ECE 4973: Lecture 16
Optical Flow

Samuel Cheng

Slide credit: Juan Carlos Niebles and Ranjay Krishna

What we will learn today?

- Optical flow
- Lucas-Kanade method
- Horn-Schunk method
- Gunnar-Farneback method
- Pyramids for large motion
- Applications

Reading: [Szeliski] Chapters: 8.4, 8.5

[Fleet & Weiss, 2005]
http://www.cs.toronto.edu/pub/jepson/teaching/vision/2503/opticalFlow.pdf

What we will learn today?

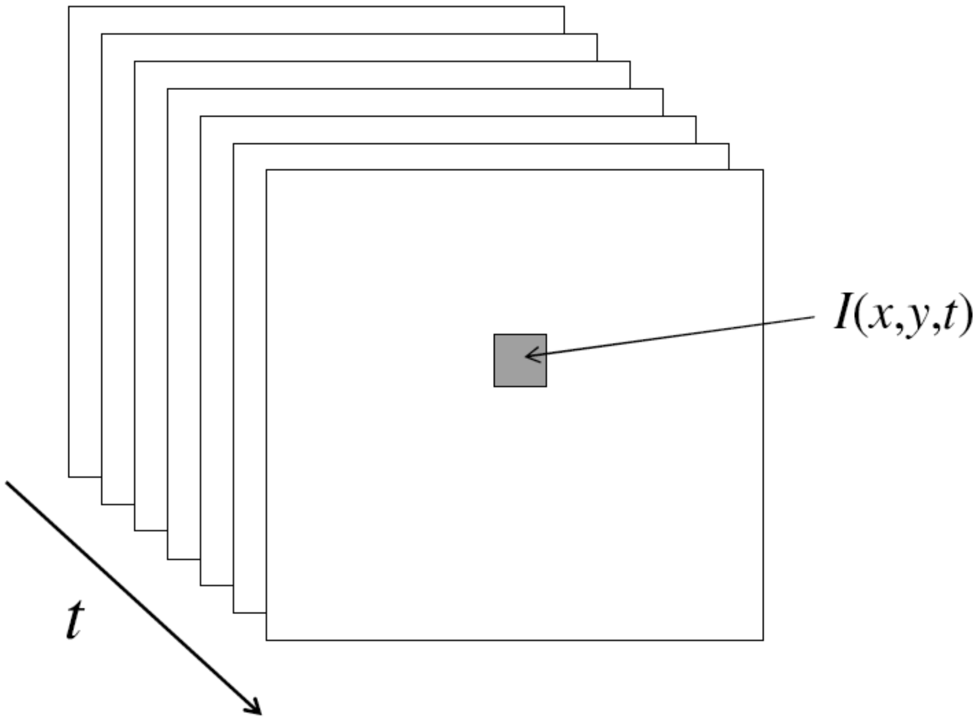
- Optical flow
- Lucas-Kanade method
- Horn-Schunk method
- Gunnar Farneback method
- Pyramids for large motion
- Applications

Reading: [Szeliski] Chapters: 8.4, 8.5

[Fleet & Weiss, 2005]
http://www.cs.toronto.edu/pub/jepson/teaching/vision/2503/opticalFlow.pdf

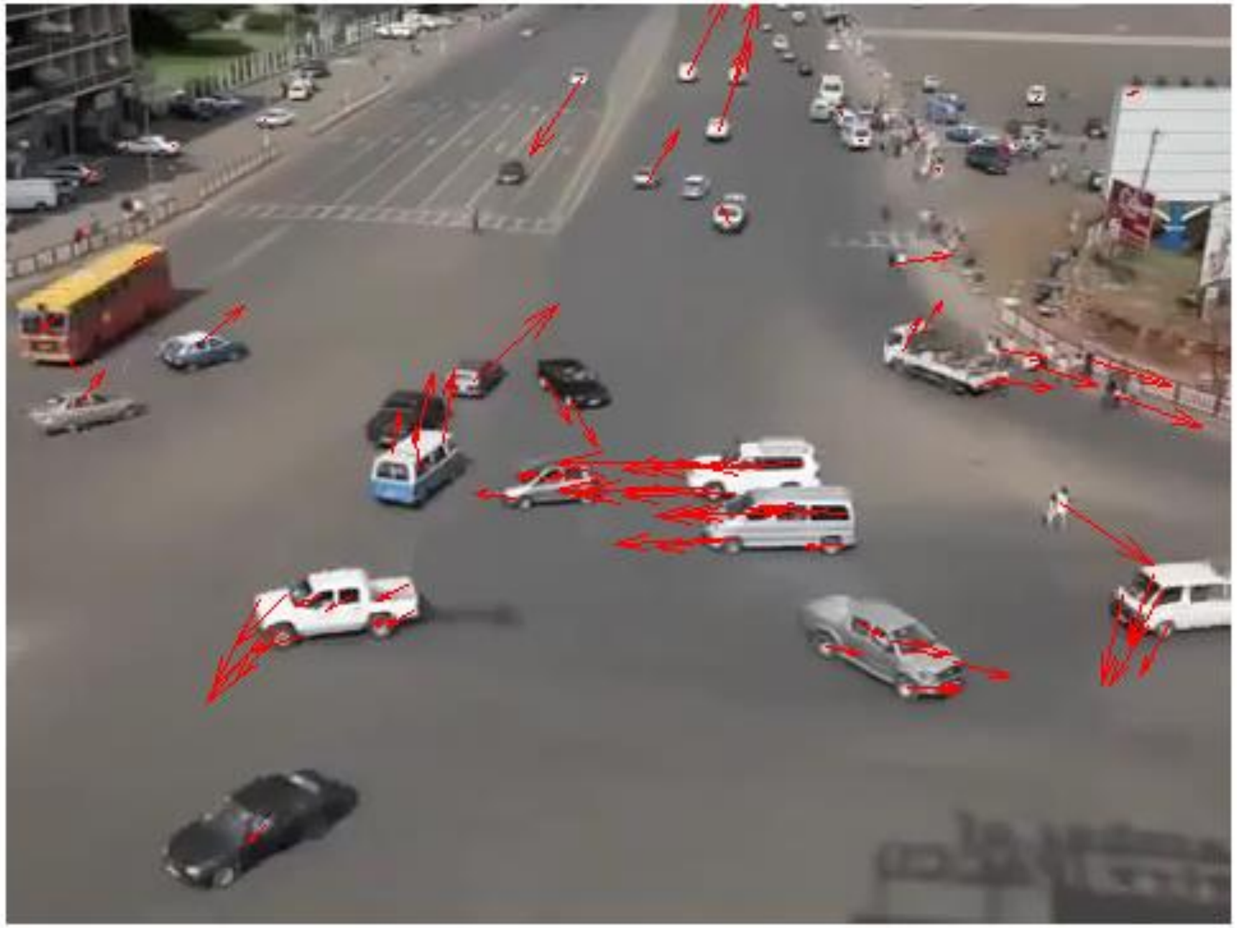
From images to videos

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



Why is motion useful?

Why is motion useful?



Optical flow

- Definition: optical flow is the *apparent* motion of brightness patterns in the image

Source: Silvio Savarese

Optical flow

- Definition: optical flow is the *apparent* motion of brightness patterns in the image

- Note: 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

Source: Silvio Savarese

Optical flow

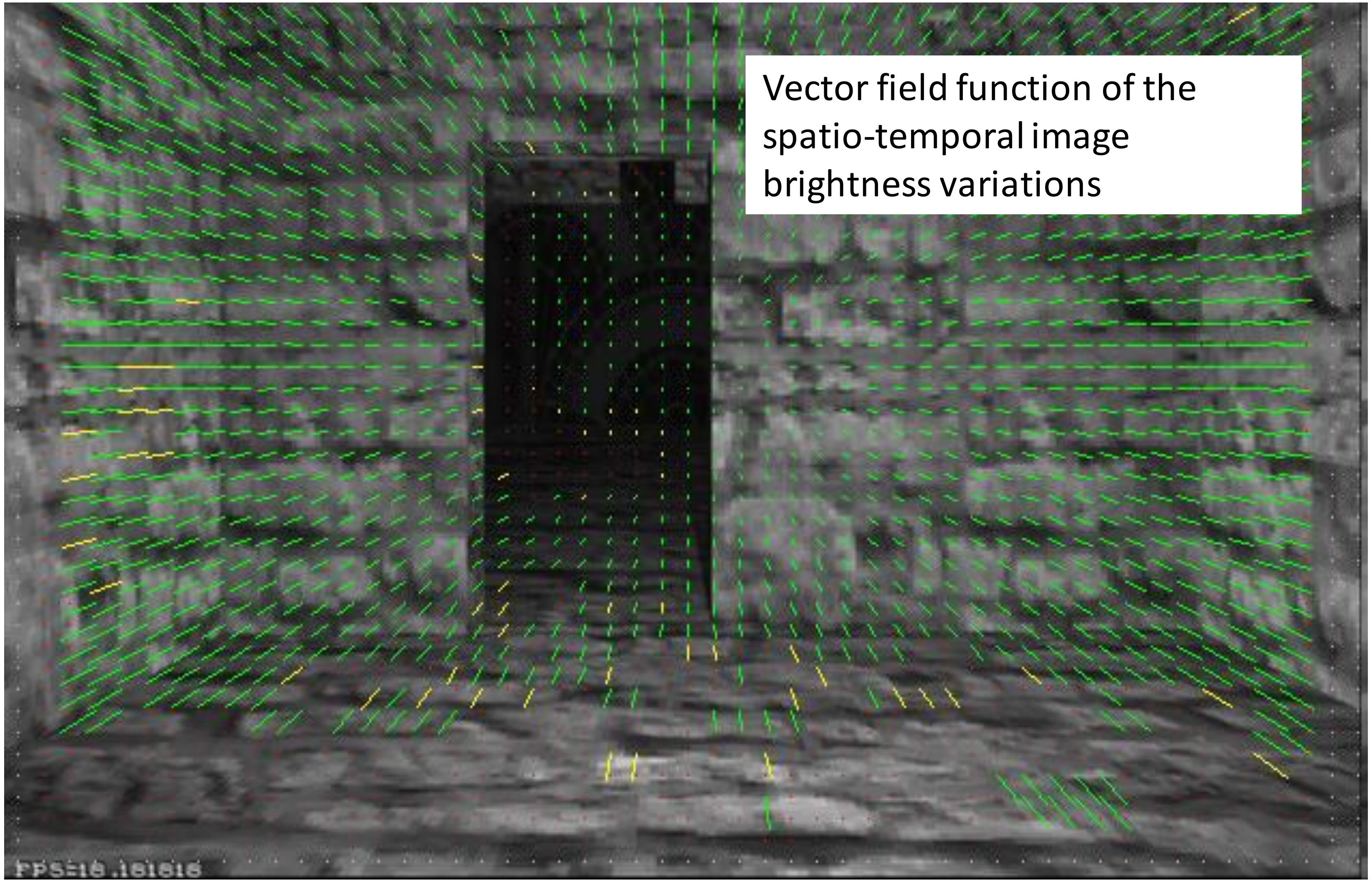
- Definition: optical flow is the *apparent* motion of brightness patterns in the image

- Note: 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

GOAL: Recover image motion at each pixel from optical flow

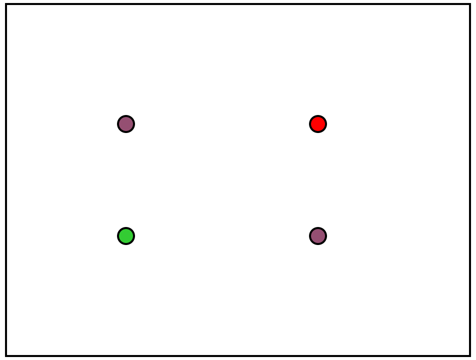
Source: Silvio Savarese

Optical flow



Picture courtesy of Selim Temizer - Learning and Intelligent Systems (LIS) Group, MIT

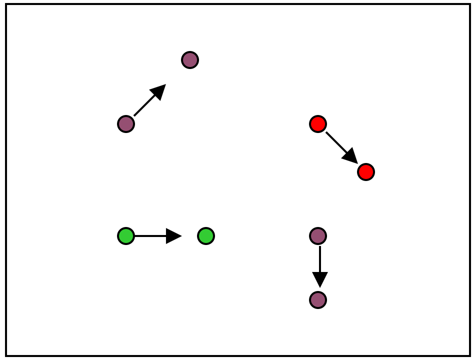
Estimating optical flow



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

Source: Silvio Savarese

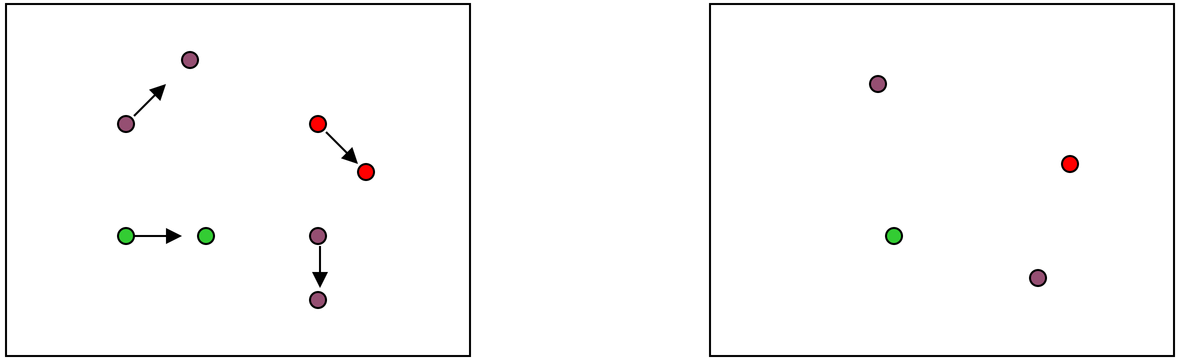
Estimating optical flow



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

Source: Silvio Savarese

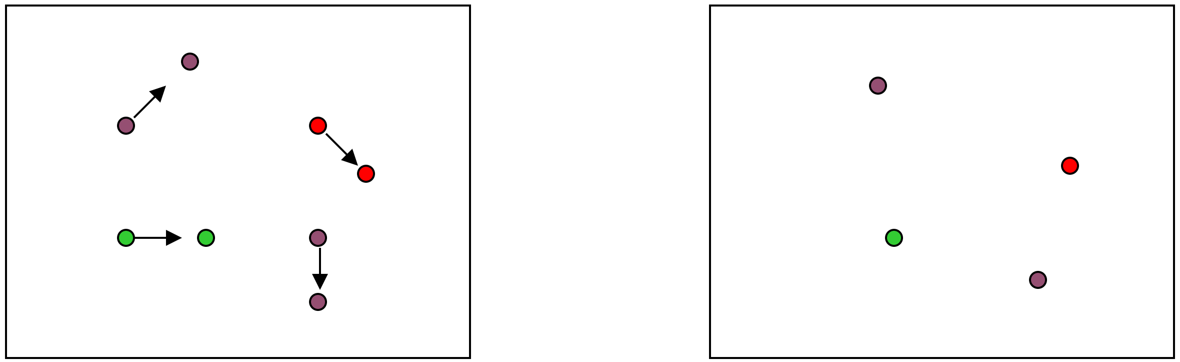
Estimating optical flow



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

Source: Silvio Savarese

Estimating optical flow

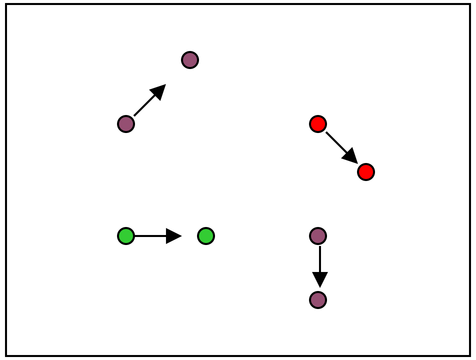


$I(x,y,t-1)$ $I(x,y,t)$

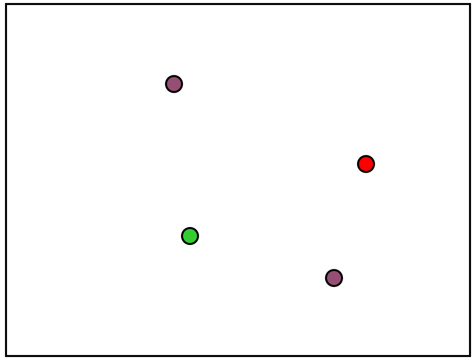
- Given two subsequent frames, estimate the apparent motion field u(x,y), v(x,y) between them

Source: Silvio Savarese

Estimating optical flow



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



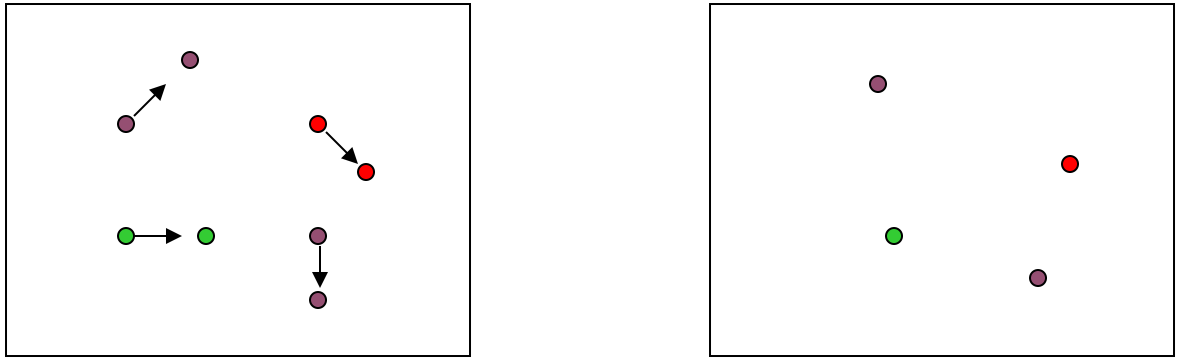
$I(x,y,t)$

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

- Key assumptions
  - **Brightness constancy:** projection of the same point looks the same in every frame

Source: Silvio Savarese

Estimating optical flow



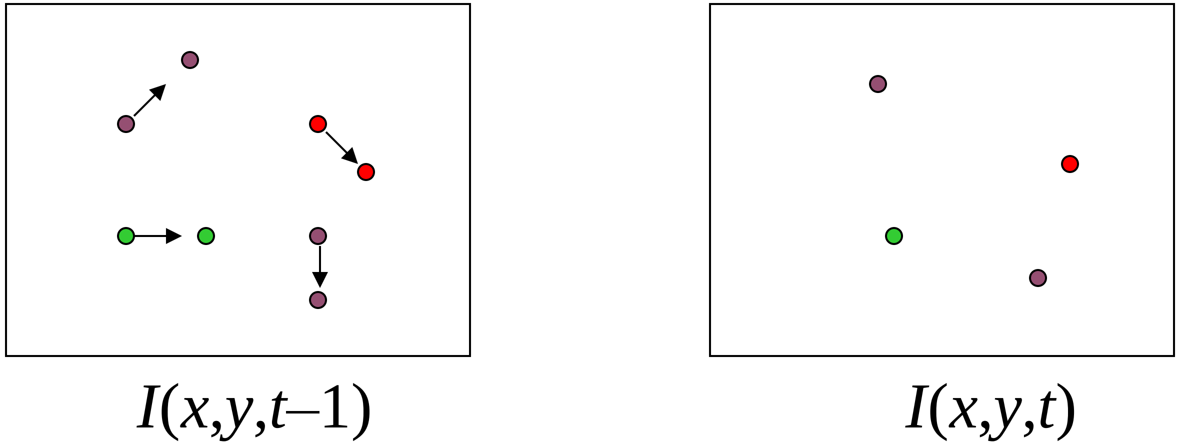
$I(x,y,t-1)$ $I(x,y,t)$

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

- Key assumptions
  - **Brightness constancy:** projection of the same point looks the same in every frame
  - **Small motion:** points do not move very far

