Optical Flow

Most slides from David Fouhey

http://web.eecs.umich.edu/~fouhey/teaching/EECS442_W22/

https://www.youtube.com/watch?v=G3QrhdfLCO8

Optical Flow

Idea first introduced by psychologist JJ Gibson in ~1940s to describe how to perceive opportunities for motion

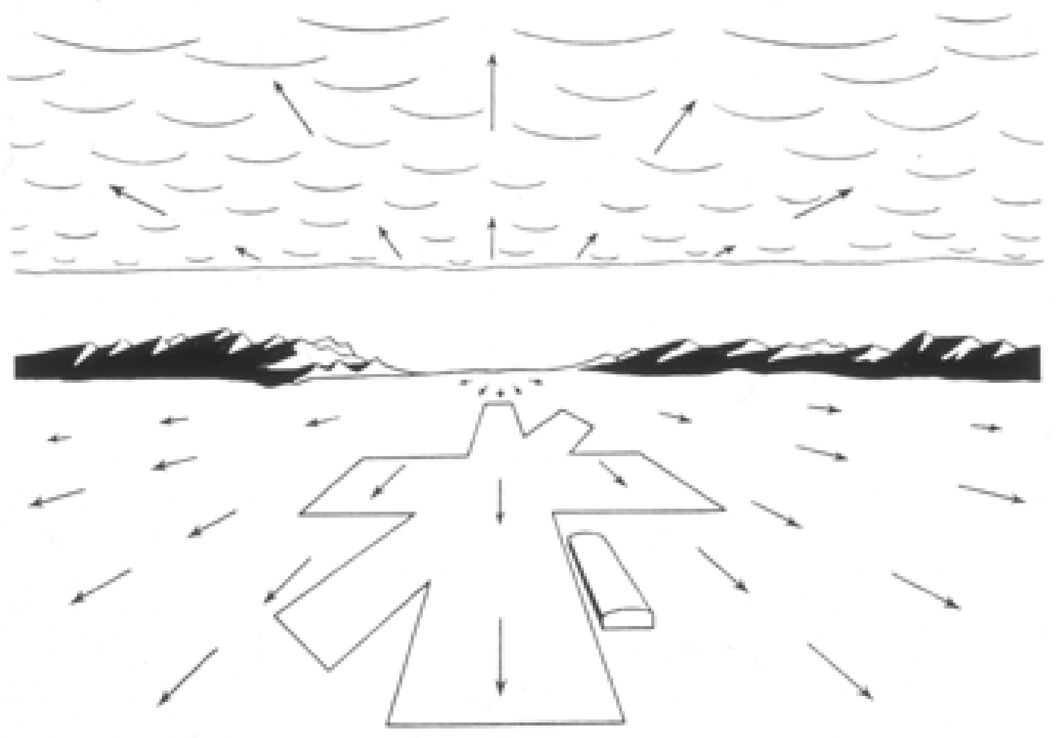
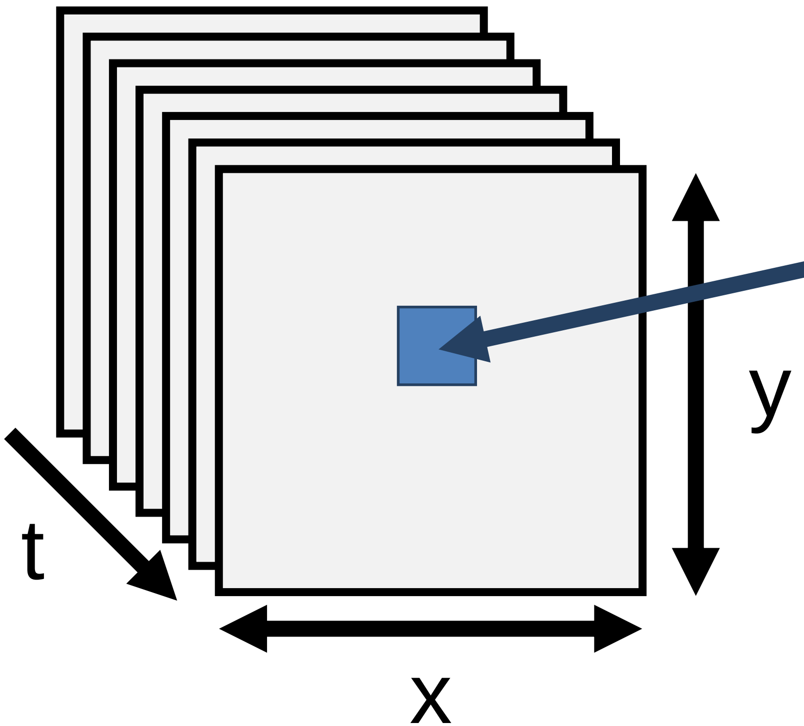


