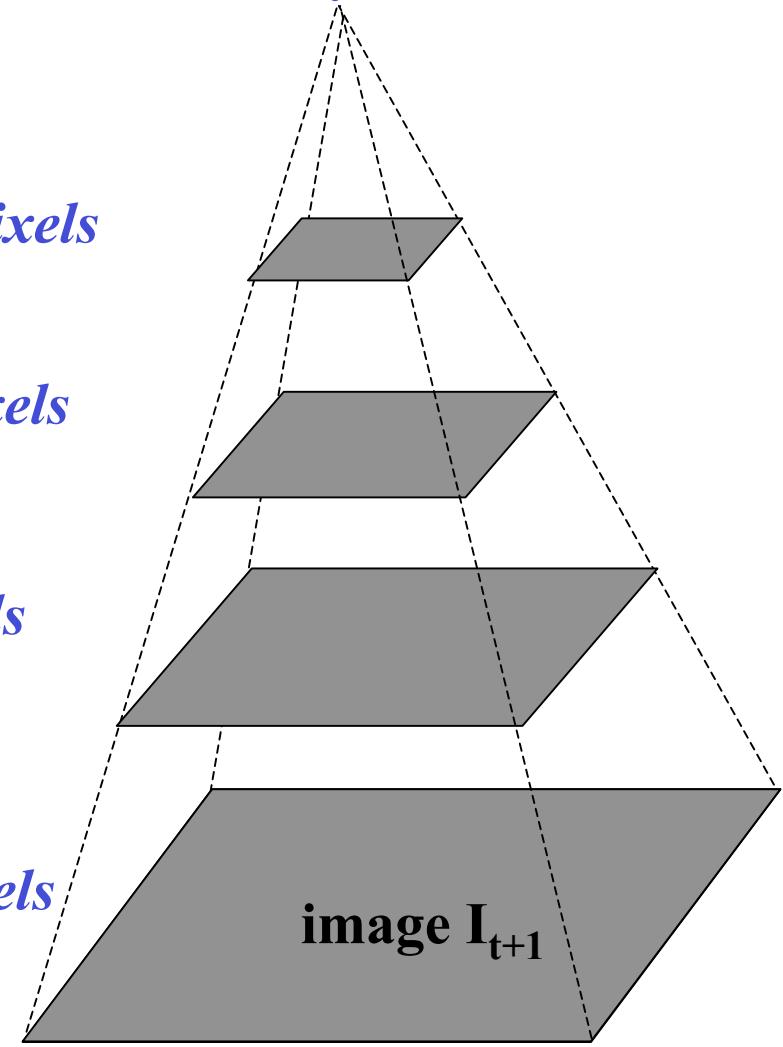
Gaussian pyramid of image I_t

$u=1.25 \text{ pixels}$

$u=2.5 \text{ pixels}$

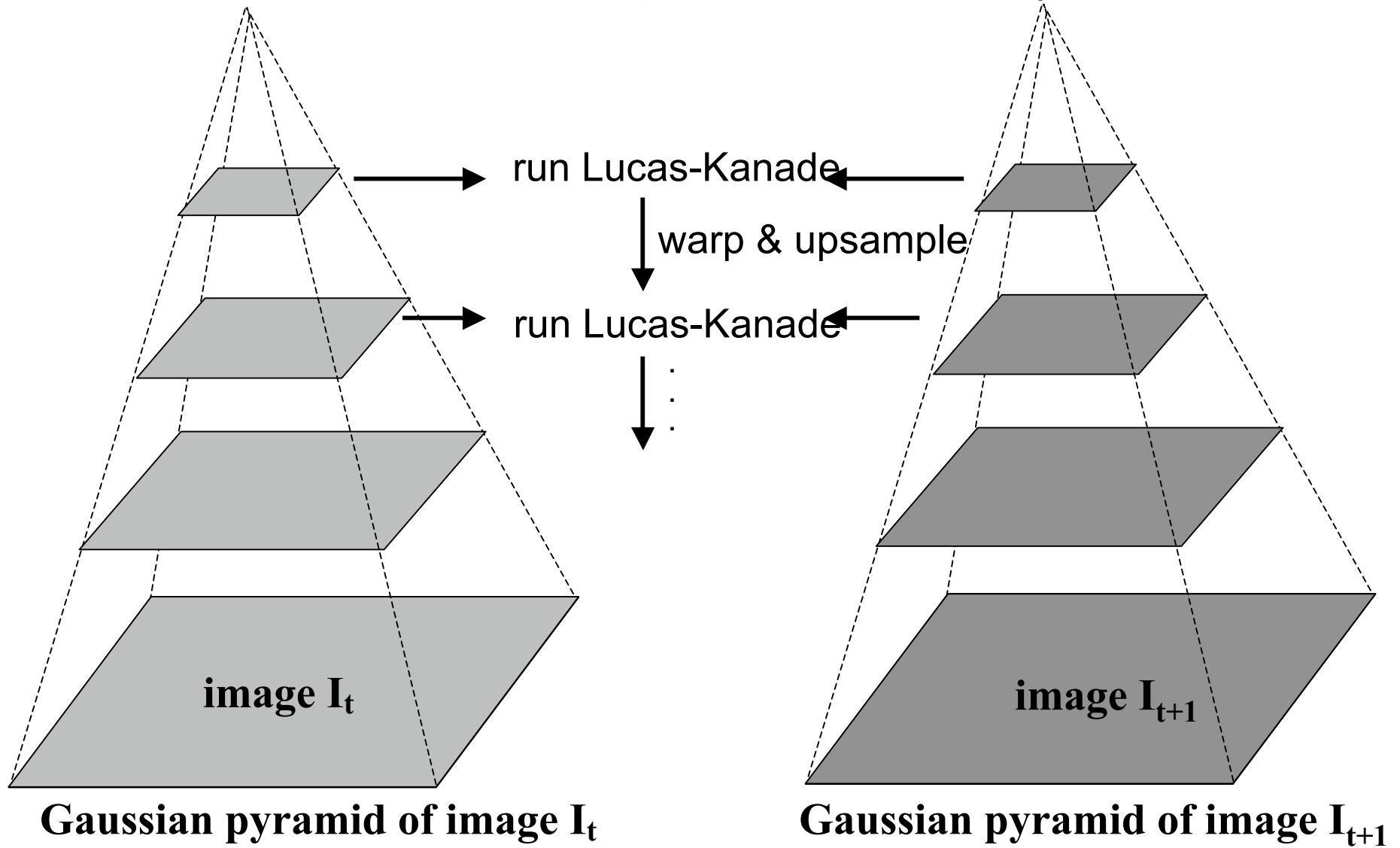
$u=5 \text{ pixels}$

$u=10 \text{ pixels}$



Gaussian pyramid of image I_{t+1}

Multi-Scale Flow Estimation



Examples: Motion Based Segmentation



Input



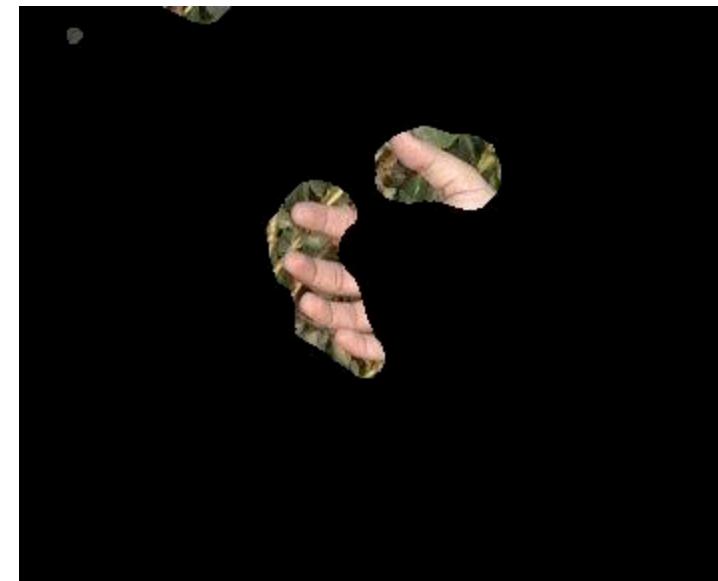
Segmentation result

Result by: L.Zelnik-Manor, M.Machline, M.Irani
“Multi-body Segmentation: Revisiting Motion Consistency” To appear, IJCV

Examples: Motion Based Segmentation



Input

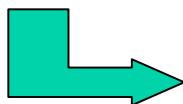


Segmentation result

Result by: L.Zelnik-Manor, M.Machline, M.Irani
“Multi-body Segmentation: Revisiting Motion Consistency” To appear, IJCV

Other break-downs

- Brightness constancy is **not** satisfied



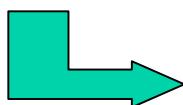
Correlation based methods

- A point does **not** move like its neighbors
 - what is the ideal window size?