Source: Silvio Savarese

Estimating optical flow

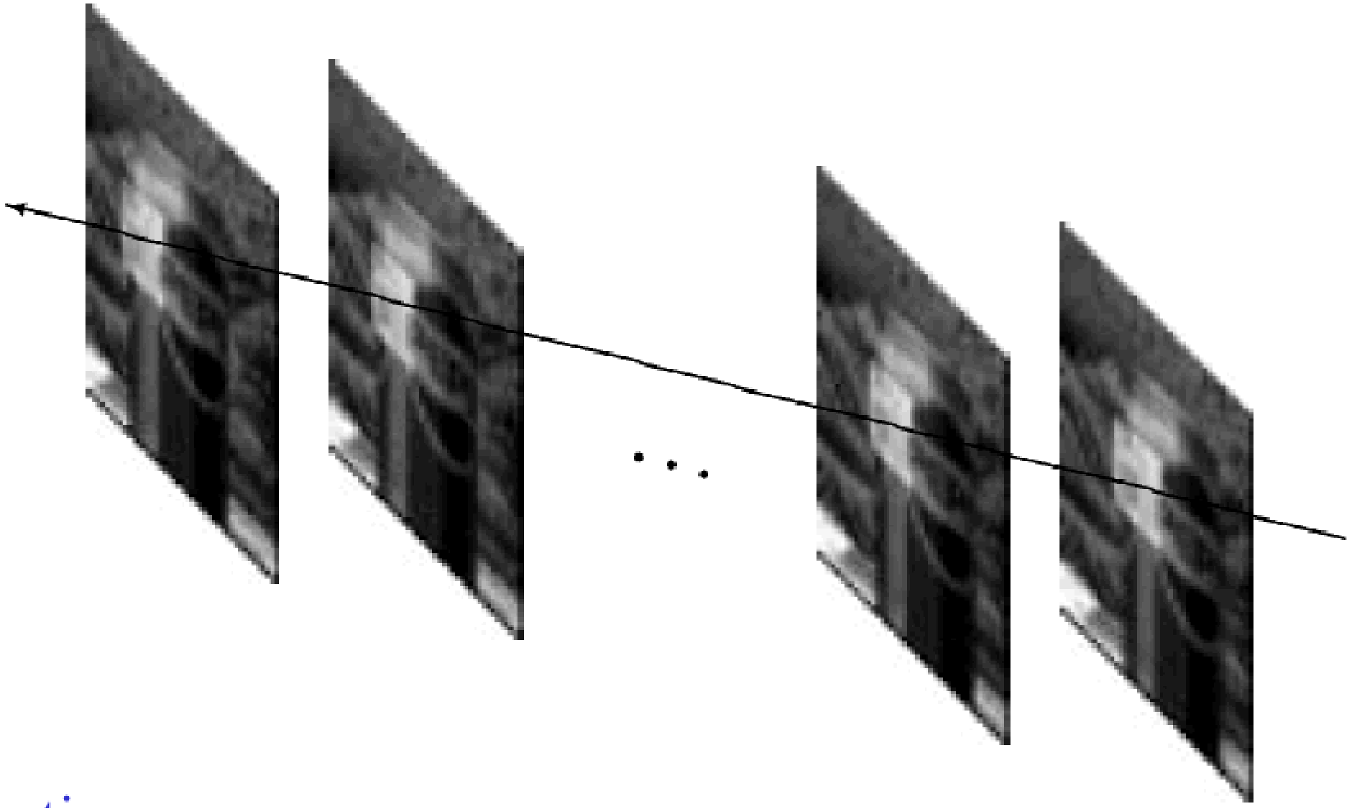


- Given two subsequent frames, estimate the apparent motion field u(x,y), v(x,y) between them

- 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

Source: Silvio Savarese

Key Assumptions: small motions

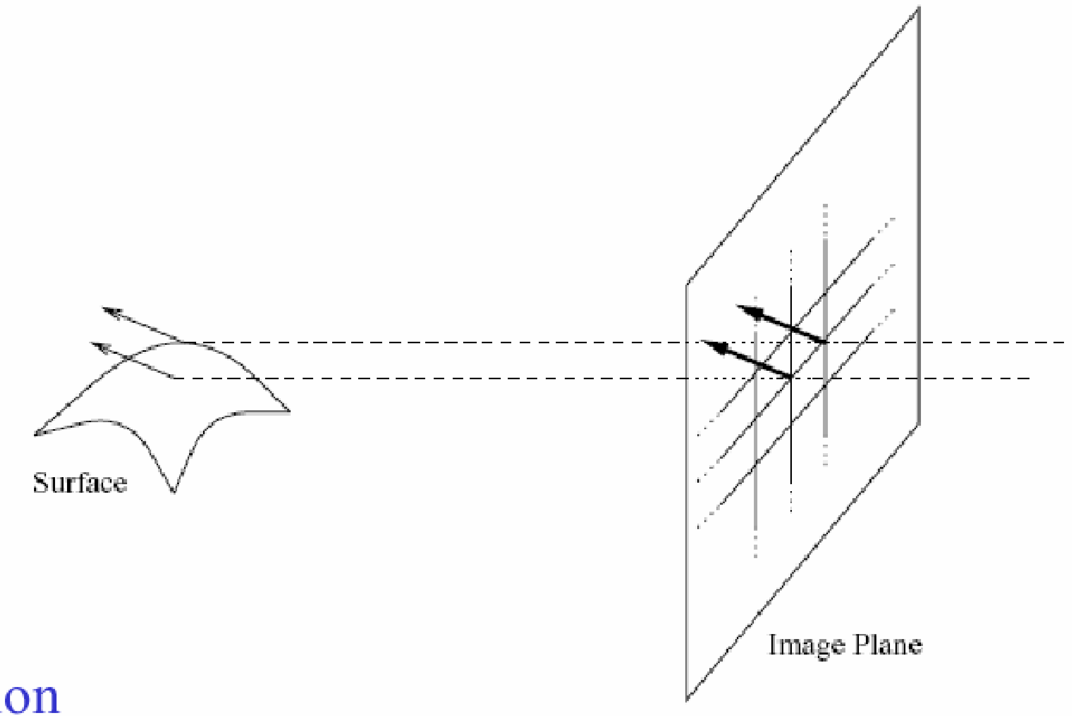


Assumption:

The image motion of a surface patch changes gradually over time.

* Slide from Michael Black, CS143 2003

Key Assumptions: spatial coherence

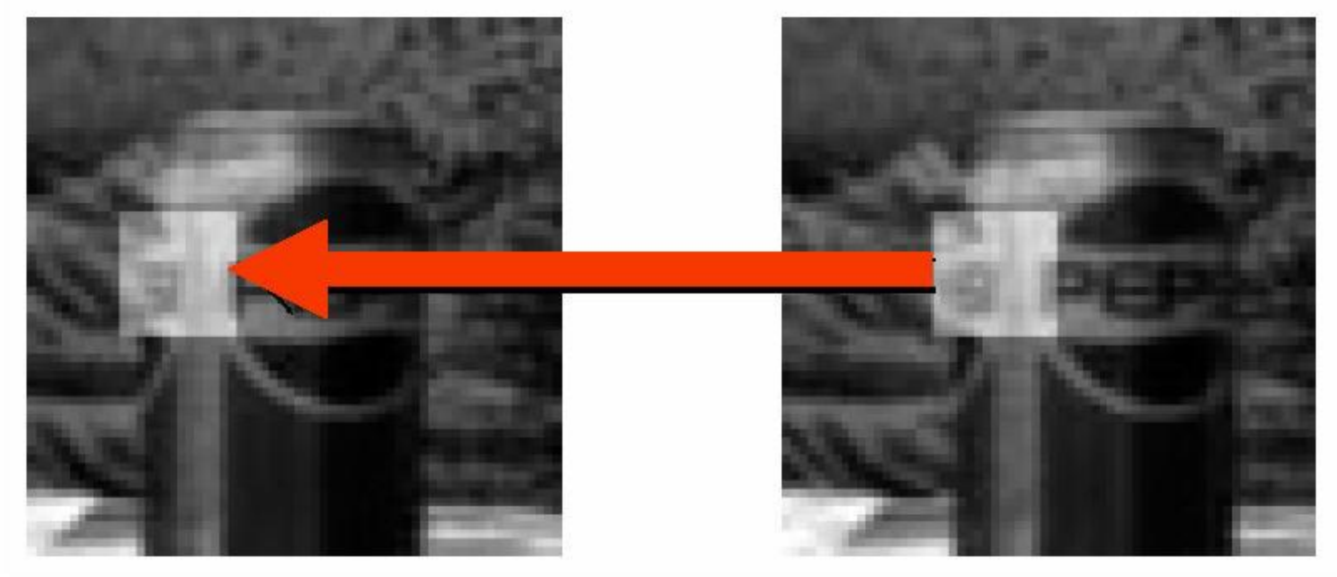


Assumption

* Neighboring points in the scene typically belong to the same surface and hence typically have similar motions.
* Since they also project to nearby points in the image, we expect spatial coherence in image flow.

* Slide from Michael Black, CS143 2003

Key Assumptions: brightness Constancy

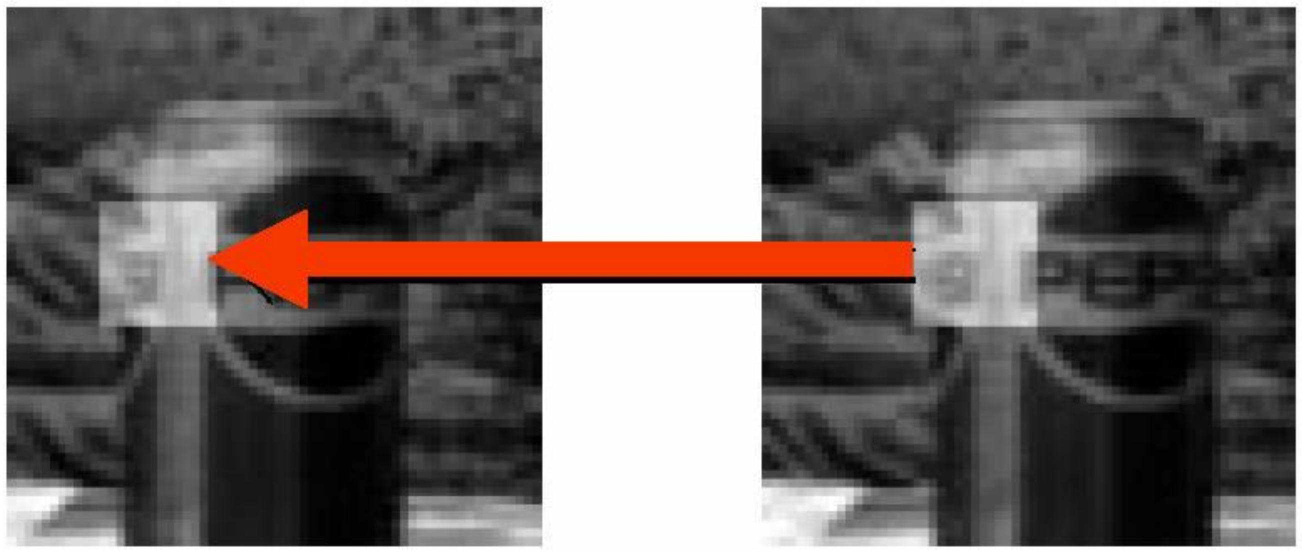


Assumption

Image measurements (e.g. brightness) in a small region remain the same although their location may change.

* Slide from Michael Black, CS143 2003

Key Assumptions: brightness Constancy



Assumption

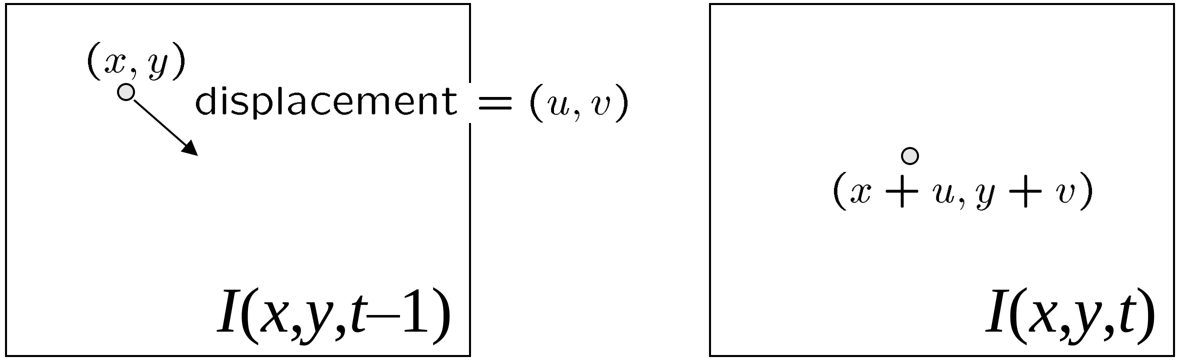
Image measurements (e.g. brightness) in a small region remain the same although their location may change.

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

(assumption)

* Slide from Michael Black, CS143 2003

The brightness constancy constraint

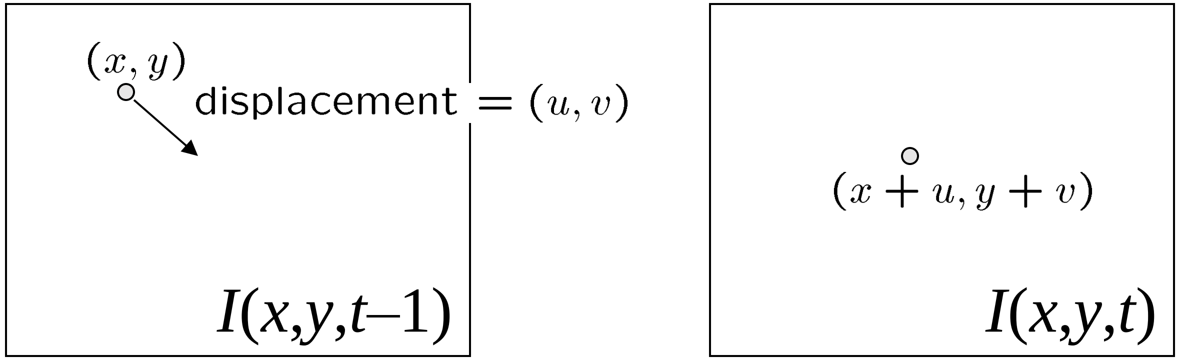


- Brightness Constancy Equation:

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

Source: Silvio Savarese

The brightness constancy constraint



- Brightness Constancy Equation:

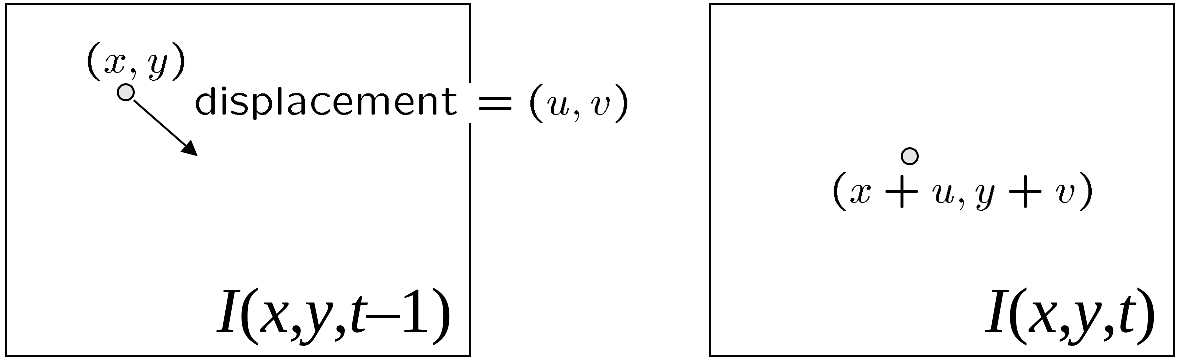
$$I(x, y, t-1) = I(x+u(x,y), y+v(x,y), t)$$

Linearizing the right side using Taylor expansion:

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

Source: Silvio Savarese

The brightness constancy constraint



- Brightness Constancy Equation:

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

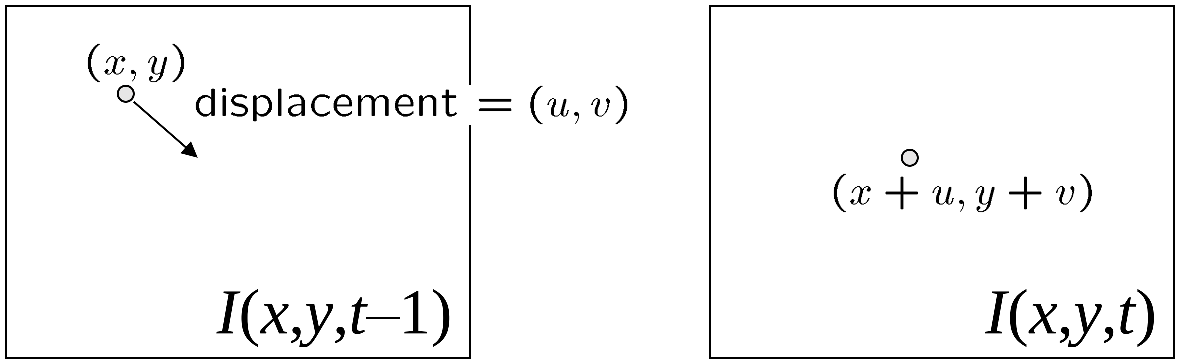
Linearizing the right side using Taylor expansion:

Image derivative along x

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

Source: Silvio Savarese

The brightness constancy constraint



- Brightness Constancy Equation:

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

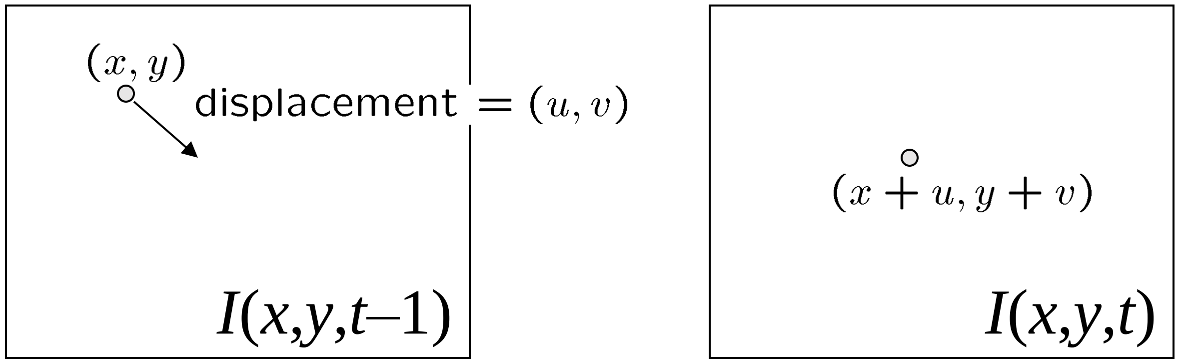
Linearizing the right side using Taylor expansion:

Image derivative along x

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

Source: Silvio Savarese

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:

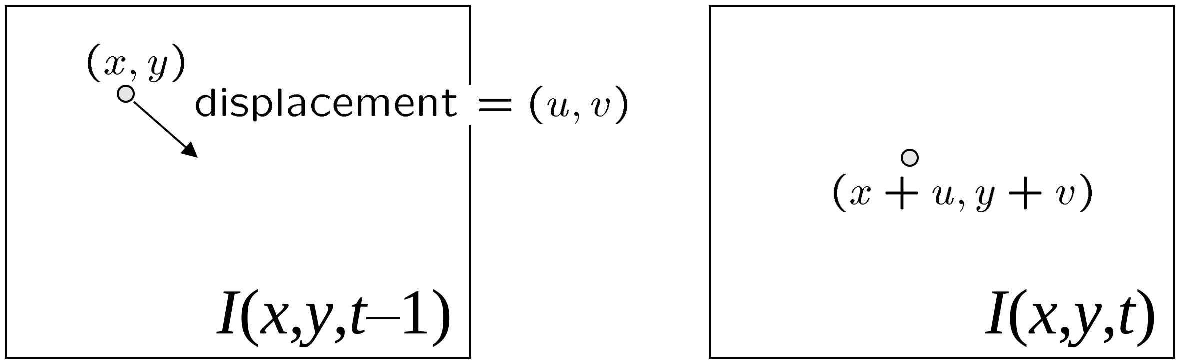
Image derivative along x

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

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

Source: Silvio Savarese

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:

Image derivative along x

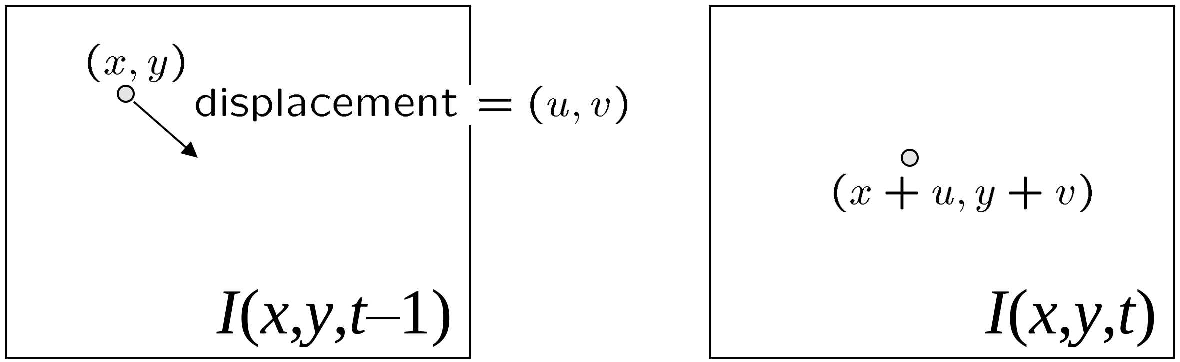
$$I(x+u, y+v, t) \approx I(x, y, t-1) + I_x \cdot u(x,y) + I_y \cdot v(x,y) + I_t$$