Image Credit: Gibson

Video

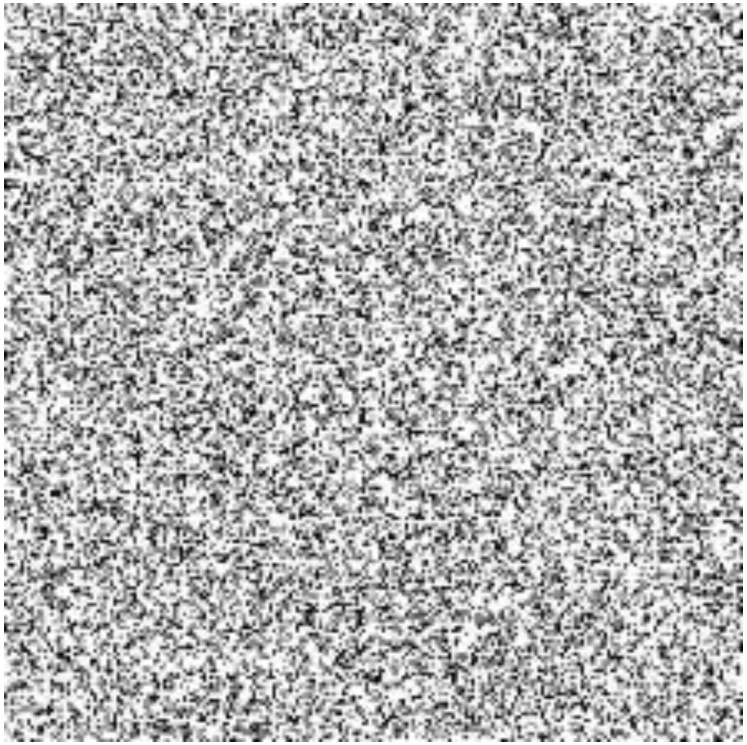
Video: sequence of frames over time
Image is function of space (x,y) and time t
(and channel c)