Regularization based methods

- The motion is **not** small (Taylor expansion doesn't hold)



Use multi-scale estimation

Regularization

Horn and Schunk (1981)

Add global smoothness term

Smoothness error:

$$E_s = \iint_D (u_x^2 + u_y^2) + (v_x^2 + v_y^2) dx dy$$

Error in brightness
constancy equation

$$E_c = \iint_D (I_x u + I_y v + I_t)^2 dx dy$$

Minimize:

$$E_c + \lambda E_s$$

Solve by calculus of variations

Robust Estimation

Black & Anandan (1993)

Regularization can over-smooth across edges



Use “smarter” regularization

Minimize:

$$\iint_D \underbrace{\rho_1(I_x u + I_y v + I_t)}_{\text{Brightness constancy}} + \lambda \underbrace{[\rho_2(u_x, u_y) \rho_2(v_x, v_y)]}_{\text{Smoothness}} dx dy$$

Examples: Motion Based Segmentation



Input



Segmentation result

- Optical Flow estimation by: M. J . Black and P. Anandan,
“A framework for the robust estimation of optical flow”, ICCV’93
- Segmentation by: L.Zelnik-Manor, M.Machline, M.Irani
“Multi-body Segmentation: Revisiting Motion Consistency” To appear, IJCV

Affine Motion

For panning camera or planar surfaces:

$$u = p_1 + p_2x + p_3y$$

$$v = p_4 + p_5x + p_6y$$

$$I_x(p_1 + p_2x + p_3y) + I_y(p_4 + p_5x + p_6y) = -I_t$$

$$\begin{bmatrix} I_x & I_x x & I_x y & | & I_y & I_y x & I_y y \end{bmatrix} \vec{p} = -I_t$$

Only 6 parameters to solve for → Better results

Segmentation of Affine Motion

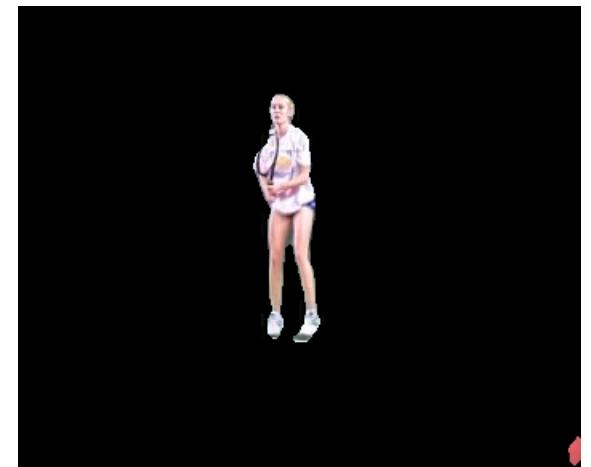


Input

=



+



Segmentation result

Result by: L.Zelnik-Manor, M.Irani
“Multi-frame estimation of planar motion”, PAMI 2000