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

$$\rightarrow \nabla I^T [u \ v]^T + I_t = 0$$

Source: Silvio Savarese

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:

Image derivative along x

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

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

$$\rightarrow \nabla I^T [u \ v]^T + I_t = 0$$

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

Source: Silvio Savarese

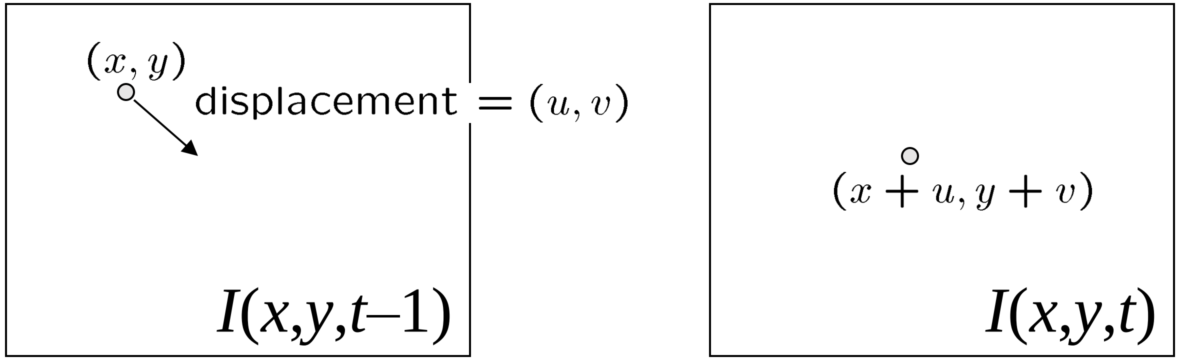
Filters used to find the derivatives

$$\begin{bmatrix}-1 & 1\\-1 & 1\end{bmatrix}\text{first image}\qquad \begin{bmatrix}-1 & -1\\1 & 1\end{bmatrix}\text{first image}\qquad \begin{bmatrix}-1 & -1\\-1 & -1\end{bmatrix}\text{first image}$$

$$\begin{bmatrix}-1 & 1\\-1 & 1\end{bmatrix}\text{second image}\qquad \begin{bmatrix}-1 & -1\\1 & 1\end{bmatrix}\text{second image}\qquad \begin{bmatrix}1 & 1\\1 & 1\end{bmatrix}\text{second image}$$

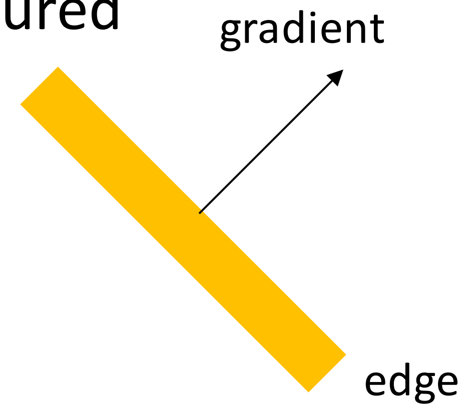
$$I_x \qquad\qquad I_y \qquad\qquad I_t$$

The brightness constancy constraint



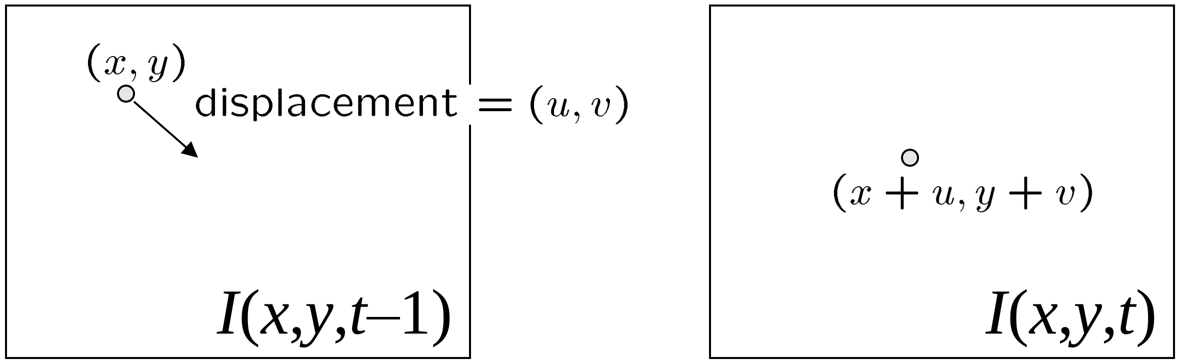
$$\nabla I^T [u \ v]^T + I_t = 0$$

- The component of the flow perpendicular to the gradient (i.e., parallel to the edge) cannot be measured



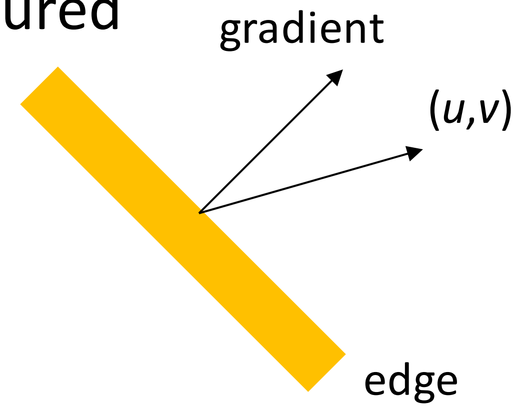
Source: Silvio Savarese

The brightness constancy constraint



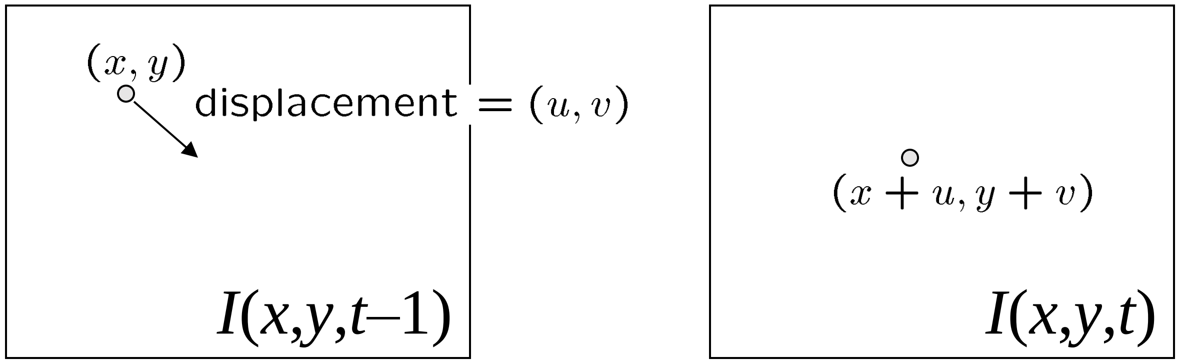
$$\nabla I^T [u \;\; v]^T + I_t = 0$$

- The component of the flow perpendicular to the gradient (i.e., parallel to the edge) cannot be measured



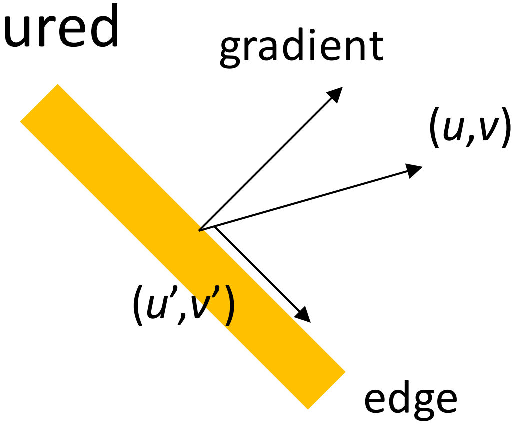
Source: Silvio Savarese

The brightness constancy constraint



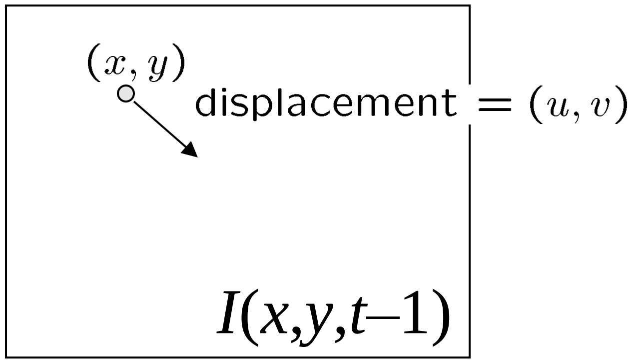
$$\nabla I^T [u \ \ v]^T + I_t = 0$$

- The component of the flow perpendicular to the gradient (i.e., parallel to the edge) cannot be measured



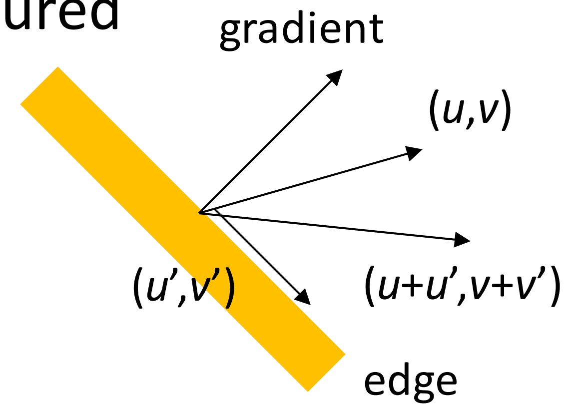
Source: Silvio Savarese

The brightness constancy constraint



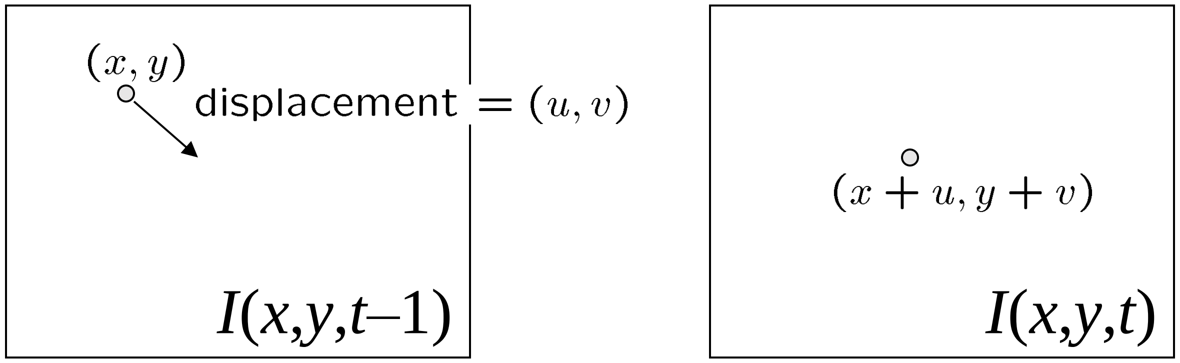
$$\nabla I^T [u \;\; v]^T + I_t = 0$$

- The component of the flow perpendicular to the gradient (i.e., parallel to the edge) cannot be measured



Source: Silvio Savarese

The brightness constancy constraint

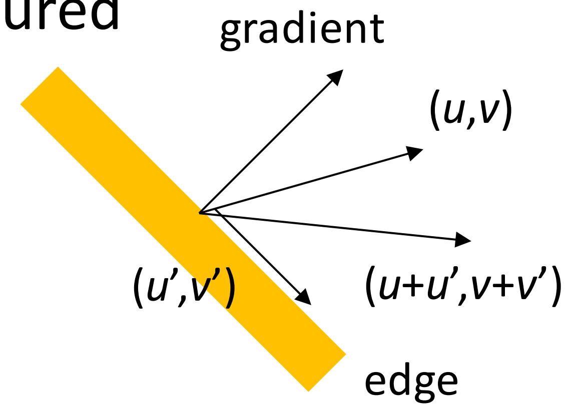


$$\nabla I^T [u \;\; v]^T + I_t = 0$$

- The component of the flow perpendicular to the gradient (i.e., parallel to the edge) cannot be measured

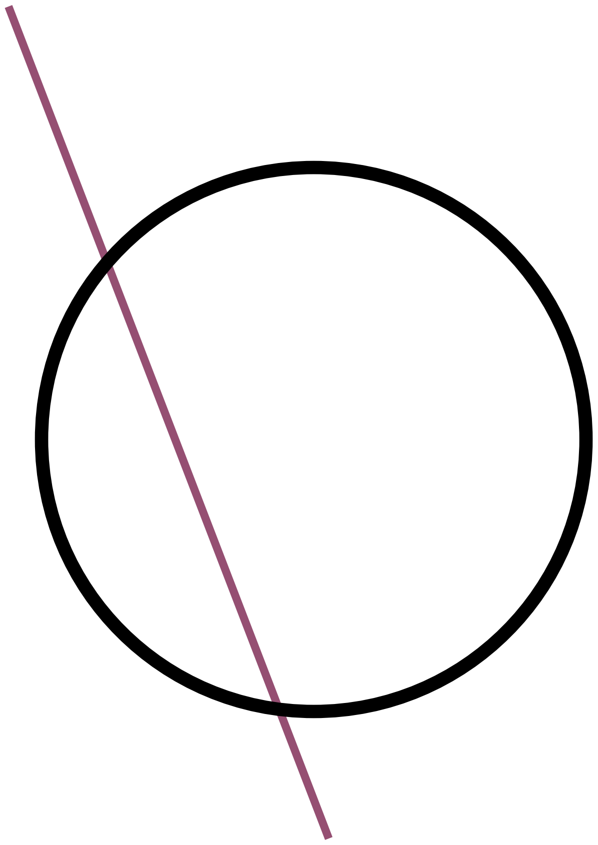
If (*u*, *v*) satisfies the equation,
so does (*u+u'*, *v+v'*) if