I(x,y,c,t)
x,y – location
c – channel
t – time

Motion and perceptual organization

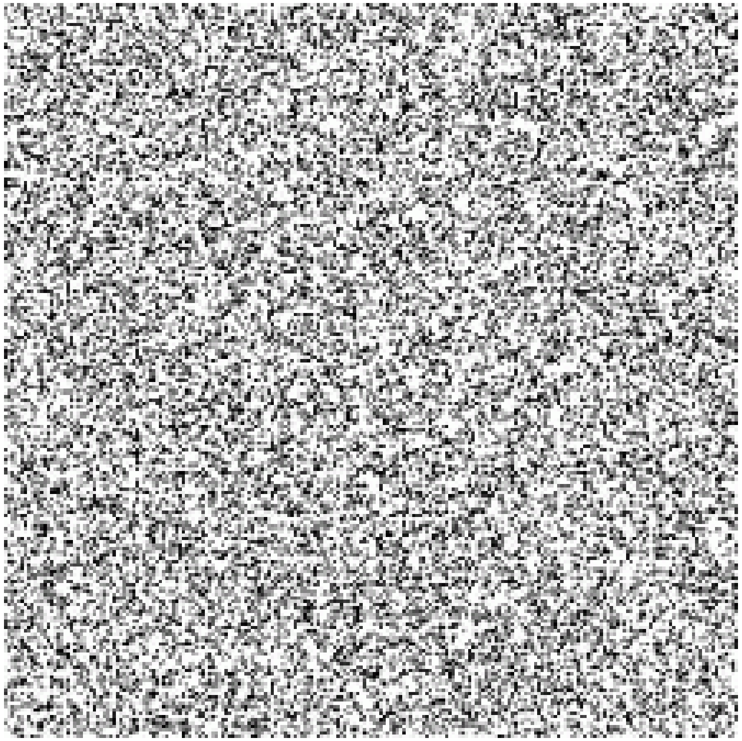
Sometimes motion is the only cue



Slide Credit: S. Lazebnik, but idea of random dot sterogram is due to B. Julesz

Motion and perceptual organization

Sometimes motion is the only cue



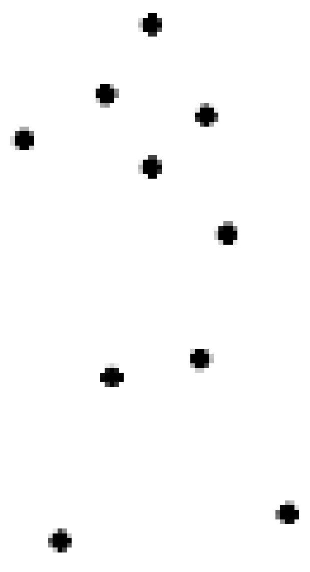
Slide Credit: S. Lazebnik, but idea of random dot sterogram is due to B. Julesz

Motion is a powerful perceptual cue

[Yang et al., “Self-supervised Video Object Segmentation by Motion Grouping”, 2021]

Motion and perceptual organization

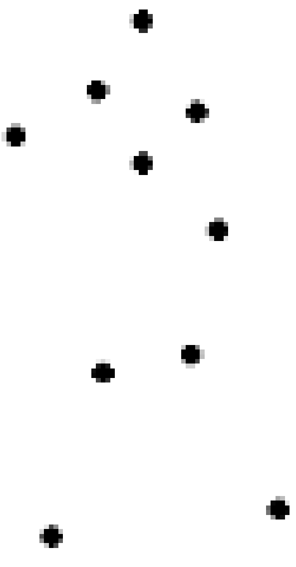
Even impoverished motion data can create a strong percept