Panoramas

Input



Motion estimation by Andrew Zisserman's group

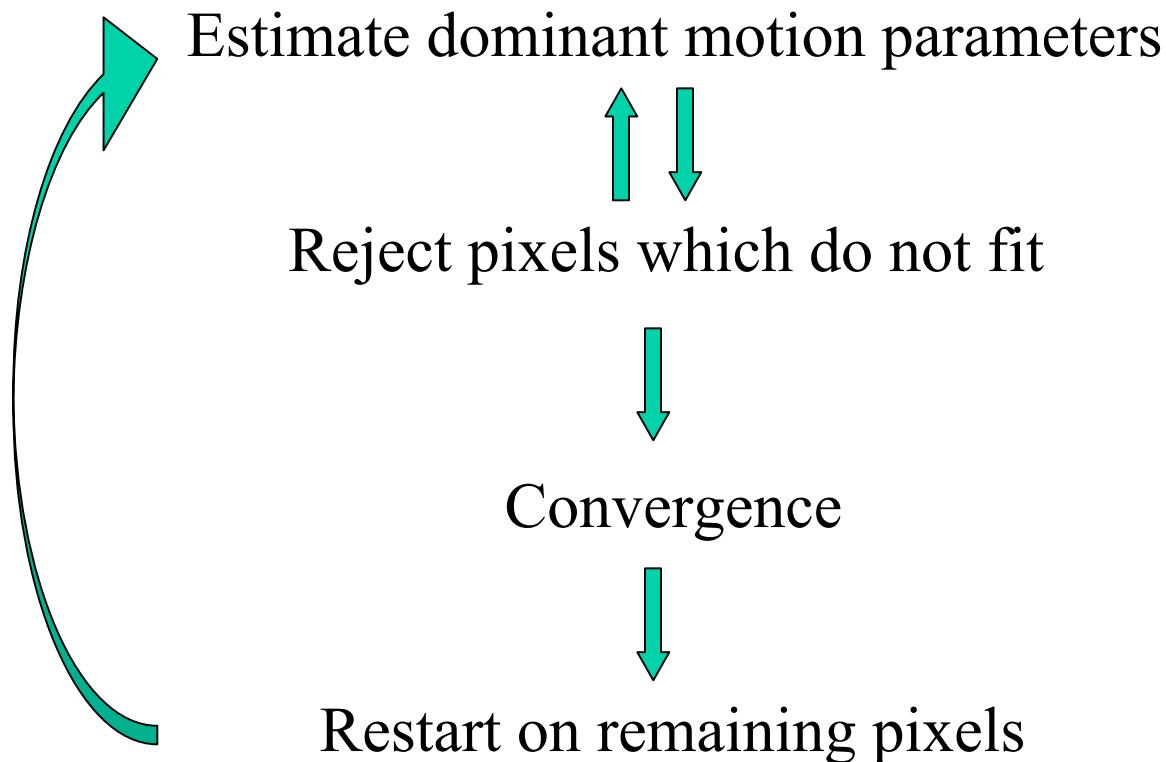
Stabilization



Result by: L.Zelnik-Manor, M.Irani
“Multi-frame estimation of planar motion”, PAMI 2000

Layered Representation

For scenes with multiple affine motions



Some Results



Nebojsa Jojic and Brendan Frey,
"Learning Flexible Sprites in Video Layers", CVPR 2001.

Action Recognition

- A bit more fun

“Recognizing Action at a Distance”

A.A. Efros, A.C. Berg, G. Mori, J. Malik

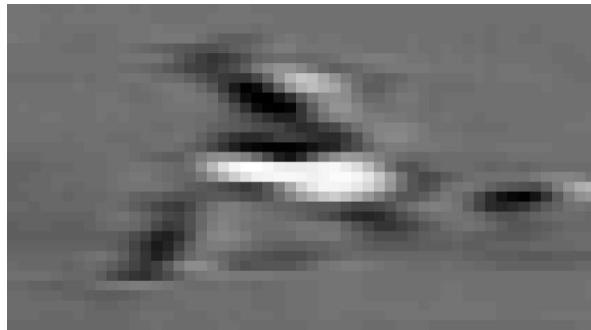
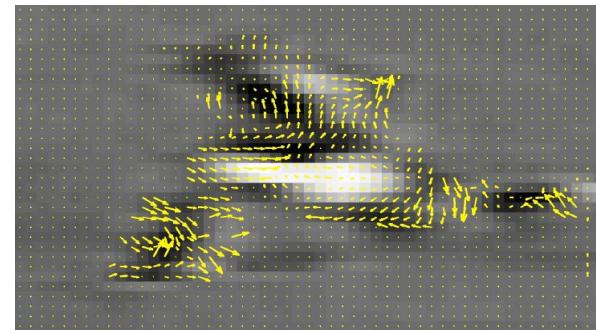


Image frame



Optical flow $F_{x,y}$

Use optical flow as a template for frame classification

“Recognizing Action at a Distance”