$$\nabla I^T [u' \;\; v']^T = 0$$



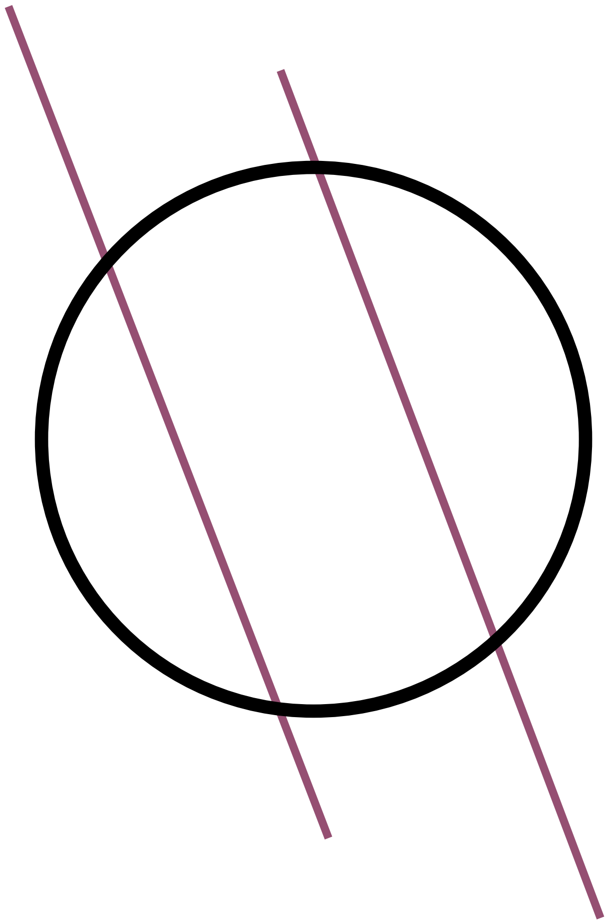
Source: Silvio Savarese

The aperture problem



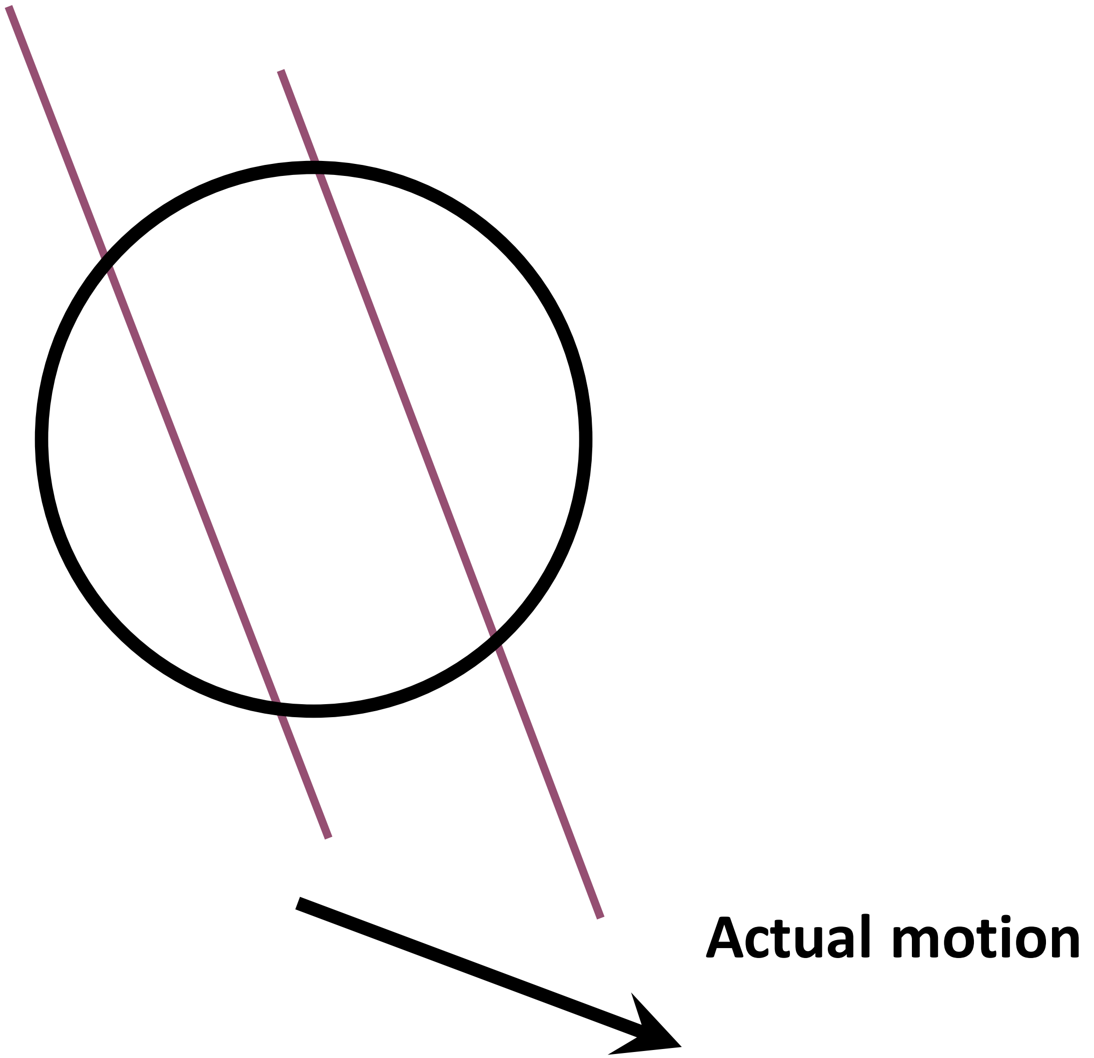
Source: Silvio Savarese

The aperture problem



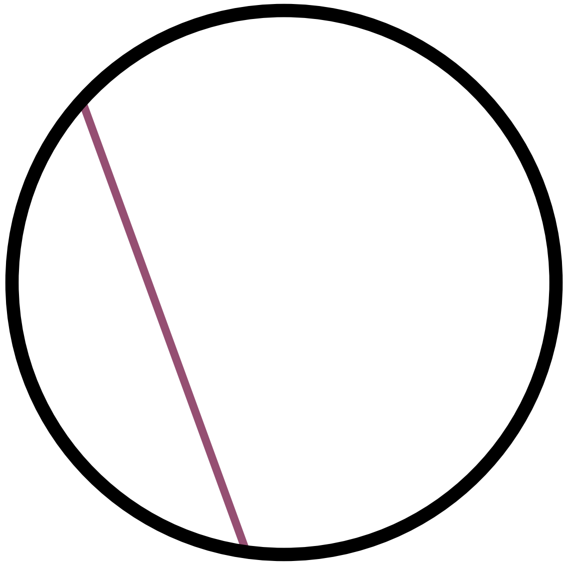
Source: Silvio Savarese

The aperture problem



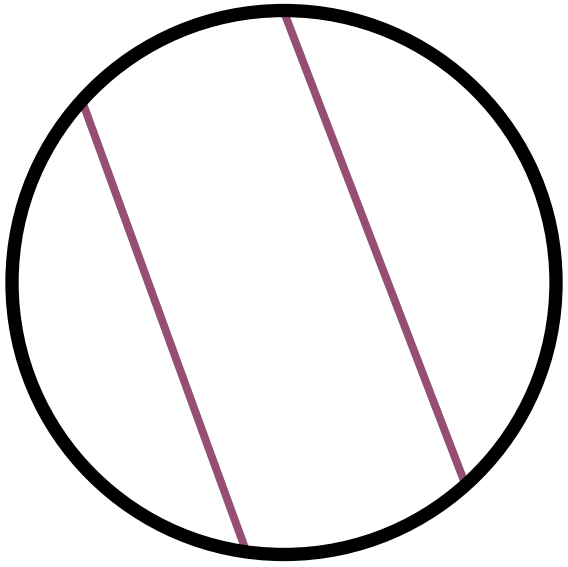
Source: Silvio Savarese

The aperture problem



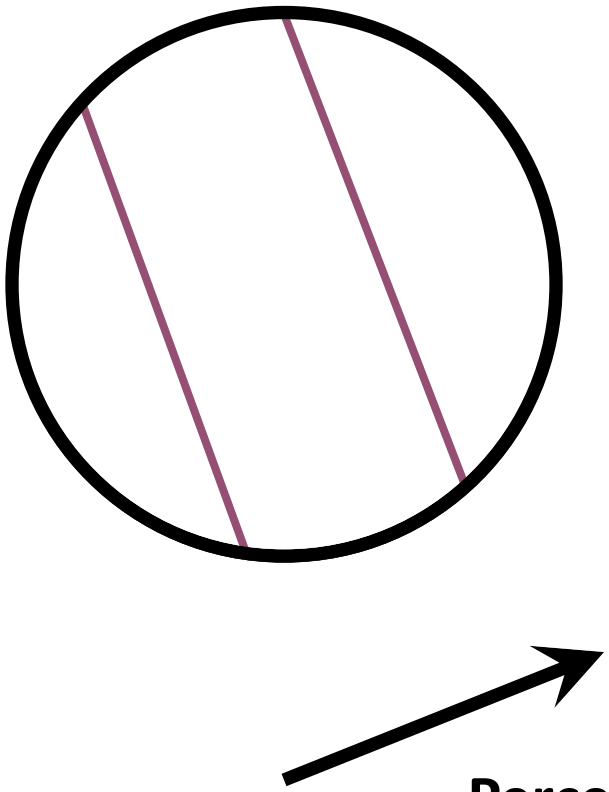
Source: Silvio Savarese

The aperture problem



Source: Silvio Savarese

The aperture problem



Perceived motion

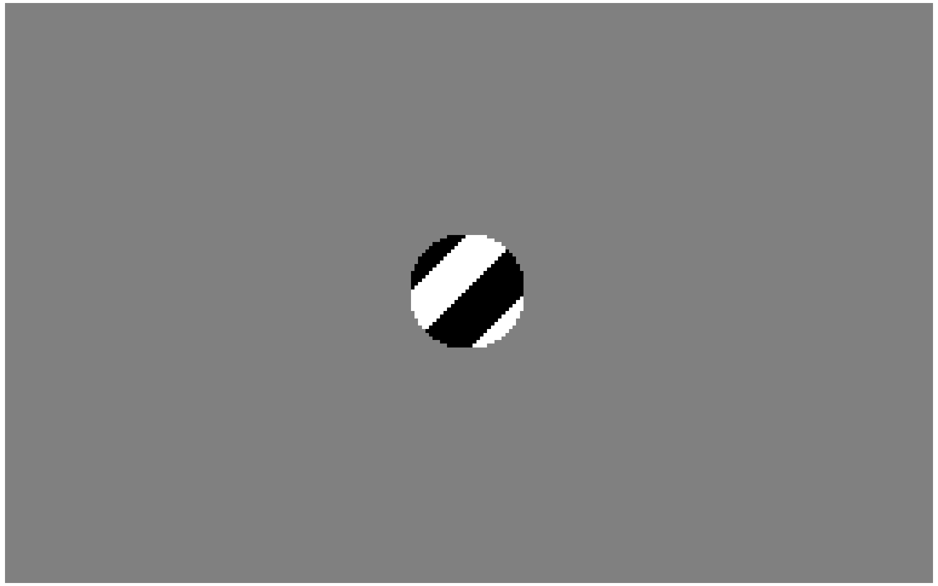
Source: Silvio Savarese

The barber pole illusion

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

Source: Silvio Savarese

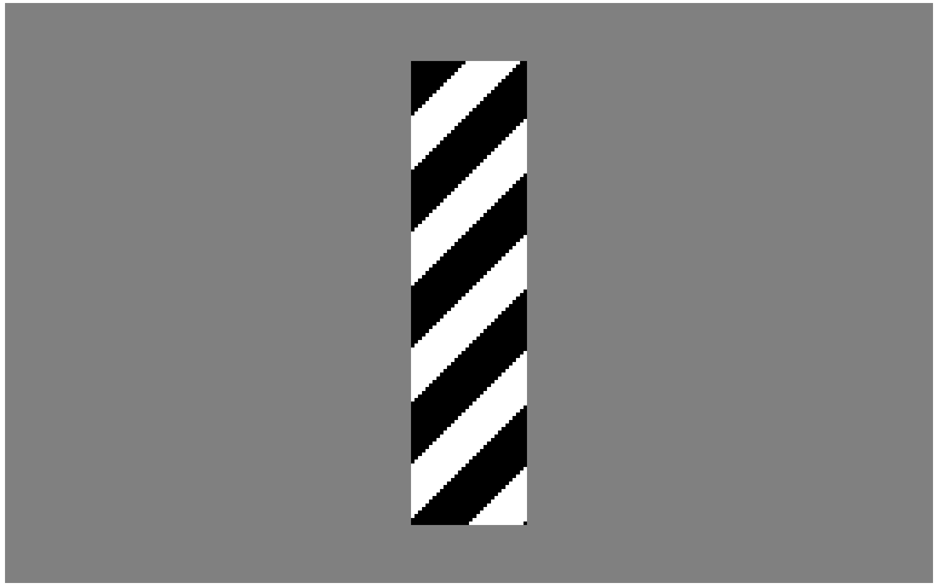
The barber pole illusion



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

Source: Silvio Savarese

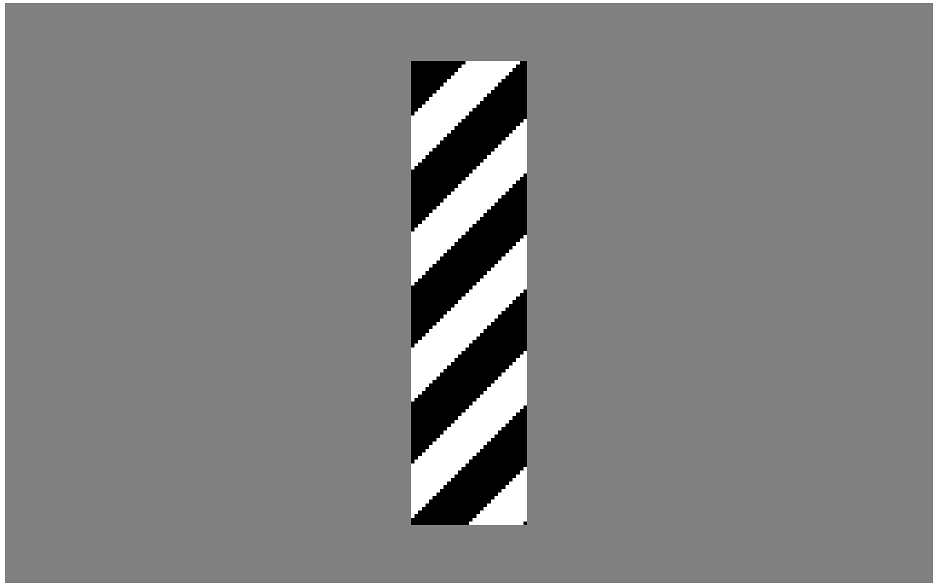
The barber pole illusion



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

Source: Silvio Savarese

The barber pole illusion



Source: Silvio Savarese

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

The brightness constancy constraint

- Can we use this equation to recover image motion (u,v) at each pixel?

$$\nabla I^T [u \ v]^T + I_t = 0$$

- How many equations and unknowns per pixel?
    - One equation (this is a scalar equation!), two unknowns (u,v)

What we will learn today?

- Optical flow
- Lucas-Kanade method
- Horn-Schunk method
- Gunnar Farneback method
- Pyramids for large motion
- Applications

Reading: [Szeliski] Chapters: 8.4, 8.5

[Fleet & Weiss, 2005]
http://www.cs.toronto.edu/pub/jepson/teaching/vision/2503/opticalFlow.pdf

Solving the ambiguity…

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.

Source: Silvio Savarese

Solving the ambiguity…

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.

- How to get more equations for a pixel?

Source: Silvio Savarese

Solving the ambiguity…

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.

- How to get more equations for a pixel?
- **Spatial coherence constraint:**

Source: Silvio Savarese

Solving the ambiguity…

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.

- How to get more equations for a pixel?
- **Spatial coherence constraint:**
- Assume the pixel's neighbors have the same (u,v)
  - If we use a 5x5 window, that gives us 25 equations per pixel

$$\nabla I(\mathrm{p_i})^T \begin{bmatrix} u \\ v \end{bmatrix} + I_t(\mathrm{p_i}) = 0$$

Source: Silvio Savarese

Solving the ambiguity…

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.

- How to get more equations for a pixel?
- **Spatial coherence constraint:**
- Assume the pixel's neighbors have the same (u,v)
  - If we use a 5x5 window, that gives us 25 equations per pixel

$$\nabla I(\mathrm{p_i})^T \begin{bmatrix} u \\ v \end{bmatrix} + I_t(\mathrm{p_i}) = 0$$

$$\begin{bmatrix} I_x(\mathbf{p_1}) & I_y(\mathbf{p_1}) \\ I_x(\mathbf{p_2}) & I_y(\mathbf{p_2}) \\ \vdots & \vdots \\ I_x(\mathbf{p_{25}}) & I_y(\mathbf{p_{25}}) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(\mathbf{p_1}) \\ I_t(\mathbf{p_2}) \\ \vdots \\ I_t(\mathbf{p_{25}}) \end{bmatrix}$$

Source: Silvio Savarese

Lucas-Kanade flow

- Overconstrained linear system:

$$\begin{bmatrix} I_x(\mathbf{p}_1) & I_y(\mathbf{p}_1) \\ I_x(\mathbf{p}_2) & I_y(\mathbf{p}_2) \\ \vdots & \vdots \\ I_x(\mathbf{p}_{25}) & I_y(\mathbf{p}_{25}) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(\mathbf{p}_1) \\ I_t(\mathbf{p}_2) \\ \vdots \\ I_t(\mathbf{p}_{25}) \end{bmatrix}$$

$$\underset{25\times2}{A} \; \underset{2\times1}{d} = \underset{25\times1}{b}$$

Source: Silvio Savarese

Lucas-Kanade flow

- Overconstrained linear system

$$\begin{bmatrix} I_x(\mathbf{p_1}) & I_y(\mathbf{p_1}) \\ I_x(\mathbf{p_2}) & I_y(\mathbf{p_2}) \\ \vdots & \vdots \\ I_x(\mathbf{p_{25}}) & I_y(\mathbf{p_{25}}) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(\mathbf{p_1}) \\ I_t(\mathbf{p_2}) \\ \vdots \\ I_t(\mathbf{p_{25}}) \end{bmatrix} \qquad \underset{25\times2}{A} \; \underset{2\times1}{d} = \underset{25\times1}{b}$$

Least squares solution for *d* given by $(A^T A) \; d = A^T b$

Source: Silvio Savarese

Lucas-Kanade flow

- Overconstrained linear system

$$\begin{bmatrix} I_x(\mathbf{p_1}) & I_y(\mathbf{p_1}) \\ I_x(\mathbf{p_2}) & I_y(\mathbf{p_2}) \\ \vdots & \vdots \\ I_x(\mathbf{p_{25}}) & I_y(\mathbf{p_{25}}) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(\mathbf{p_1}) \\ I_t(\mathbf{p_2}) \\ \vdots \\ I_t(\mathbf{p_{25}}) \end{bmatrix} \qquad \underset{25\text{x}2}{A} \; \underset{2\text{x}1}{d} = \underset{25\text{x}1}{b}$$

Least squares solution for *d* given by $(A^T A)\ d = A^T b$

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

The summations are over all pixels in the K x K window

Source: Silvio Savarese

Conditions for solvability

- Optimal (u, v) satisfies Lucas-Kanade equation

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

$$A^T A \qquad\qquad A^T b$$

Source: Silvio Savarese

Conditions for solvability

- Optimal (u, v) satisfies Lucas-Kanade equation

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

$$A^T A \qquad\qquad\qquad A^T b$$

When is This Solvable?

- **$A^T A$** should be invertible
- **$A^T A$** should not be too small due to noise
  - eigenvalues λ_1 and λ_2 of **$A^T A$** should not be too small
- **$A^T A$** should be well-conditioned
  - λ_1 / λ_2 should not be too large (λ_1 = larger eigenvalue)

Source: Silvio Savarese

Conditions for solvability

- Optimal (u, v) satisfies Lucas-Kanade equation

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

$$A^T A \qquad\qquad A^T b$$

When is This Solvable?

- **$A^T A$** should be invertible
- **$A^T A$** should not be too small due to noise
  - eigenvalues λ_1 and λ_2 of **$A^T A$** should not be too small
- **$A^T A$** should be well-conditioned
  - λ_1/λ_2 should not be too large (λ_1 = larger eigenvalue)

Does this remind anything to you?

Source: Silvio Savarese

M = A^{T}A is the *second moment matrix* !

(Harris corner detector…)

$$A^T A = \begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix} = \sum \begin{bmatrix} I_x \\ I_y \end{bmatrix} [I_x \ I_y] = \sum \nabla I (\nabla I)^T$$

Source: Silvio Savarese

M = A^TA is the *second moment matrix* !

(Harris corner detector…)

$$A^T A = \left[\begin{array}{cc} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{array} \right] = \sum \left[\begin{array}{c} I_x \\ I_y \end{array} \right] [I_x \ I_y] = \sum \nabla I (\nabla I)^T$$

- Eigenvectors and eigenvalues of A^TA relate to edge direction and magnitude

Source: Silvio Savarese

M = $A^T A$ is the *second moment matrix* !

(Harris corner detector…)

$$A^T A = \begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix} = \sum \begin{bmatrix} I_x \\ I_y \end{bmatrix} [I_x \ I_y] = \sum \nabla I (\nabla I)^T$$

- Eigenvectors and eigenvalues of $A^T A$ relate to edge direction and magnitude
  - The eigenvector associated with the larger eigenvalue points in the direction of fastest intensity change

Source: Silvio Savarese

M = $A^T A$ is the *second moment matrix* !

(Harris corner detector…)

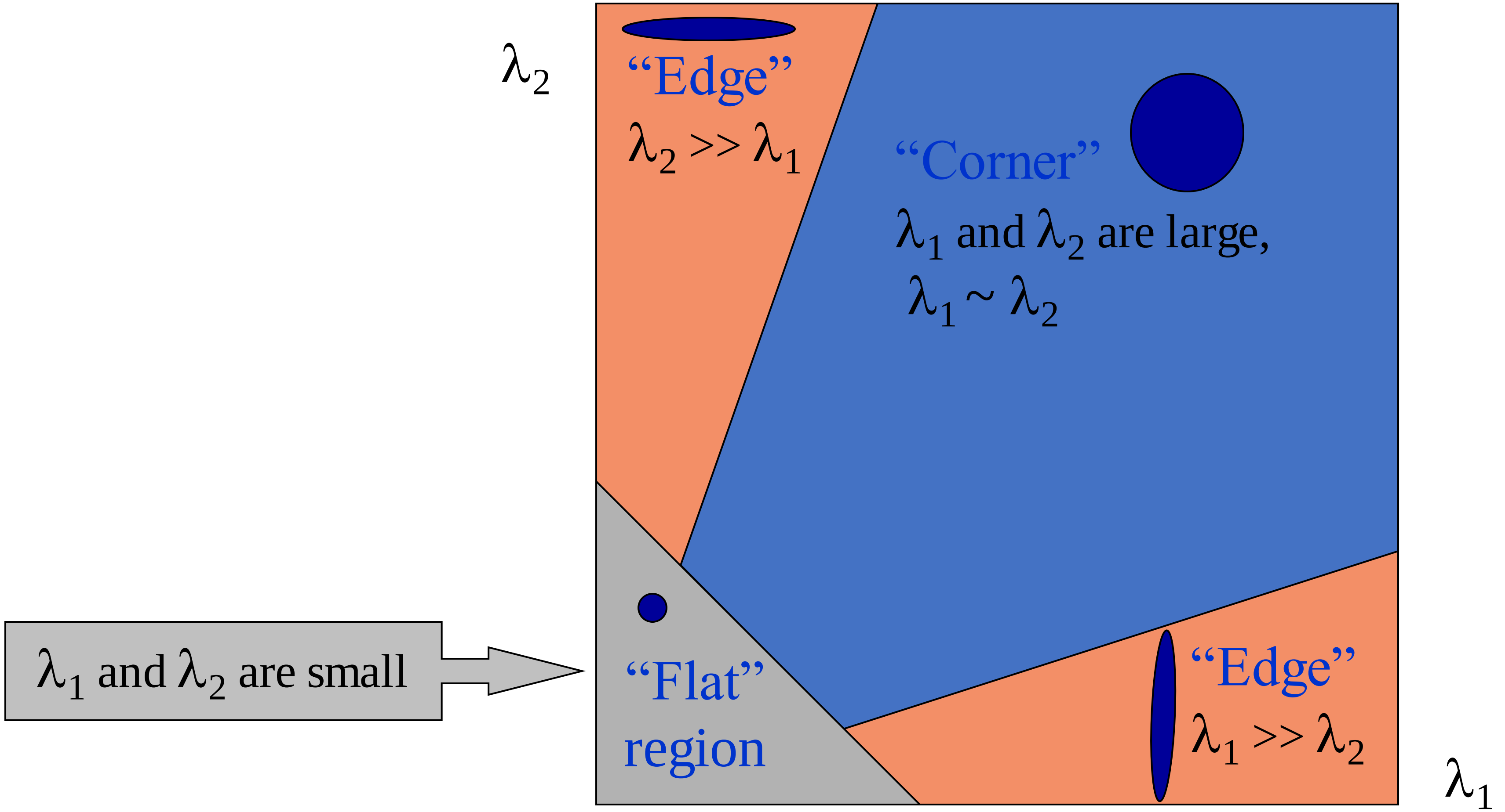
$$A^T A = \begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix} = \sum \begin{bmatrix} I_x \\ I_y \end{bmatrix} [I_x \ I_y] = \sum \nabla I (\nabla I)^T$$

- Eigenvectors and eigenvalues of $A^T A$ relate to edge direction and magnitude
  - The eigenvector associated with the larger eigenvalue points in the direction of fastest intensity change
  - The other eigenvector is orthogonal to it

Source: Silvio Savarese

Interpreting the eigenvalues

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



Source: Silvio Savarese

Edge

$$\sum \nabla I (\nabla I)^T$$

– gradients very large or very small
– large λ_1, small λ_2

Source: Silvio Savarese

Low-texture region

$$\sum \nabla I (\nabla I)^T$$

– gradients have small magnitude

– small λ_1, small λ_2

Source: Silvio Savarese

High-texture region

$$\sum \nabla I (\nabla I)^T$$

– gradients are different, large magnitudes

– large λ_1, large λ_2

Source: Silvio Savarese

Errors in Lucas-Kanade

What are the potential causes of errors in this procedure?

- Suppose A^TA is easily invertible
- Suppose there is not much noise in the image

* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Errors in Lucas-Kanade

Inherent assumptions of this procedure

- Suppose A^TA is easily invertible
- Suppose there is not much noise in the image

- When our assumptions are violated
  - Brightness constancy is **not** satisfied
  - The motion is **not** small
  - A point does **not** move like its neighbors
    - window size is too large

* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

What we will learn today?