Slide Credit: S. Lazebnik

Motion and perceptual organization

Even impoverished motion data can create a strong percept

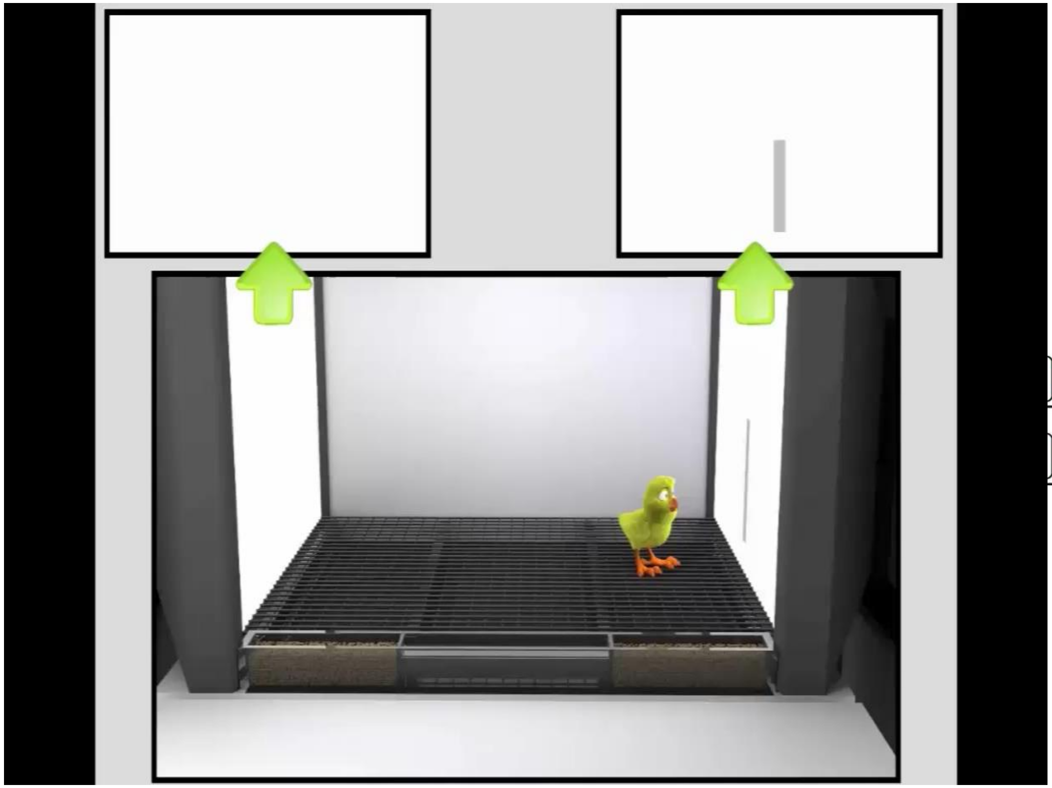


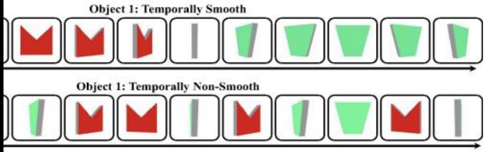
Slide Credit: S. Lazebnik

"It irritated him that the "dog" of 3:14 in the afternoon, seen in profile, should be indicated by the same noun as the dog of 3:15, seen frontally…"