A.A. Efros, A.C. Berg, G. Mori, J. Malik

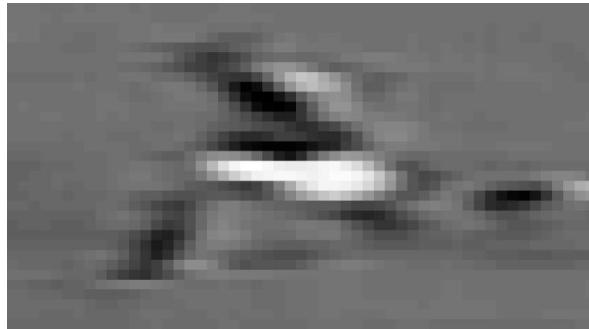
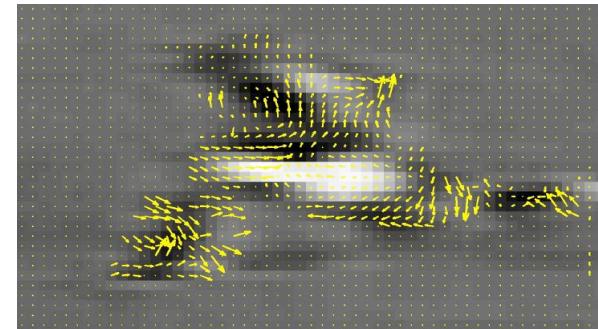
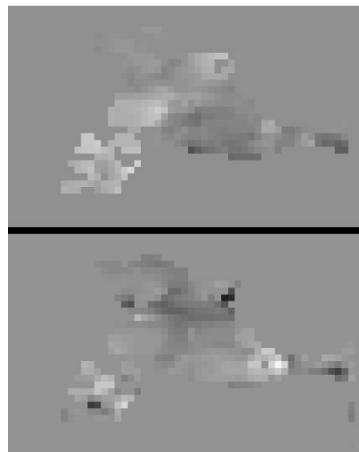


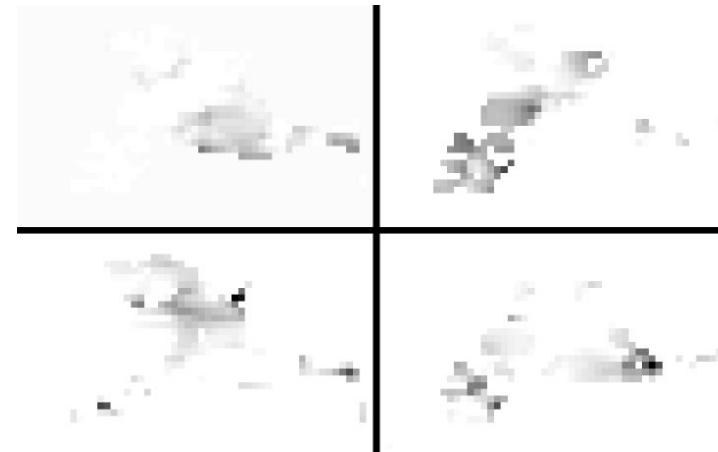
Image frame



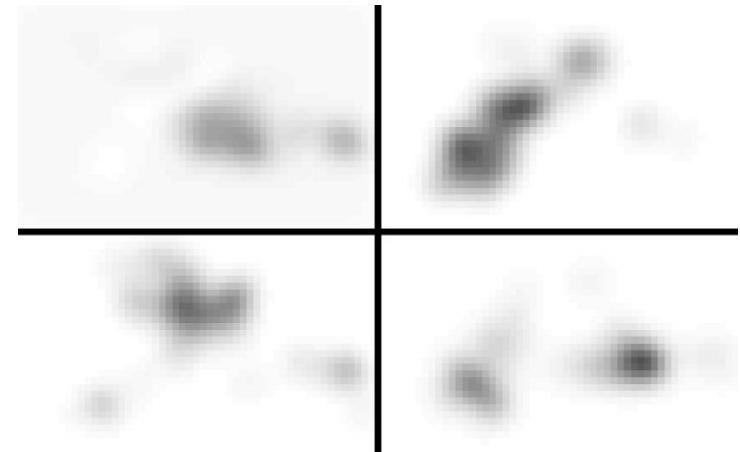
Optical flow $F_{x,y}$



F_x, F_y



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



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

“Recognizing Action at a Distance”

A.A. Efros, A.C. Berg, G. Mori, J. Malik

Database:



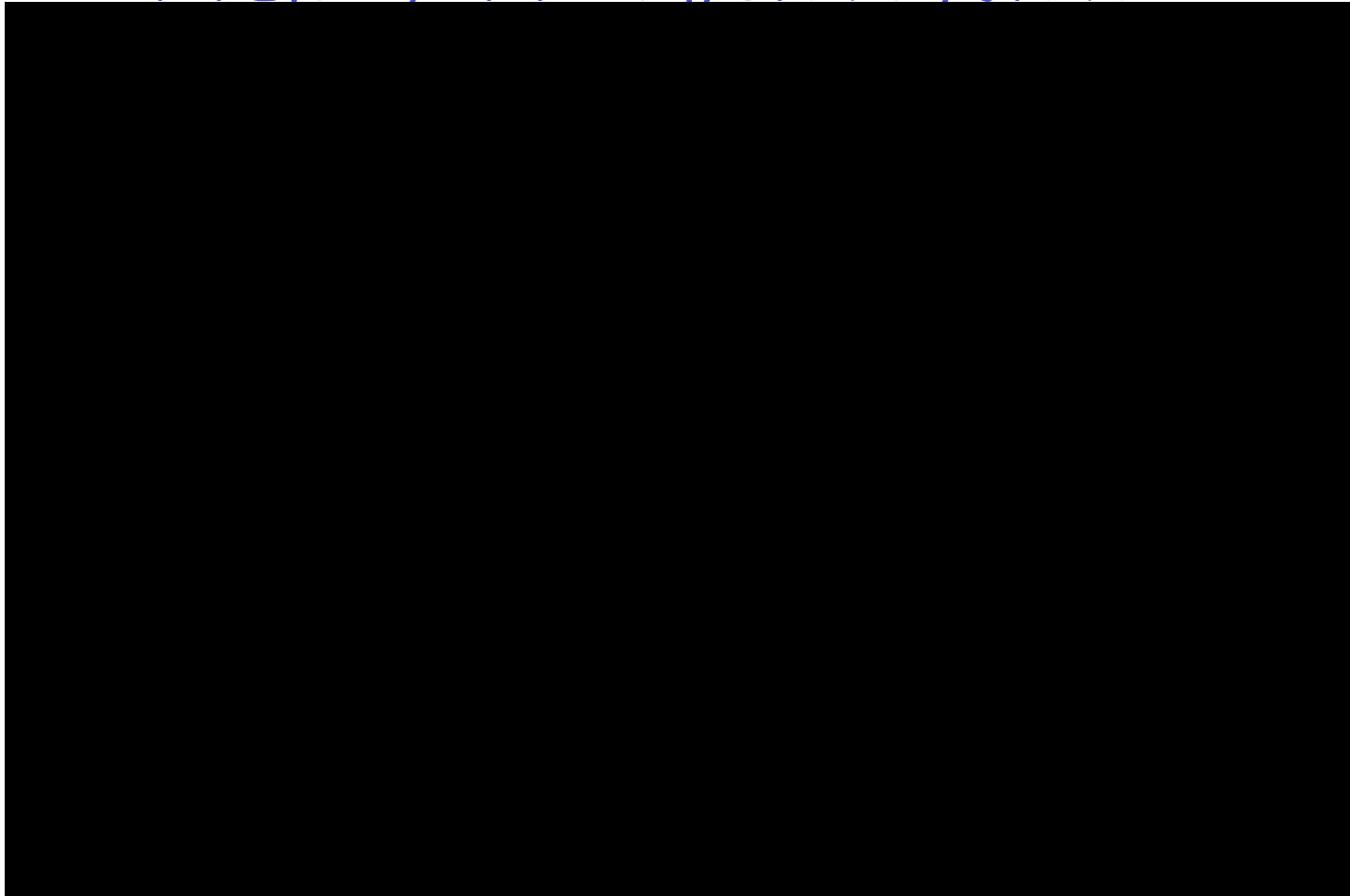
Test sequence:



For each frame in test sequence find closest frame in database

“Recognizing Action at a Distance”

A.A. Efros, A.C. Berg, G. Mori, J. Malik



- Red bars show classification results

“Recognizing Action at a Distance”

A.A. Efros, A.C. Berg, G. Mori, J. Malik

View “Greg In World Cup” video

References on Optical Flow

Lucas-Kanade method:

- B.D. Lucas and T. Kanade “*An Iterative Image Registration Technique with an Application to Stereo Vision*” IJCAI '81 pp. 674-679
- S. Baker and I. Matthews “*Lucas-Kanade 20 Years On: A Unifying Framework*” IJCV, Vol. 56, No. 3, March, 2004, pp. 221 - 255.
http://www.ri.cmu.edu/projects/project_515.html (papers + code)

Regularization based methods:

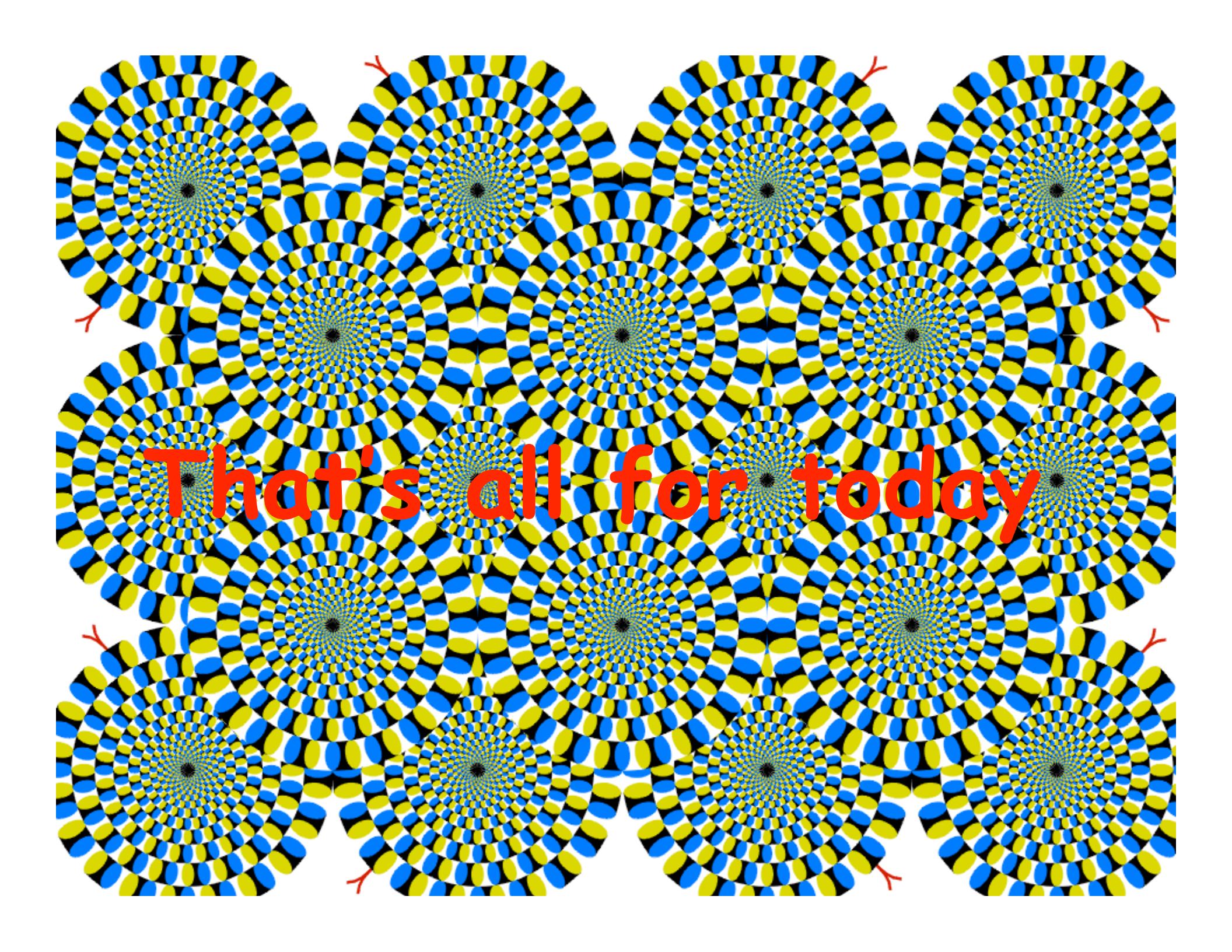
- B. K. P. Horn and B. Schunck, "Determining Optical Flow," Artificial Intelligence, 17 (1981), pp. 185-203
- Black, M. J. and Anandan, P., “*A framework for the robust estimation of optical flow*”, ICCV'93, May, 1993, pp. 231-236 ([papers + code](#))

Comparison of various optical flow techniques:

Barron, J.L., Fleet, D.J., and Beauchemin, S. “*Performance of optical flow techniques*”. IJCV, 1994, 12(1):43-77

Layered representation (affine):

James R. Bergen P. Anandan Keith J. Hanna Rajesh Hingorani “*Hierarchical Model-Based Motion Estimation*” ECCV'92, pp. 237-- 252



That's all for today