- Optical flow
- Lucas-Kanade method
- Horn-Schunk method
- Gunnar Farneback method
- Pyramids for large motion
- Applications

Reading: [Szeliski] Chapters: 8.4, 8.5

[Fleet & Weiss, 2005]
http://www.cs.toronto.edu/pub/jepson/teaching/vision/2503/opticalFlow.pdf

Horn-Schunk method for optical flow

- The flow is formulated as a global energy function which is should be minimized:

$$E = \iint \left[(I_x u + I_y v + I_t)^2 + \alpha^2 (\|\nabla u\|^2 + \|\nabla v\|^2) \right] \mathrm{d}x \mathrm{d}y$$

Horn-Schunk method for optical flow

- The flow is formulated as a global energy function which is should be minimized:

$$E = \iint \left[(I_x u + I_y v + I_t)^2 + \alpha^2 (\|\nabla u\|^2 + \|\nabla v\|^2) \right] \mathrm{d}x \mathrm{d}y$$

- The first part of the function is the brightness consistency.

Horn-Schunk method for optical flow

- The flow is formulated as a global energy function which is should be minimized:

$$E = \iint \left[(I_x u + I_y v + I_t)^2 + \alpha^2 \left(\|\nabla u\|^2 + \|\nabla v\|^2 \right) \right] \mathrm{d}x \mathrm{d}y$$

- The second part is the smoothness constraint. It's trying to make sure that the changes between frames are small.

Horn-Schunk method for optical flow

- The flow is formulated as a global energy function which is should be minimized:

$$E = \iint \left[(I_x u + I_y v + I_t)^2 + \alpha^2 (\|\nabla u\|^2 + \|\nabla v\|^2) \right] \mathrm{d}x\mathrm{d}y$$

- α is a regularization constant. Larger values of α lead to smoother flow.

Horn-Schunk method for optical flow

- The flow is formulated as a global energy function which is should be minimized:

$$E = \iint \underbrace{\left[(I_x u + I_y v + I_t)^2 + \alpha^2 (\|\nabla u\|^2 + \|\nabla v\|^2)\right]}_{L(u, v, u_x, u_y, v_x, v_y)} \mathrm{d}x \mathrm{d}y$$

- We want to find u, v to minimize E. Note that u, v themselves are function. E is a "functional" of u, v. By calculus of variation, as ϵ→0, for arbitrary $\tilde{u}(x, y), \tilde{v}(x, y)$

$$\frac{1}{\epsilon}\left[E\left(u + \epsilon\tilde{u}, v + \epsilon\tilde{v}, u_x + \epsilon\tilde{u}_x, u_y + \epsilon\tilde{u}_y, v_x + \epsilon\tilde{v}_x, v_y + \epsilon\tilde{v}_y\right) - E\left(u, v, u_x, u_y, v_x, v_y\right)\right] = 0$$

Euler Lagrange equation (1-D case)

- $E(u) = \int L(u, u_x)dx$
- If u is an extremum, $\frac{E(u+\epsilon\tilde{u})-E(u)}{\epsilon} = 0$ for any $\tilde{u}$

Euler Lagrange equation (1-D case)

- $E(u) = \int L(u, u_x)dx$
- If u is an extremum, $\frac{E(u+\epsilon\tilde{u})-E(u)}{\epsilon} = 0$ for any $\tilde{u}$

$$\Rightarrow \frac{1}{\epsilon}\int L(u + \epsilon\tilde{u}, u_x + \epsilon\tilde{u}_x) - L(u, u_x)dx = 0$$

Euler Lagrange equation (1-D case)

- $E(u) = \int L(u, u_x)dx$
- If u is an extremum, $\frac{E(u+\epsilon\tilde{u})-E(u)}{\epsilon} = 0$ for any $\tilde{u}$

$$\Rightarrow \frac{1}{\epsilon}\int L(u + \epsilon\tilde{u}, u_x + \epsilon\tilde{u}_x) - L(u, u_x)dx = 0$$

$$\Rightarrow \int \frac{\partial L}{\partial u}\tilde{u} + \frac{\partial L}{\partial u_x}\tilde{u}_x dx = 0$$

Euler Lagrange equation (1-D case)

- $E(u) = \int L(u, u_x)dx$
- If u is an extremum, $\frac{E(u+\epsilon\tilde{u})-E(u)}{\epsilon} = 0$ for any $\tilde{u}$

$$\Rightarrow \frac{1}{\epsilon}\int L(u + \epsilon\tilde{u}, u_x + \epsilon\tilde{u}_x) - L(u, u_x)dx = 0$$

$$\Rightarrow \int \frac{\partial L}{\partial u}\tilde{u} + \frac{\partial L}{\partial u_x}\tilde{u}_x dx = 0$$

$$\Rightarrow \int \tilde{u}\left(\frac{\partial L}{\partial u} - \frac{\partial}{\partial x}\frac{\partial L}{\partial u_x}\right)dx = 0$$

Euler Lagrange equation (1-D case)

- $E(u) = \int L(u, u_x)dx$
- If u is an extremum, $\frac{E(u+\epsilon\tilde{u})-E(u)}{\epsilon} = 0$ for any $\tilde{u}$

$$\Rightarrow \frac{1}{\epsilon}\int L(u + \epsilon\tilde{u}, u_x + \epsilon\tilde{u}_x) - L(u, u_x)dx = 0$$

$$\Rightarrow \int \frac{\partial L}{\partial u}\tilde{u} + \frac{\partial L}{\partial u_x}\tilde{u}_x dx = 0$$

$$\Rightarrow \int \tilde{u}\left(\frac{\partial L}{\partial u} - \frac{\partial}{\partial x}\frac{\partial L}{\partial u_x}\right)dx = 0$$

$$\Rightarrow \frac{\partial L}{\partial u} - \frac{\partial}{\partial x}\frac{\partial L}{\partial u_x} = 0$$

Euler Lagrange equation (2-D case)

- $E(u, v) = \iint L(u, v, u_x, u_y, v_x, v_y) dx\, dy$

Euler Lagrange equation (2-D case)

- $E(u, v) = \iint L(u, v, u_x, u_y, v_x, v_y) dx\, dy$
- If (u, v) is an extremum, $\frac{E(u+\epsilon\tilde{u}, v+\epsilon\tilde{v}) - E(u,v)}{\epsilon} = 0$ for any $(\tilde{u}, \tilde{v}) \Rightarrow \cdots$

Euler Lagrange equation (2-D case)

- $E(u,v) = \iint L(u, v, u_x, u_y, v_x, v_y) dx\, dy$
- If (u,v) is an extremum, $\frac{E(u+\epsilon\tilde{u}, v+\epsilon\tilde{v}) - E(u,v)}{\epsilon} = 0$ for any $(\tilde{u}, \tilde{v}) \Rightarrow \cdots$

$$\Rightarrow \begin{cases} \frac{\partial L}{\partial u} - \frac{\partial}{\partial x}\frac{\partial L}{\partial u_x} - \frac{\partial}{\partial y}\frac{\partial L}{\partial u_y} = 0 \\ \frac{\partial L}{\partial v} - \frac{\partial}{\partial x}\frac{\partial L}{\partial v_x} - \frac{\partial}{\partial y}\frac{\partial L}{\partial v_y} = 0 \end{cases}$$

Horn-Schunk method for optical flow

$$E = \iint \underbrace{\left[(I_x u + I_y v + I_t)^2 + \alpha^2 (\|\nabla u\|^2 + \|\nabla v\|^2)\right]}_{L(u,v,u_x,u_y,v_x,v_y)} \mathrm{d}x\mathrm{d}y$$

Horn-Schunk method for optical flow

$$E = \iint \underbrace{\left[(I_x u + I_y v + I_t)^2 + \alpha^2 (\|\nabla u\|^2 + \|\nabla v\|^2)\right]}_{L(u,v,u_x,u_y,v_x,v_y)} \mathrm{d}x\mathrm{d}y$$

$$\text{E-L: } \begin{cases} \dfrac{\partial L}{\partial u} - \dfrac{\partial}{\partial x}\dfrac{\partial L}{\partial u_x} - \dfrac{\partial}{\partial y}\dfrac{\partial L}{\partial u_y} = 0 \\ \dfrac{\partial L}{\partial v} - \dfrac{\partial}{\partial x}\dfrac{\partial L}{\partial v_x} - \dfrac{\partial}{\partial y}\dfrac{\partial L}{\partial v_y} = 0 \end{cases}$$

Horn-Schunk method for optical flow

$$E = \iint \underbrace{\left[(I_x u + I_y v + I_t)^2 + \alpha^2(\|\nabla u\|^2 + \|\nabla v\|^2)\right]}_{L(u,v,u_x,u_y,v_x,v_y)} \mathrm{d}x\mathrm{d}y$$

E-L: $$\begin{cases} \frac{\partial L}{\partial u} - \frac{\partial}{\partial x}\frac{\partial L}{\partial u_x} - \frac{\partial}{\partial y}\frac{\partial L}{\partial u_y} = 0 \\ \frac{\partial L}{\partial v} - \frac{\partial}{\partial x}\frac{\partial L}{\partial v_x} - \frac{\partial}{\partial y}\frac{\partial L}{\partial v_y} = 0 \end{cases}$$

$$\frac{\partial L}{\partial u} = 2(I_x u + I_y v + I_t)I_x$$

$$\frac{\partial L}{\partial v} = 2(I_x u + I_y v + I_t)I_y$$

$$\frac{\partial L}{\partial u_x} = 2\alpha^2 u_x, \quad \frac{\partial L}{\partial u_y} = 2\alpha^2 u_y$$

$$\frac{\partial L}{\partial v_x} = 2\alpha^2 v_x, \quad \frac{\partial L}{\partial v_y} = 2\alpha^2 v_y$$

Horn-Schunk method for optical flow

$$E = \iint \underbrace{\left[(I_x u + I_y v + I_t)^2 + \alpha^2 (\|\nabla u\|^2 + \|\nabla v\|^2)\right]}_{L(u,v,u_x,u_y,v_x,v_y)} \mathrm{d}x\mathrm{d}y$$

$$\frac{\partial L}{\partial u} = 2(I_x u + I_y v + I_t) I_x$$

$$\frac{\partial L}{\partial v} = 2(I_x u + I_y v + I_t) I_y$$

$$\frac{\partial L}{\partial u_x} = 2\alpha^2 u_x, \quad \frac{\partial L}{\partial u_y} = 2\alpha^2 u_y$$

$$\frac{\partial L}{\partial v_x} = 2\alpha^2 v_x, \quad \frac{\partial L}{\partial v_y} = 2\alpha^2 v_y$$

E-L: $\begin{cases} \frac{\partial L}{\partial u} - \frac{\partial}{\partial x}\frac{\partial L}{\partial u_x} - \frac{\partial}{\partial y}\frac{\partial L}{\partial u_y} = 0 \\ \frac{\partial L}{\partial v} - \frac{\partial}{\partial x}\frac{\partial L}{\partial v_x} - \frac{\partial}{\partial y}\frac{\partial L}{\partial v_y} = 0 \end{cases}$

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

Horn-Schunk method for optical flow

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

Horn-Schunk method for optical flow

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

- Where the Laplace operator can be often computed as

$$\nabla^2 u(x, y) = \bar{u}(x, y) - u(x, y)$$

where $\overline{u}(x, y)$ is the weighted average of *u* measured at (*x*,*y*).

Horn-Schunk method for optical flow

- Now we substitute in:

$$\nabla^2 u(x,y) = \bar{u}(x,y) - u(x,y)$$

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

Horn-Schunk method for optical flow

- Now we substitute in: $\nabla^2 u(x, y) = \bar{u}(x, y) - u(x, y)$

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

- We get:

$$(I_x^2 + \alpha^2)u + I_x I_y v = \alpha^2 \bar{u} - I_x I_t$$
$$I_x I_y u + (I_y^2 + \alpha^2)v = \alpha^2 \bar{v} - I_y I_t$$

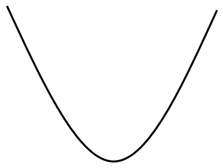
- Which is linear in u and v and can be solved for each pixel individually.

Dense Optical Flow with Michael Black's method

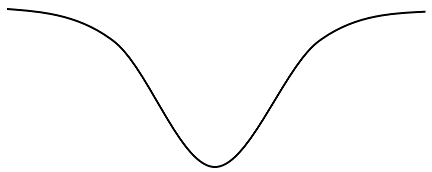
$$E = \iint \left[(I_x u + I_y v + I_t)^2 + \alpha^2 (\|\nabla u\|^2 + \|\nabla v\|^2) \right] \mathrm{d}x\mathrm{d}y$$

- Michael Black took Horn-Schunk's method one step further, starting from the regularization constant:
- Which looks like a quadratic:

$$\|\nabla u\|^2 + \|\nabla v\|^2$$



- And replaced it with this:



- Why does this regularization work better?

What we will learn today?

- Optical flow
- Lucas-Kanade method
- Horn-Schunk method
- Gunnar Farneback method
- Pyramids for large motion
- Applications

Gunnar Farneback Optical Flow

- Model image intensity with quadratic function
- Image 1:
- Image 2:

Gunnar Farneback Optical Flow

- Model image intensity with quadratic function
- Image 1:

$$f_1(\mathbf{x}) = \mathbf{x}^T A\mathbf{x} + {b_1}^T\mathbf{x} + c_1$$

- Image 2:

Gunnar Farneback Optical Flow

- Model image intensity with quadratic function
- Image 1:

$$f_1(\mathbf{x}) = \mathbf{x}^T A\mathbf{x} + {b_1}^T \mathbf{x} + c_1$$

- Image 2:

$$f_2(\mathbf{x}) = f_1(\mathbf{x} + \mathbf{d}) = (\mathbf{x} - \mathbf{d})^T A(\mathbf{x} - \mathbf{d}) + {b_1}^T (\mathbf{x} - \mathbf{d}) + c_1$$

Gunnar Farneback Optical Flow

- Model image intensity with quadratic function
- Image 1:

$$f_1(\mathbf{x}) = \mathbf{x}^T A\mathbf{x} + {b_1}^T \mathbf{x} + c_1$$

- Image 2:

$$f_2(\mathbf{x}) = f_1(\mathbf{x} + \mathbf{d}) = (\mathbf{x} - \mathbf{d})^T A(\mathbf{x} - \mathbf{d}) + {b_1}^T (\mathbf{x} - \mathbf{d}) + c_1$$

$$= \mathbf{x}^T A\mathbf{x} + \underbrace{(b_1 - 2A_1\mathbf{d})^T}_{b_2^T}\mathbf{x} + \cdots$$

Gunnar Farneback Optical Flow

- Model image intensity with quadratic function
- Image 1:

$$f_1(\mathbf{x}) = \mathbf{x}^T A\mathbf{x} + {b_1}^T \mathbf{x} + c_1$$

- Image 2:

$$f_2(\mathbf{x}) = f_1(\mathbf{x} + \mathbf{d}) = (\mathbf{x} - \mathbf{d})^T A(\mathbf{x} - \mathbf{d}) + {b_1}^T (\mathbf{x} - \mathbf{d}) + c_1$$

$$= \mathbf{x}^T A\mathbf{x} + \underbrace{(b_1 - 2A_1\mathbf{d})^T}_{b_2^T}\mathbf{x} + \cdots$$

$$\Rightarrow b_2 = b_1 - 2A\mathbf{d}$$

Gunnar Farneback Optical Flow

- Model image intensity with quadratic function
- Image 1:

$$f_1(\mathbf{x}) = \mathbf{x}^T A\mathbf{x} + {b_1}^T\mathbf{x} + c_1$$

- Image 2:

$$f_2(\mathbf{x}) = f_1(\mathbf{x}+\mathbf{d}) = (\mathbf{x}-\mathbf{d})^T A(\mathbf{x}-\mathbf{d}) + {b_1}^T(\mathbf{x}-\mathbf{d}) + c_1$$

$$= \mathbf{x}^T A\mathbf{x} + \underbrace{(b_1 - 2A\mathbf{d})^T}_{b_2^T}\mathbf{x} + \cdots$$

$$\Rightarrow b_2 = b_1 - 2A\mathbf{d}$$

$$\Rightarrow \mathbf{d} = -\frac{1}{2}A^{-1}(b_2 - b_1) = A^{-1}\Delta\mathrm{b} \qquad \Delta b = -\frac{1}{2}(b_2 - b_1)$$

Gunnar Farneback Optical Flow

- A's and b's should vary with location. Thus

$$A(\mathbf{x}) = \frac{A_1(\mathbf{x}) + A_2(\mathbf{x})}{2}$$

$$\Delta b(\mathbf{x}) = -\frac{1}{2}(b_2(\mathbf{x}) - b_1(\mathbf{x}))$$

$$A(\mathbf{x})\mathbf{d}(\mathbf{x}) = \Delta b(\mathbf{x})$$

Gunnar Farneback Optical Flow

- A's and b's should vary with location. Thus

$$A(\mathbf{x}) = \frac{A_1(\mathbf{x}) + A_2(\mathbf{x})}{2}$$

$$\Delta b(\mathbf{x}) = -\frac{1}{2}(b_2(\mathbf{x}) - b_1(\mathbf{x}))$$

$$A(\mathbf{x})\mathbf{d}(\mathbf{x}) = \Delta b(\mathbf{x})$$

- Consider a window instead, and minimizes

$$\sum_{\Delta\mathbf{x}\in\mathcal{N}} w(\Delta\mathbf{x})\|A(\mathbf{x}+\Delta\mathbf{x})d(\mathbf{x}) - \Delta b(\mathbf{x}+\Delta\mathbf{x})\|^2$$

$$\mathbf{d}(\mathbf{x}) = \left(\sum wA^TA\right)^{-1}\sum wA^T\Delta b$$

Iterative update

- Assume previous some a priori displacement field $\tilde{\mathbf{d}}(\mathbf{x})$

$$A(\mathbf{x}) = \frac{A_1(\mathbf{x}) + A_2(\mathbf{x} + \tilde{\mathbf{d}}(\mathbf{x}))}{2}$$

$$\Delta b(\mathbf{x}) = -\frac{1}{2}(b_2(\mathbf{x}) - b_1(\mathbf{x})) + A(\mathbf{x})\tilde{\mathbf{d}}(\mathbf{x})$$

$$\mathbf{d}(\mathbf{x}) \leftarrow A(\mathbf{x})^{-1}\Delta b(\mathbf{x})$$

What we will learn today?

- Optical flow
- Lucas-Kanade method
- Horn-Schunk method
- Gunnar Farneback method
- Pyramids for large motion
- Applications

Recap

Source: Silvio Savarese

Recap

- Key assumptions (Errors in Lucas-Kanade)

  - **Small motion:** points do not move very far

Source: Silvio Savarese

Recap

- Key assumptions (Errors in Lucas-Kanade)
  - **Small motion:** points do not move very far
  - **Brightness constancy:** projection of the same point looks the same in every frame

Source: Silvio Savarese

Recap

- Key assumptions (Errors in Lucas-Kanade)

  - **Small motion:** points do not move very far
  - **Brightness constancy:** projection of the same point looks the same in every frame
  - **Spatial coherence:** points move like their neighbors

Source: Silvio Savarese

Recap

- Key assumptions (Errors in Lucas-Kanade)
  - **Small motion:** points do not move very far
  - **Brightness constancy:** projection of the same point looks the same in every frame
  - **Spatial coherence:** points move like their neighbors

Source: Silvio Savarese

Recap

- Key assumptions (Errors in Lucas-Kanade)

  - **Small motion:** points do not move very far
  - **Brightness constancy:** projection of the same point looks the same in every frame
  - **Spatial coherence:** points move like their neighbors

Source: Silvio Savarese

Revisiting the small motion assumption

* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Revisiting the small motion assumption

- Is this motion small enough?
  - Probably not—it's much larger than one pixel (2nd order terms dominate)
  - How might we solve this problem?

* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Revisiting the small motion assumption

- Is this motion small enough?
  - Probably not—it's much larger than one pixel (2^{nd} order terms dominate)
  - How might we solve this problem?