-- from <u>*Funes the Memorious*</u>

Temporal continuity crucial for visual development





Optical flow

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

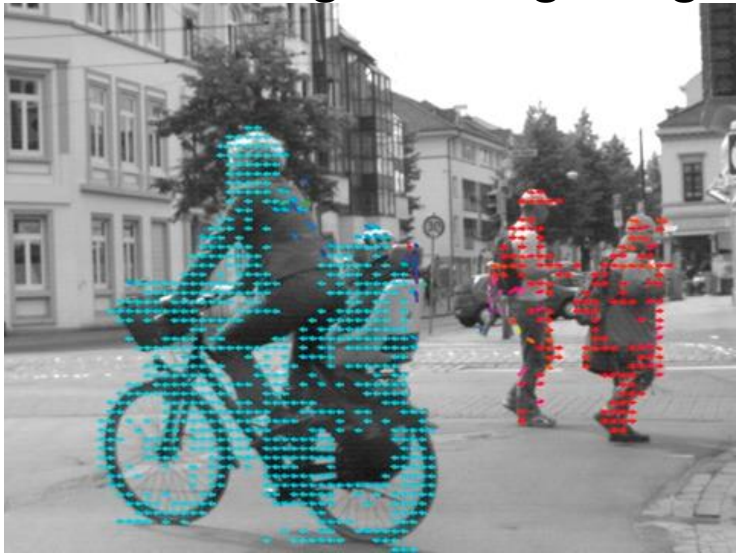


Image source

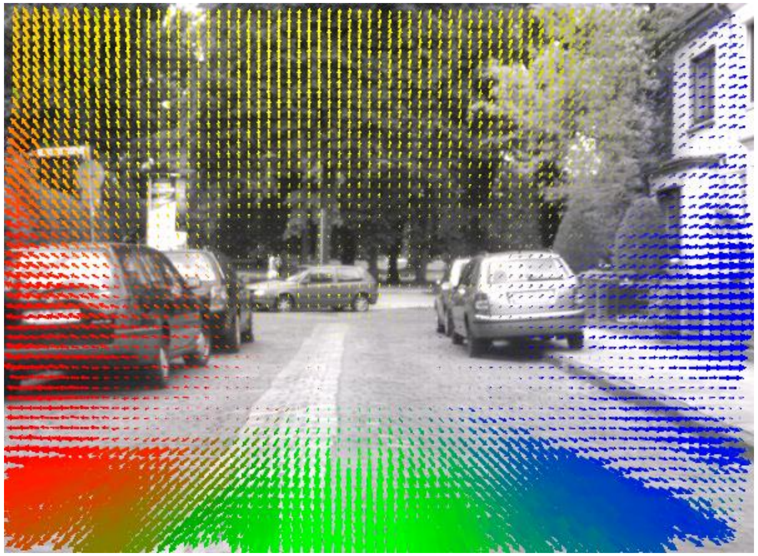
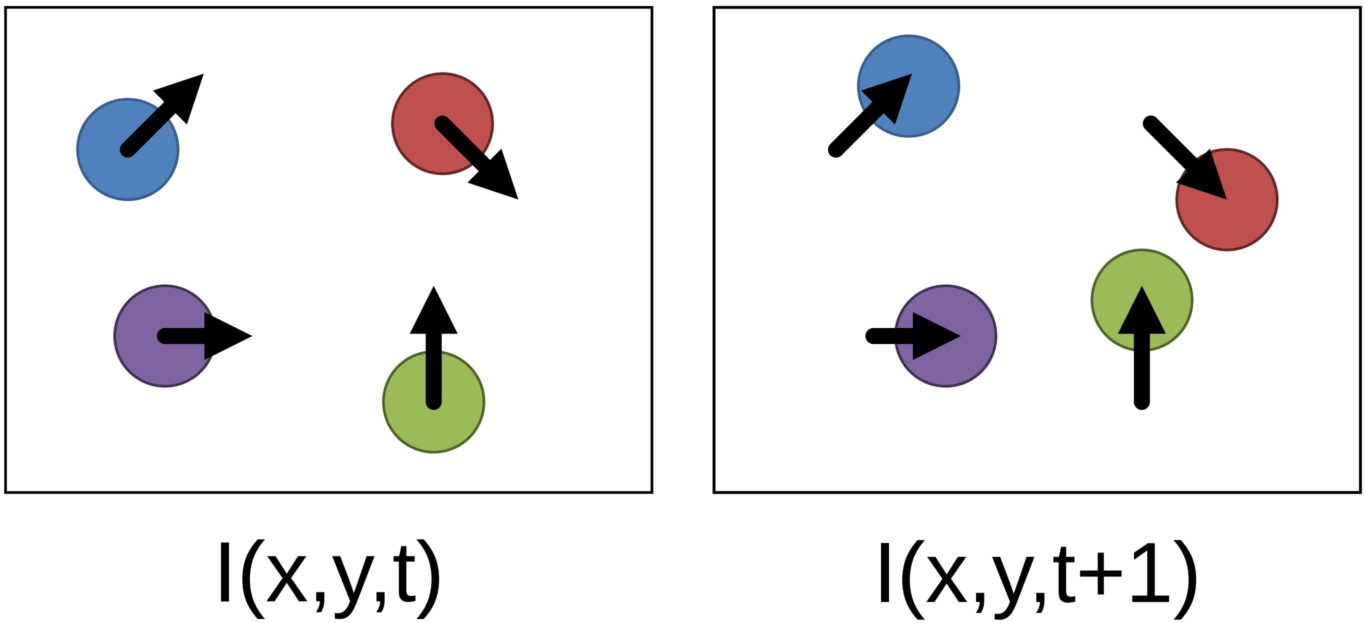


Image source

Source: S. Lazebnik

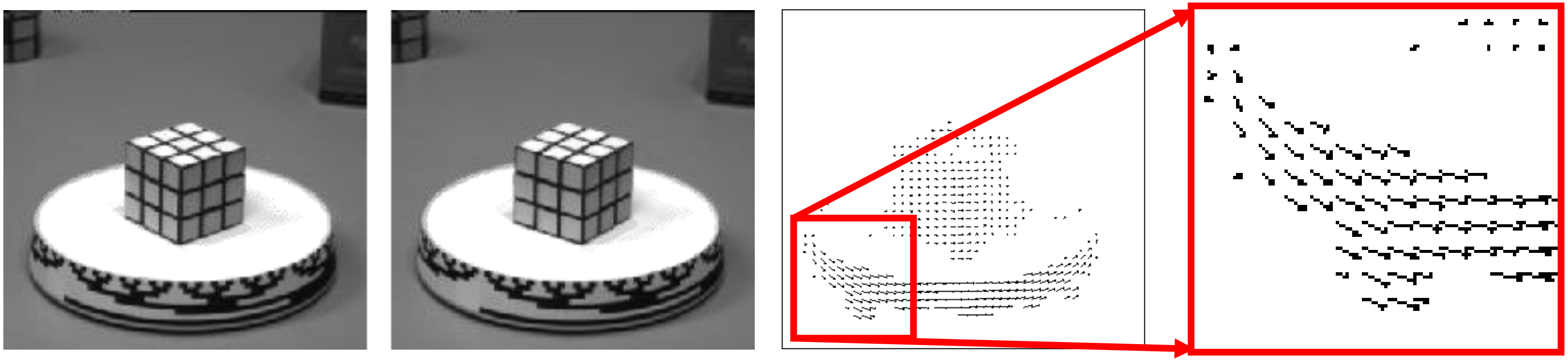
Problem Definition: Optical Flow



Want to estimate pixel motion from image I(x,y,t) to image I(x,y,t+1)