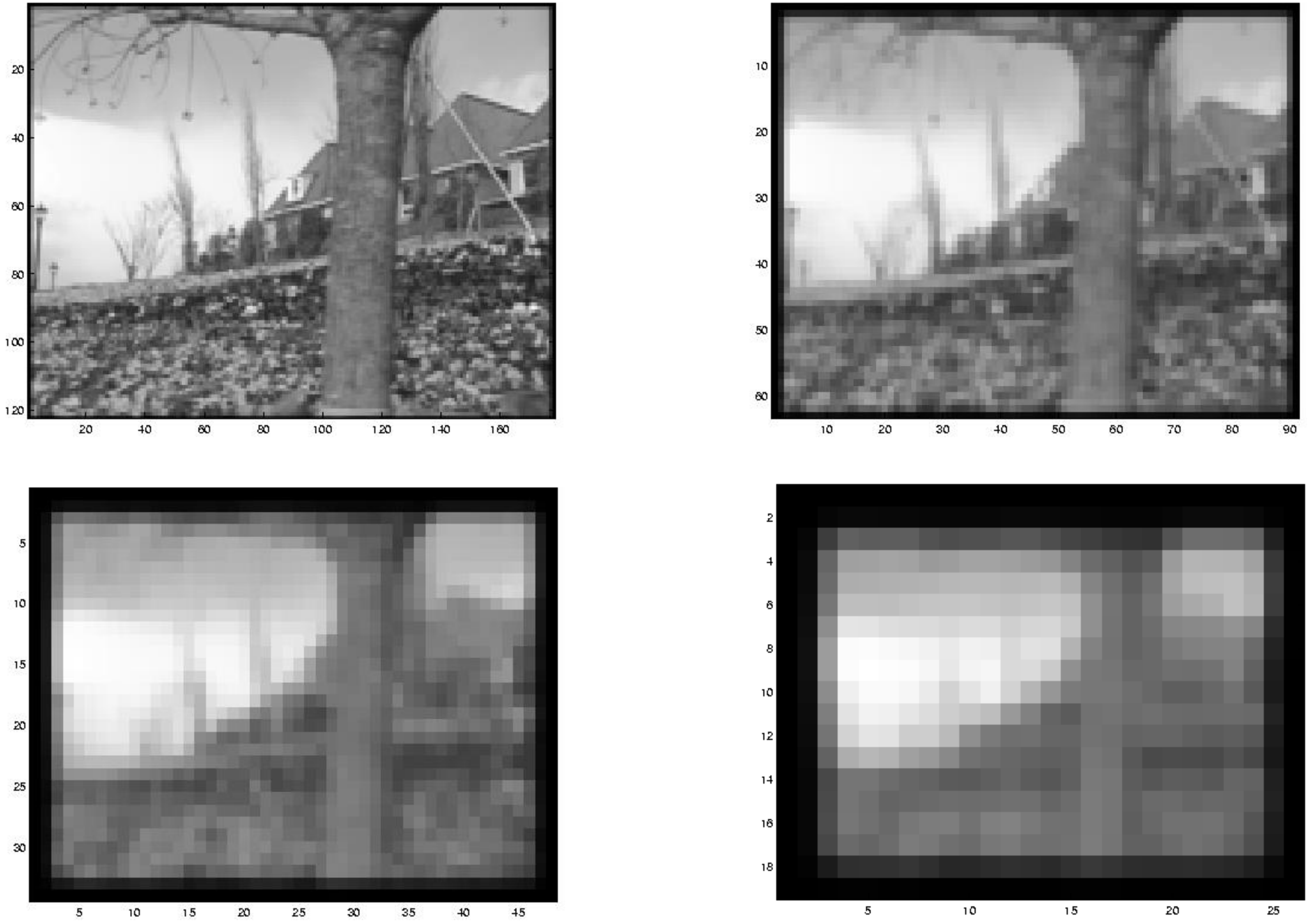
* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Revisiting the small motion assumption

- Is this motion small enough?
  - Probably not—it's much larger than one pixel (2nd order terms dominate)
  - How might we solve this problem?

* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Reduce the resolution!

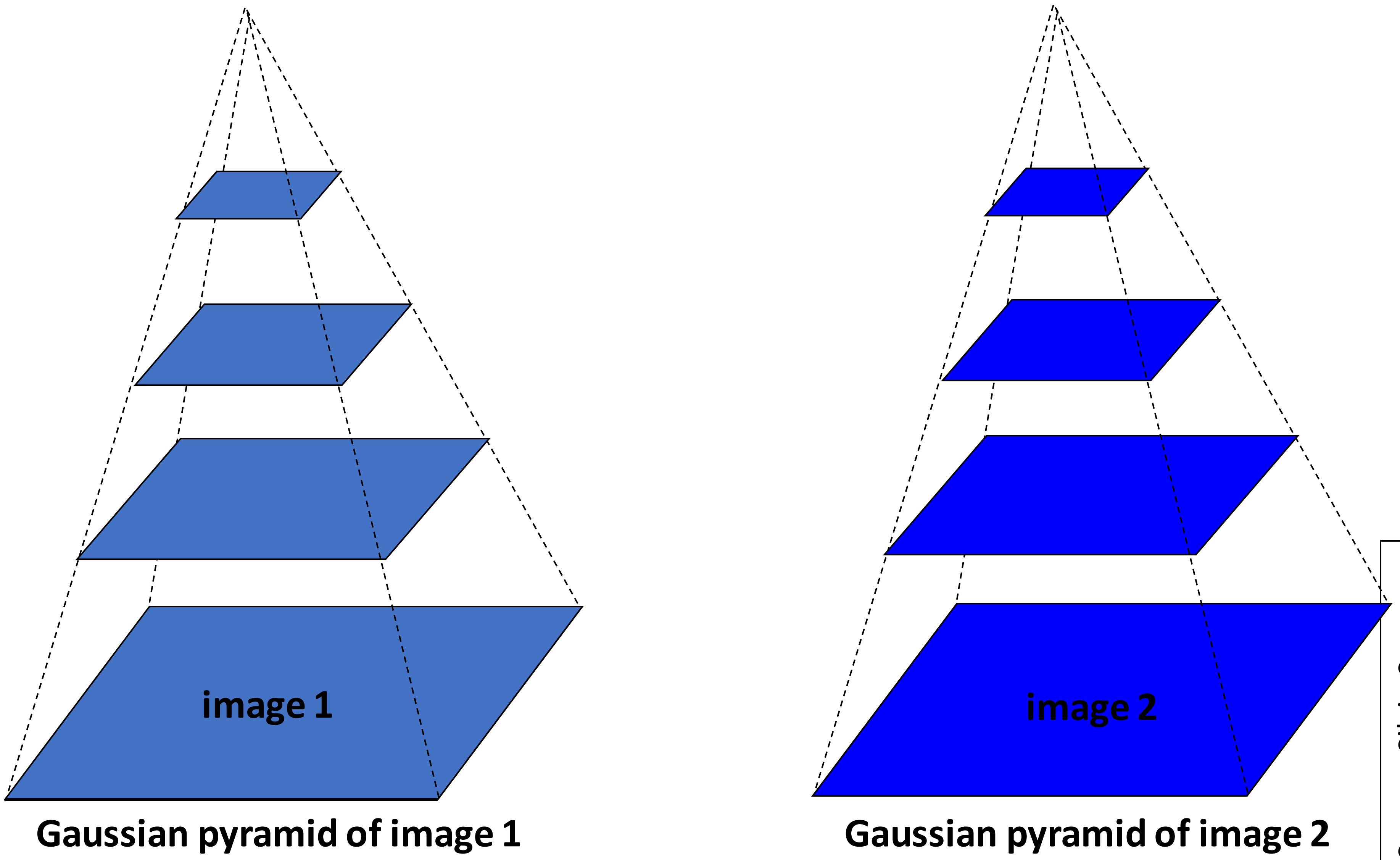


* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Coarse-to-fine optical flow estimation

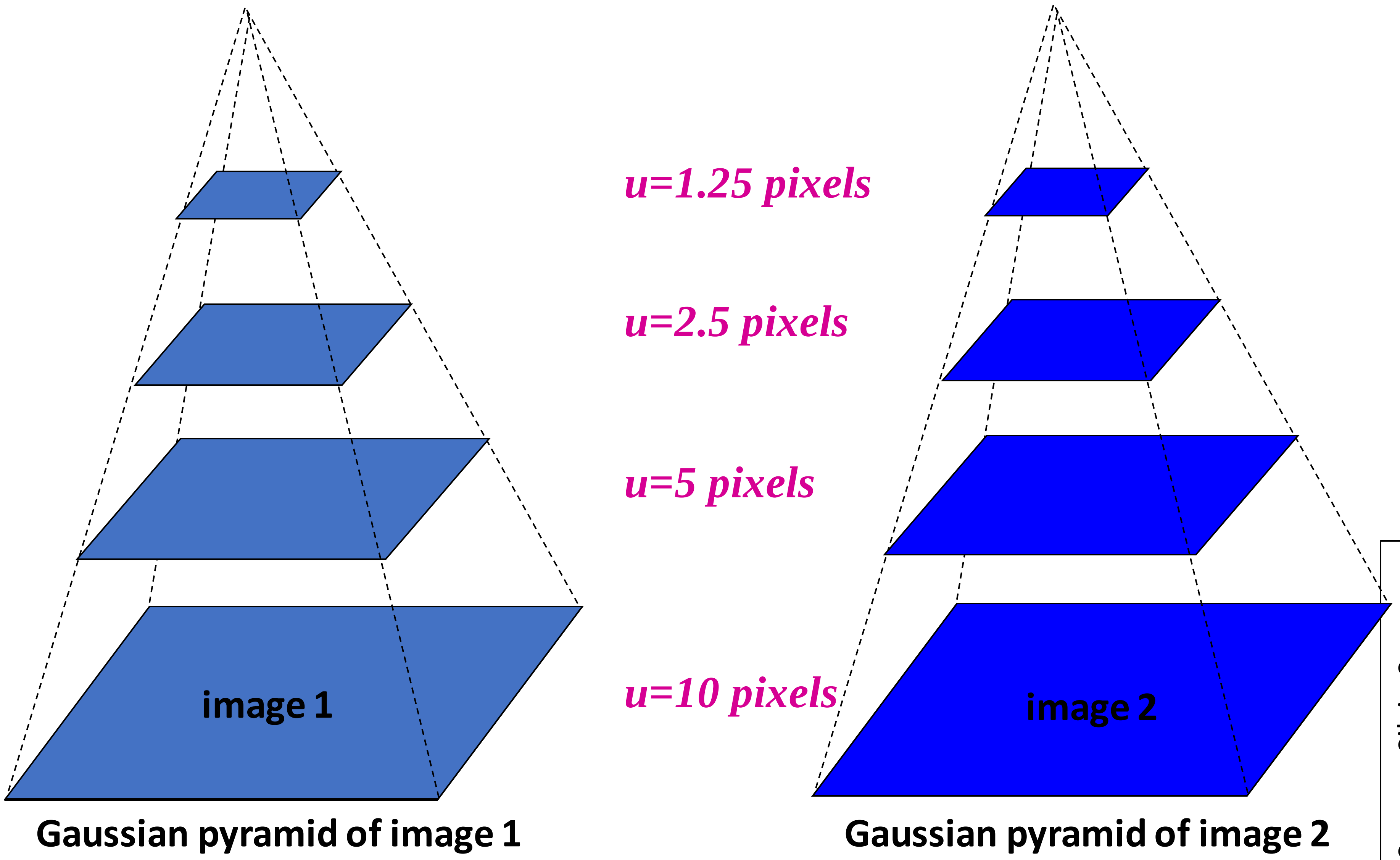
Source: Silvio Savarese

Coarse-to-fine optical flow estimation



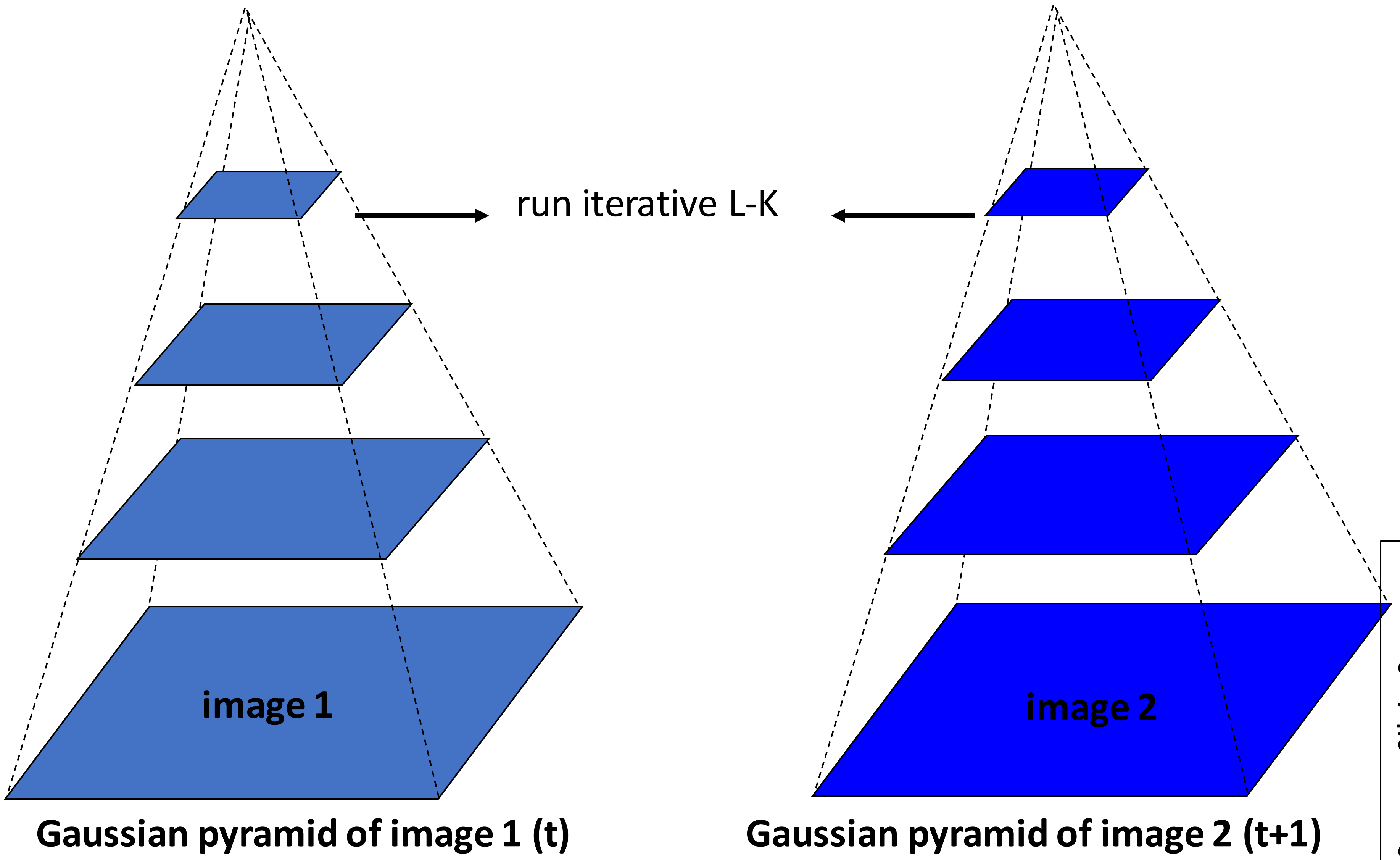
Source: Silvio Savarese

Coarse-to-fine optical flow estimation



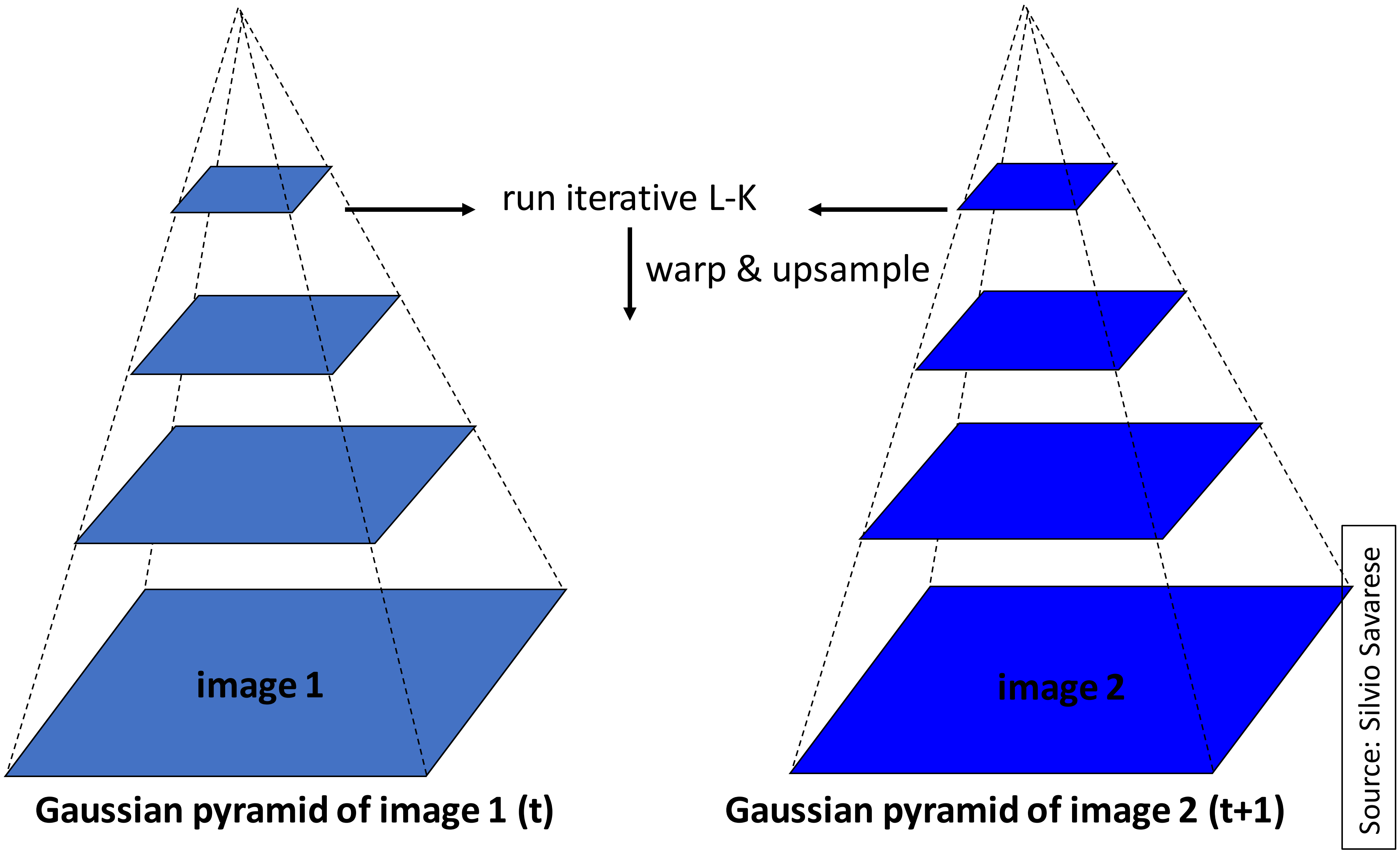
Source: Silvio Savarese

Coarse-to-fine optical flow estimation

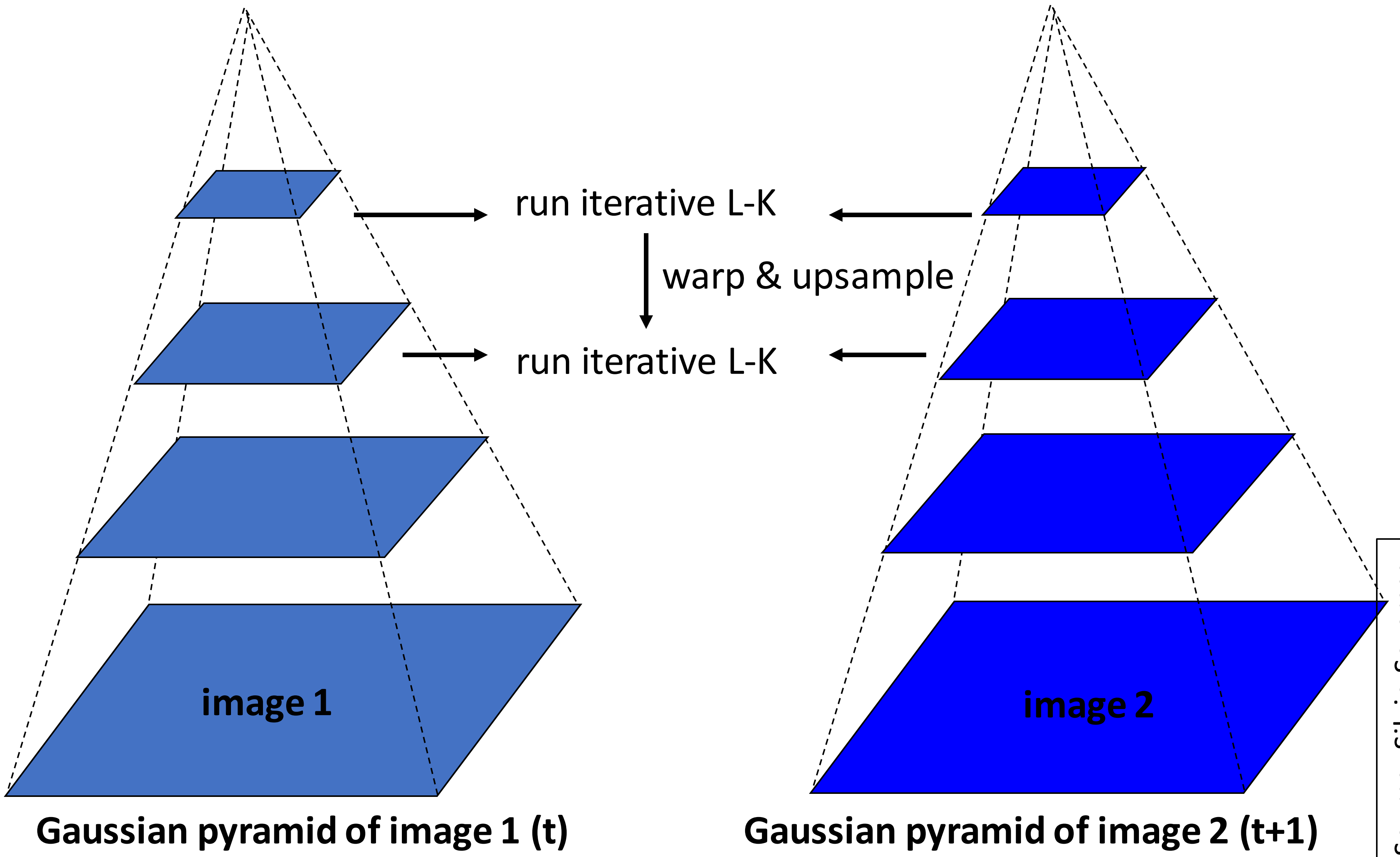


Source: Silvio Savarese

Coarse-to-fine optical flow estimation

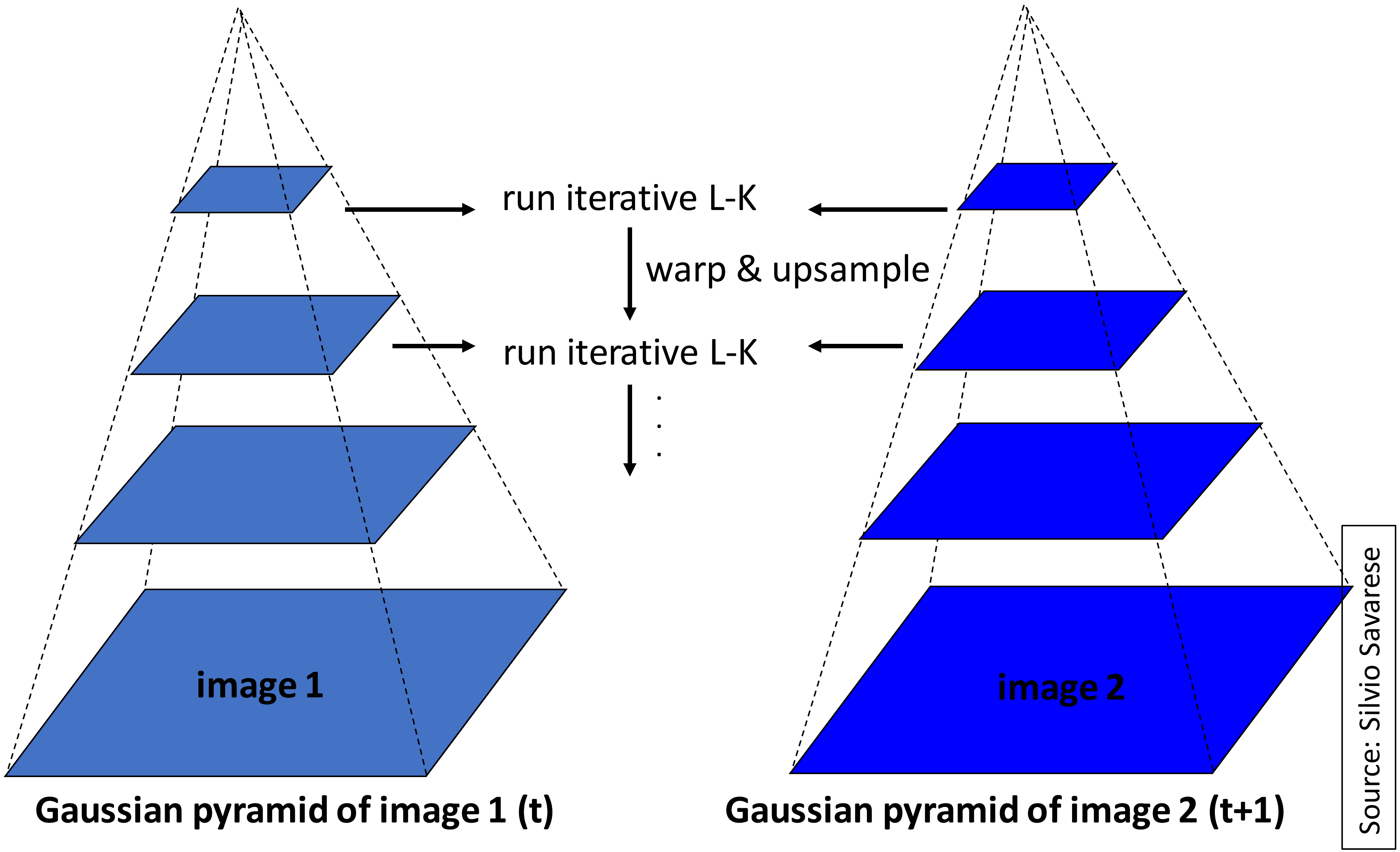


Coarse-to-fine optical flow estimation

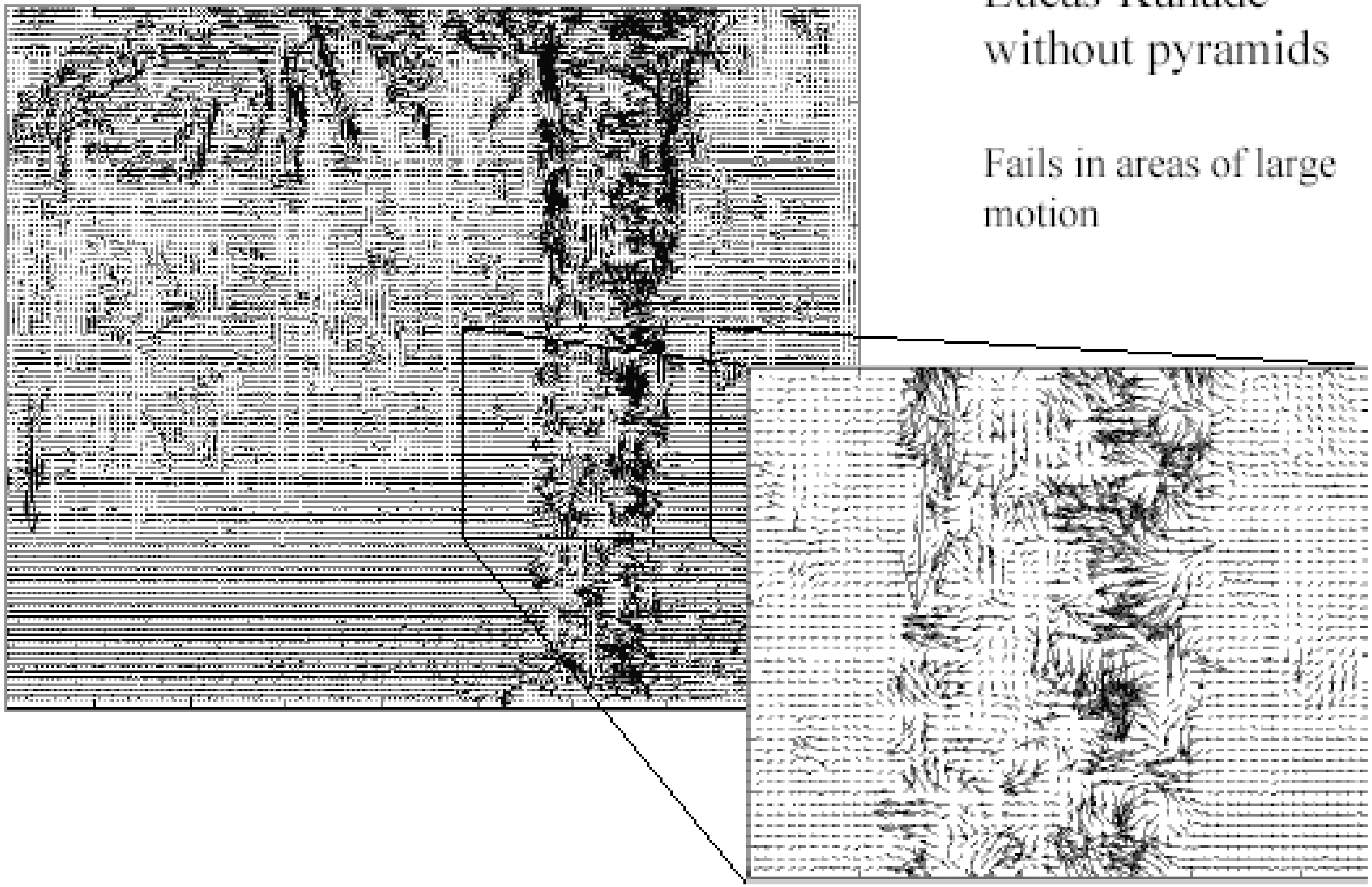


Source: Silvio Savarese

Coarse-to-fine optical flow estimation



Optical Flow Results

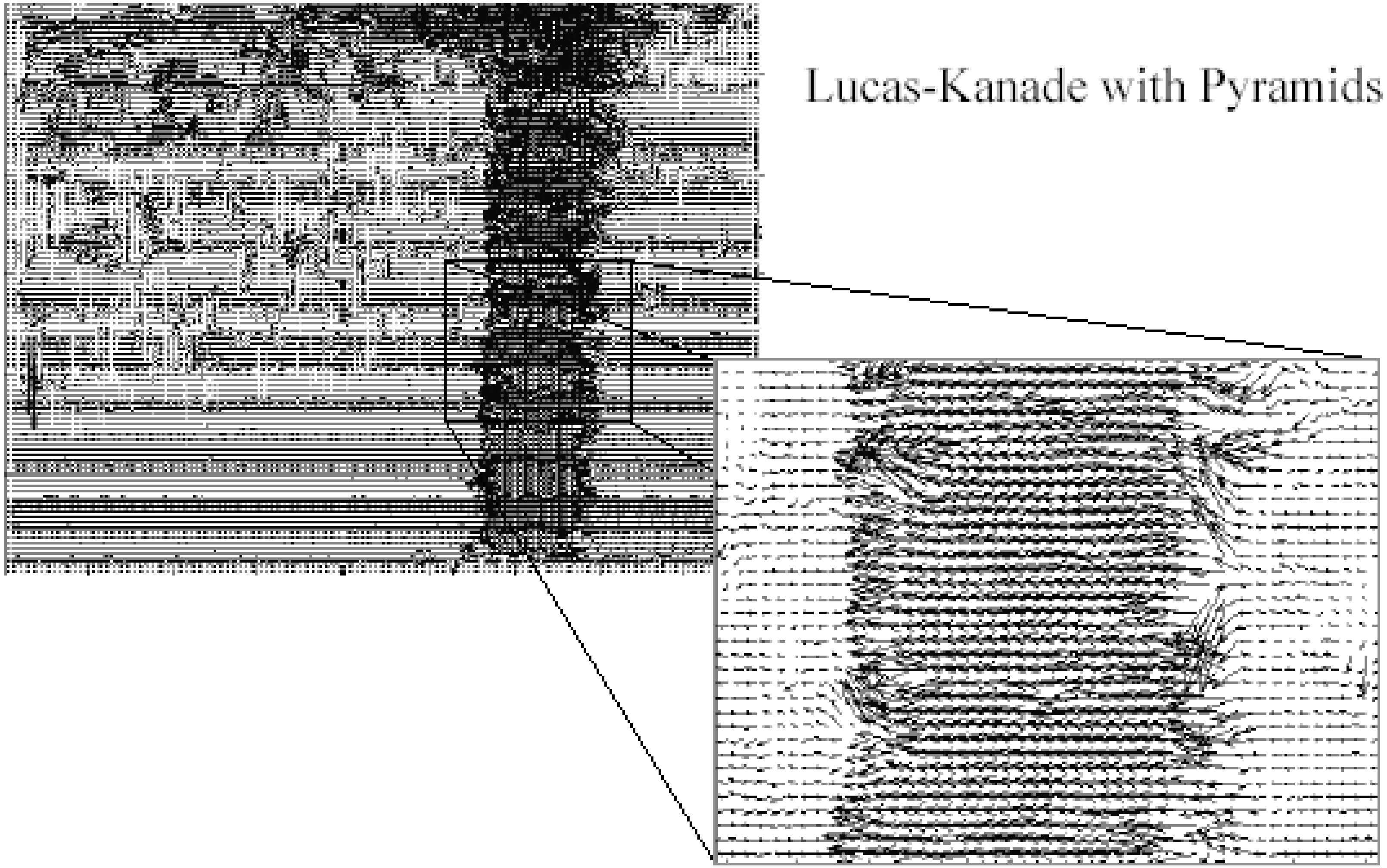


Lucas-Kanade without pyramids

Fails in areas of large motion

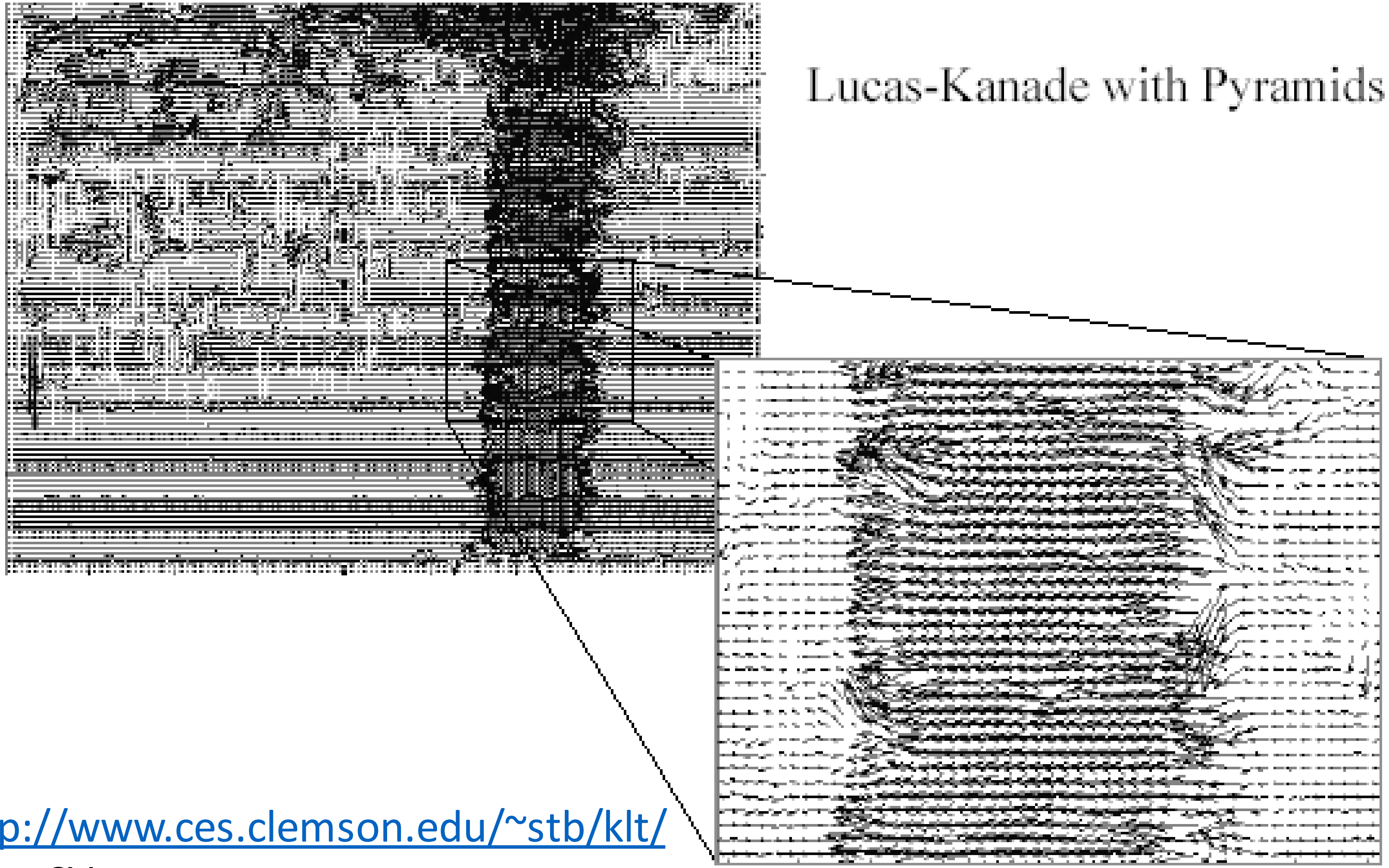
* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Optical Flow Results



* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Optical Flow Results



- http://www.ces.clemson.edu/~stb/klt/
- OpenCV

* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

What we will learn today?

- Optical flow
- Lucas-Kanade method
- Horn-Schunk method
- Pyramids for large motion
- Common fate
- Applications

Recap

- Key assumptions (Errors in Lucas-Kanade)

  - **Small motion:** points do not move very far
  - **Brightness constancy:** projection of the same point looks the same in every frame
  - **Spatial coherence:** points move like their neighbors

Source: Silvio Savarese

Motion segmentation

- How do we represent the motion in this scene?

Source: Silvio Savarese

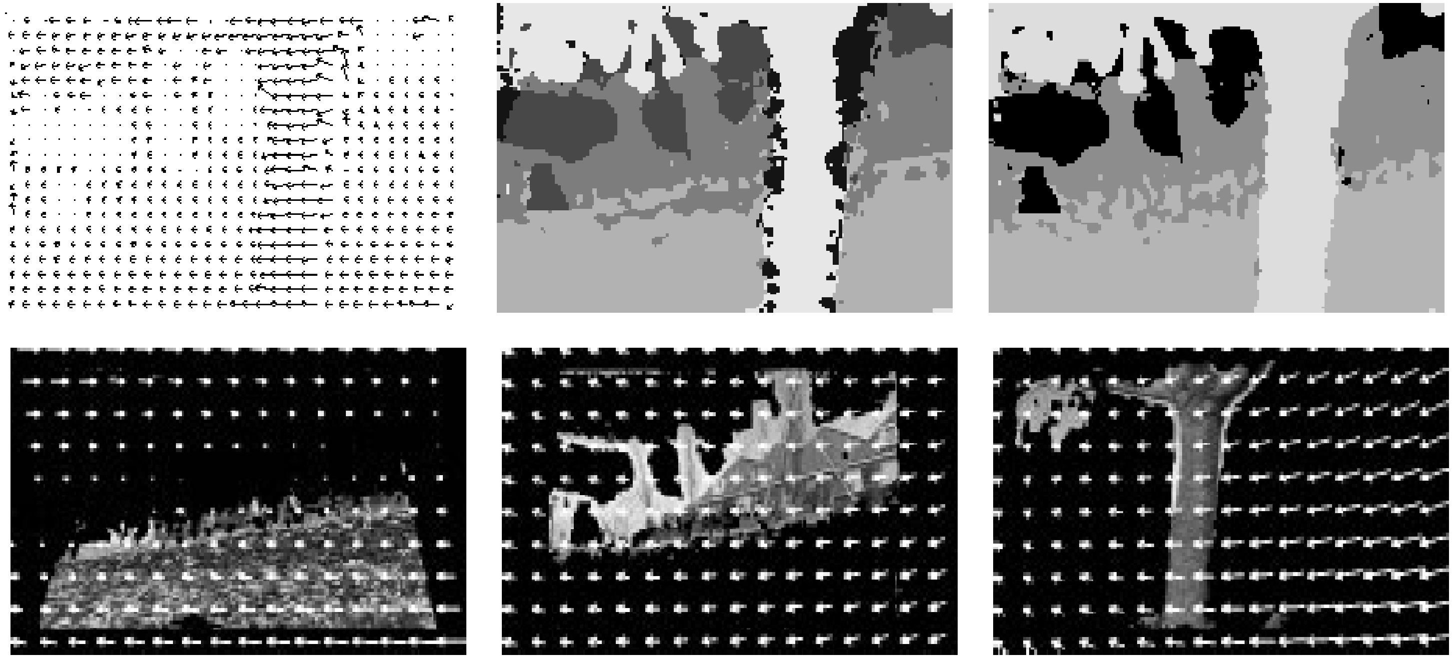
Motion segmentation

J. Wang and E. Adelson. Layered Representation for Motion Analysis. *CVPR 1993*.

- Break image sequence into "layers" each of which has a coherent (affine) motion

Source: Silvio Savarese

Example result



J. Wang and E. Adelson. [Layered Representation for Motion Analysis.](#) *CVPR 1993*.

Source: Silvio Savarese

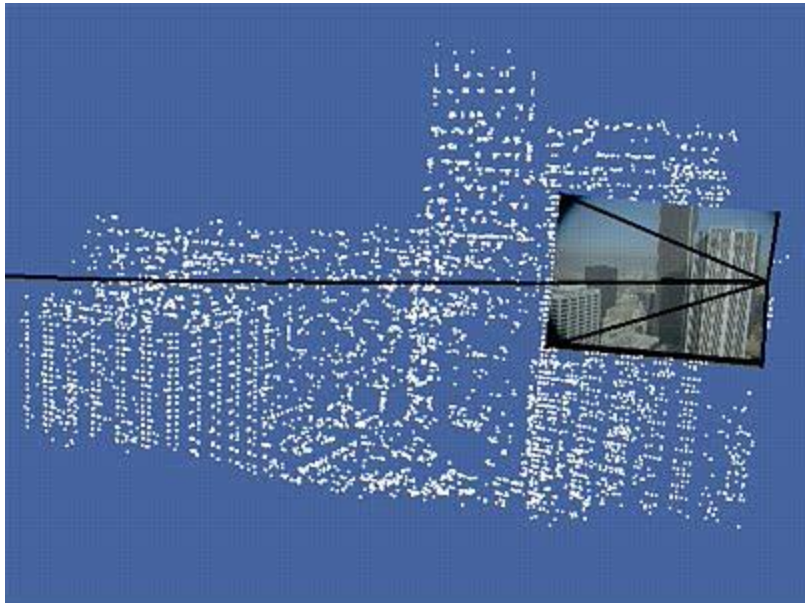
What we will learn today?

- Optical flow
- Lucas-Kanade method
- Horn-Schunk method
- Gunnar Farneback method
- Pyramids for large motion
- Applications

Uses of motion

- Tracking features
- Segmenting objects based on motion cues
- Learning dynamical models
- Improving video quality
  - Motion stabilization
  - Super resolution
- Tracking objects
- Recognizing events and activities

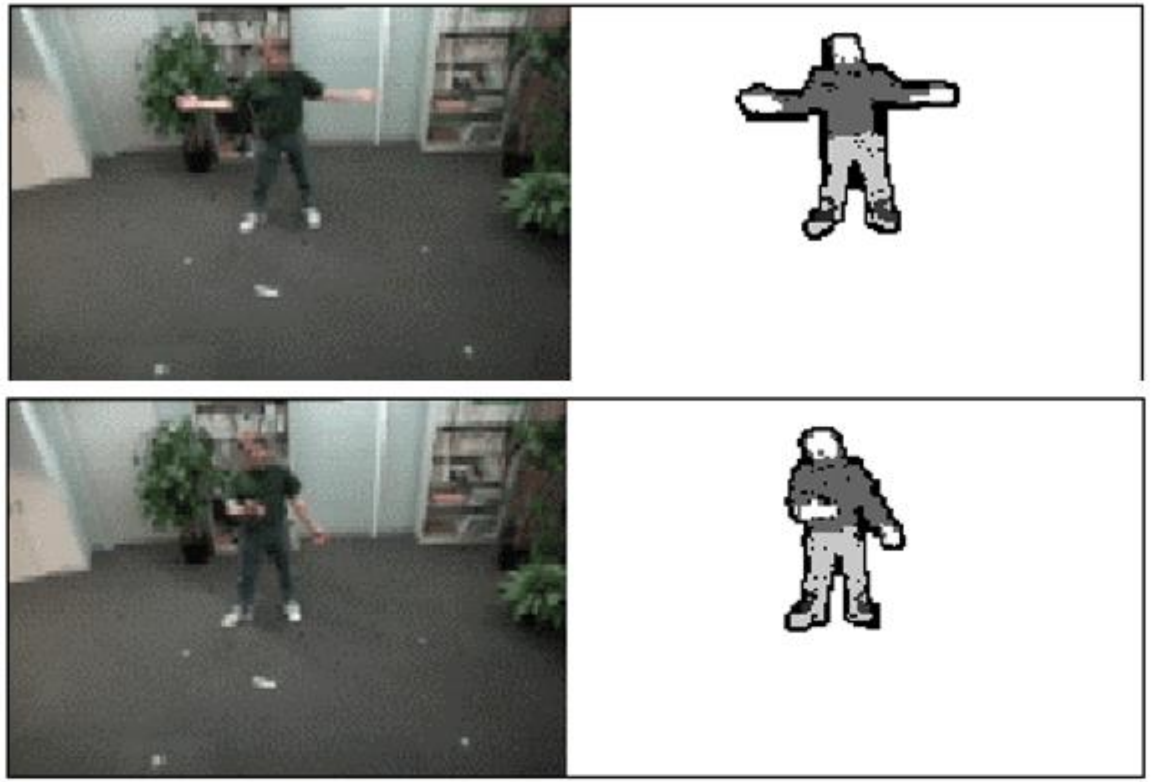
Estimating 3D structure



Source: Silvio Savarese

Segmenting objects based on motion cues

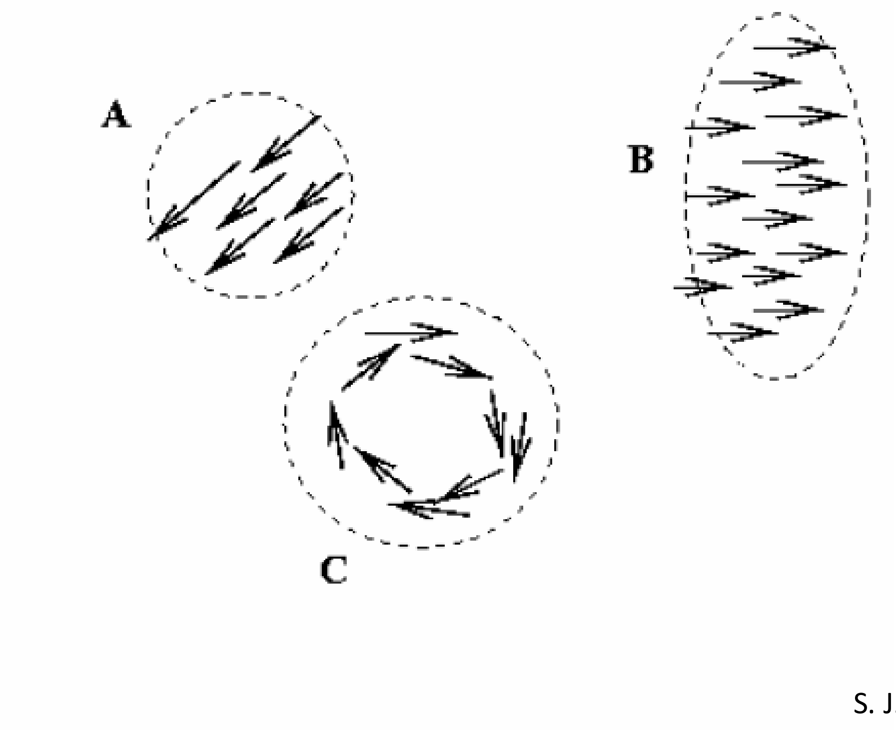
- Background subtraction
  - A static camera is observing a scene
  - Goal: separate the static *background* from the moving *foreground*

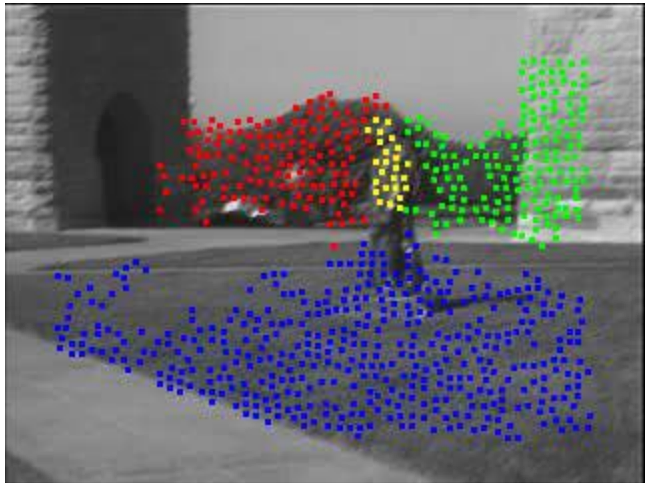


Source: Silvio Savarese

Segmenting objects based on motion cues

- Motion segmentation
  - Segment the video into multiple *coherently* moving objects

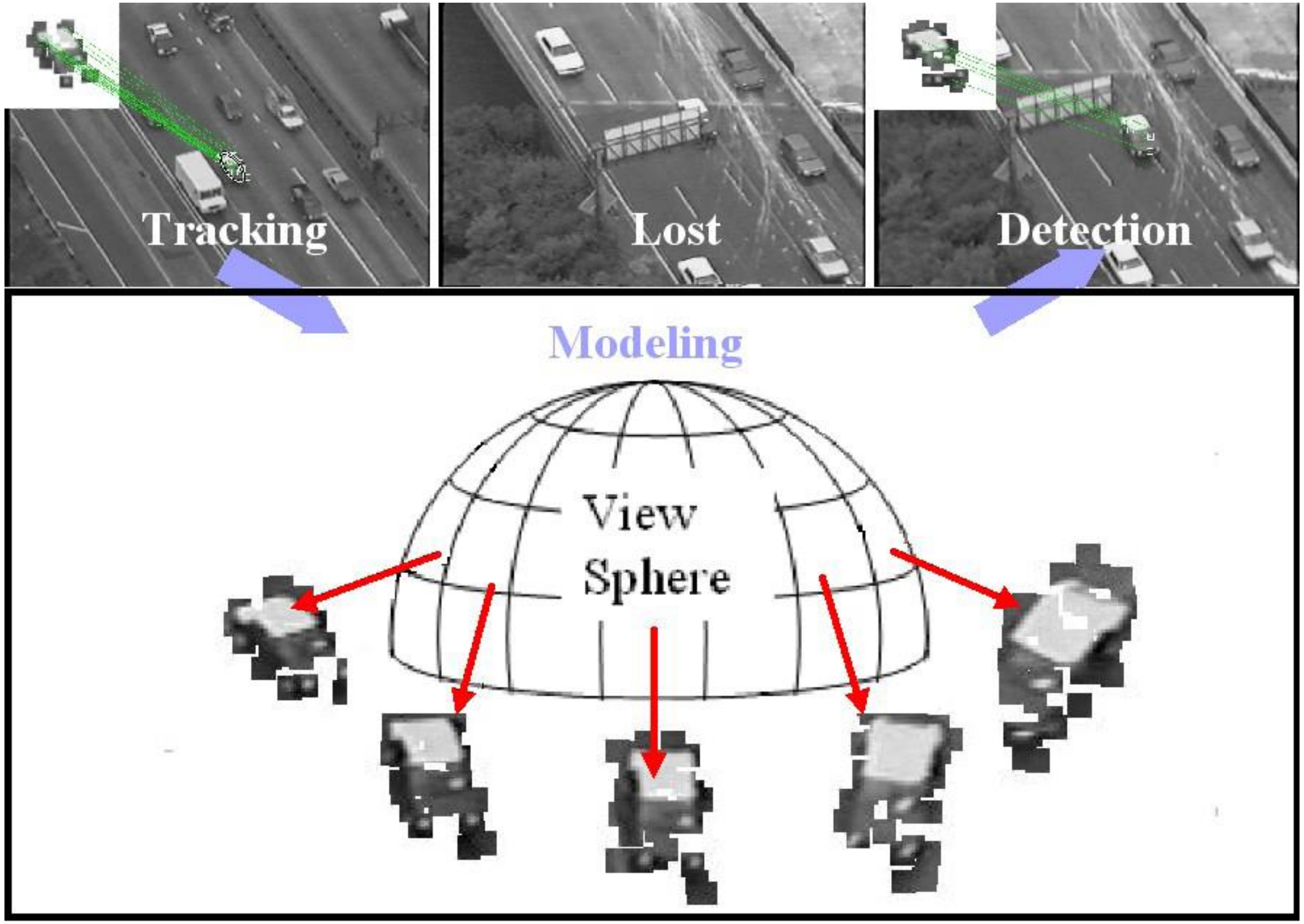




S. J. Pundlik and S. T. Birchfield, Motion Segmentation at Any Speed, Proceedings of the British Machine Vision Conference (BMVC) 2006

Source: Silvio Savarese

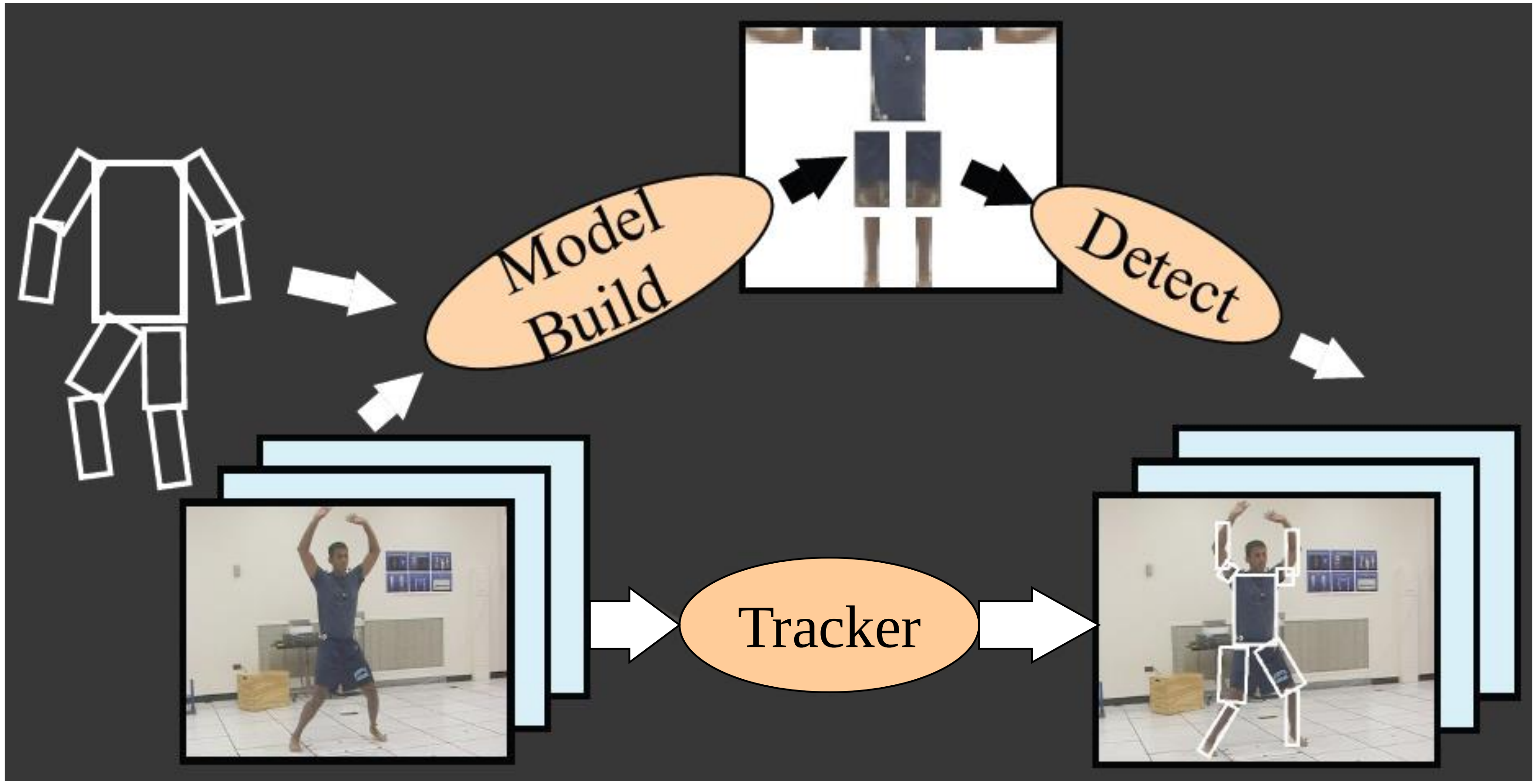
Tracking objects



Z.Yin and R.Collins, "On-the-fly Object Modeling while Tracking," *IEEE Computer Vision and Pattern Recognition (CVPR '07),* Minneapolis, MN, June 2007.

Source: Silvio Savarese

Recognizing events and activities



D. Ramanan, D. Forsyth, and A. Zisserman. Tracking People by Learning their Appearance. PAMI 2007.

Source: Silvio Savarese

When do the optical flow assumptions fail?

In other words, in what situations does the displacement of pixel patches
not represent physical movement of points in space?

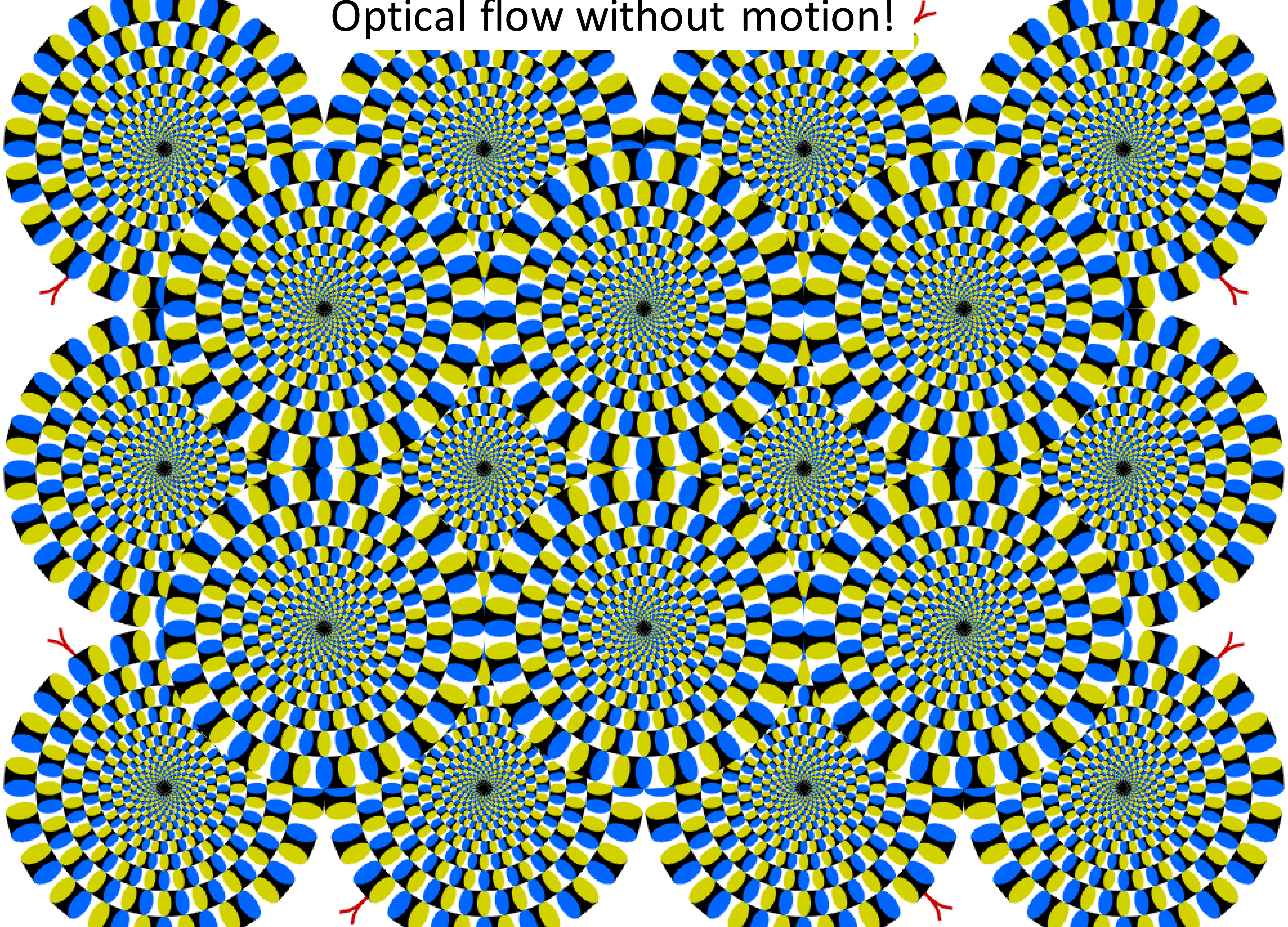
1. Well, TV is based on illusory motion
   – the set is stationary yet things seem to move

2. A uniform rotating sphere
   – nothing seems to move, yet it is rotating

3. Changing directions or intensities of lighting can make things seem to move
   – for example, if the specular highlight on a rotating sphere moves.

4. Muscle movement can make some spots on a cheetah move opposite direction of motion.
   – And infinitely more break downs of optical flow.

Optical flow without motion!



Summary

- Optical flow: apparent motion in a video sequence
- Optical flow are based on following assumptions:
  - Brightness constancy
  - Small motion
  - Spatial coherence
- Optical flow methods
  - Lucas-Kanade: same motion over a patch
  - Horn-Schunk: enforcing small motion with total variation penalty
  - Gunnar-Farneback: model intensity as quadratic function
  - Combine with pyramid to address larger motions
- Applications: motion segmentation, reconstruction, etc.