Optical flow

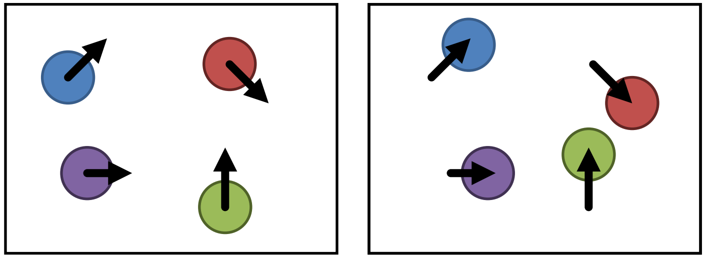
Optical flow is the *apparent* motion of objects



Will start by estimating motion of each pixel separately

Then will consider motion of entire image

Optical Flow



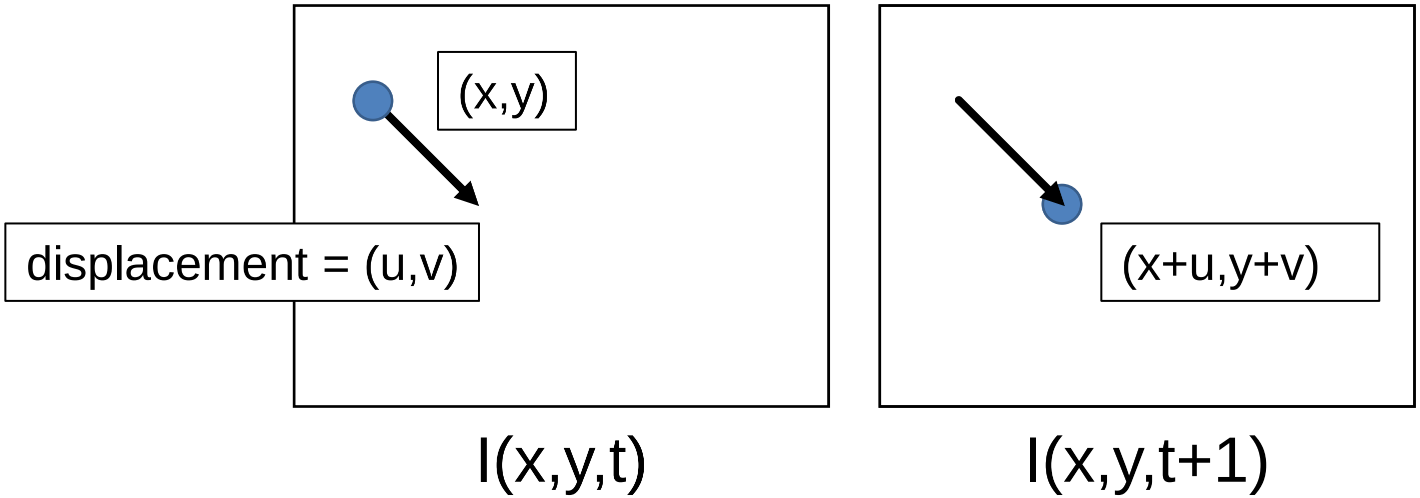
I(x,y,t) I(x,y,t+1)

Solve correspondence problem: given pixel at time t, find **nearby** pixels of the **same color** at time t+1

Key assumptions:

- **Color/brightness constancy**: point at time t looks same at time t+1
- **Small motion**: points do not move very far

Optical Flow



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

Wrong way to do things: brute force match

Aside: Taylor Series for Images

Recall Taylor Series – way of *linearizing* a function:

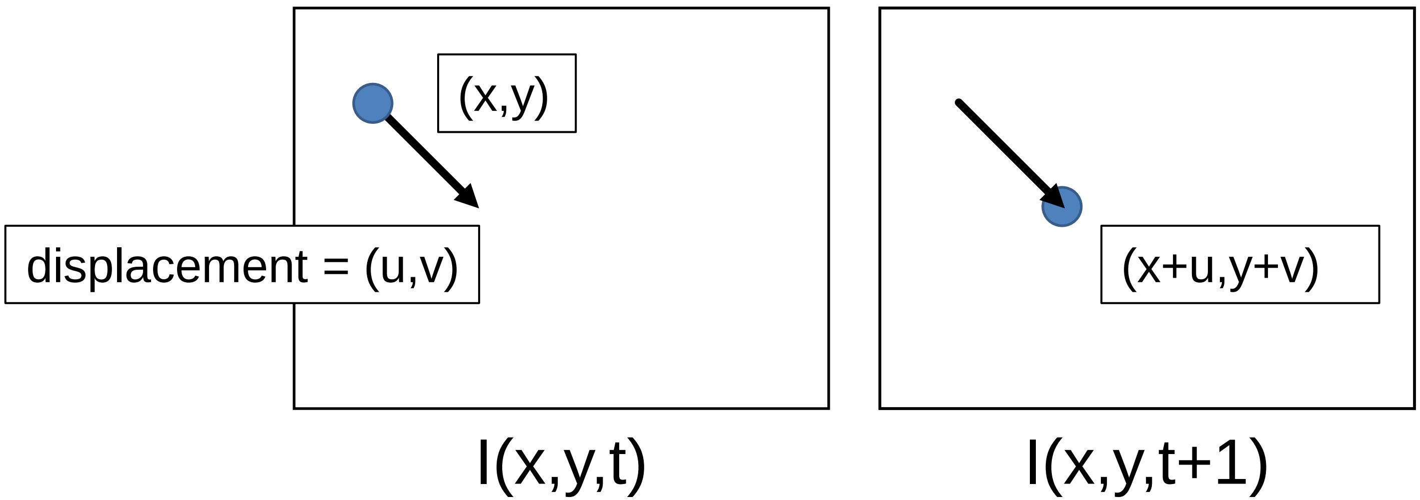
$$f(x+d) \approx f(x) + \frac{\partial f}{\partial x} d$$

Do the same with images, treating them as function of x, y

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

For brevity: Ix = Ix at point (x,y), Iy = Iy at point (x,y)

Optical Flow



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

Taylor Expansion: $I(x + u, y + v, t) = I(x, y, t) + I_x u + I_y v + \cdots$

Optical Flow Equation

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

$$\begin{aligned} 0 &\approx I(x+u, y+v, t+1) - I(x, y, t) \\ &= I(x, y, t+1) + I_x u + I_y v - I(x, y, t) \\ &= \underbrace{I(x, y, t+1) - I(x, y, t)} + I_x u + I_y v \end{aligned}$$

Taylor Expansion

If you had to guess, what would you call this?

Optical Flow Equation

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

$$\begin{aligned}
0 &\approx I(x+u, y+v, t+1) - I(x, y, t) \\
&= I(x, y, t+1) + I_x u + I_y v - I(x, y, t) \\
&= I(x, y, t+1) - I(x, y, t) + I_x u + I_y v \\
&= I_t + I_x u + I_y v \\
&= I_t + \nabla I \cdot [u, v]
\end{aligned}$$

Taylor Expansion

When is this approximation exact?

[u,v] = [0,0]

When is it bad?

u or v big.

Adapted from S. Lazebnik slides

Optical Flow Equation

Brightness constancy equation

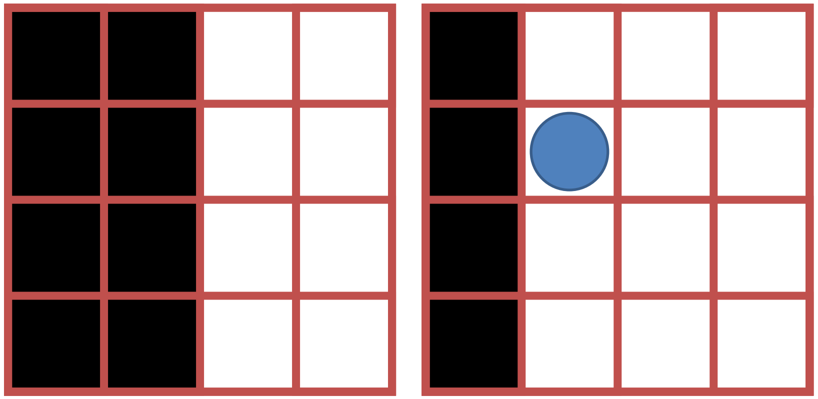
$$I_x u + I_y v + I_t = 0$$

What do static image gradients have to do with motion estimation?

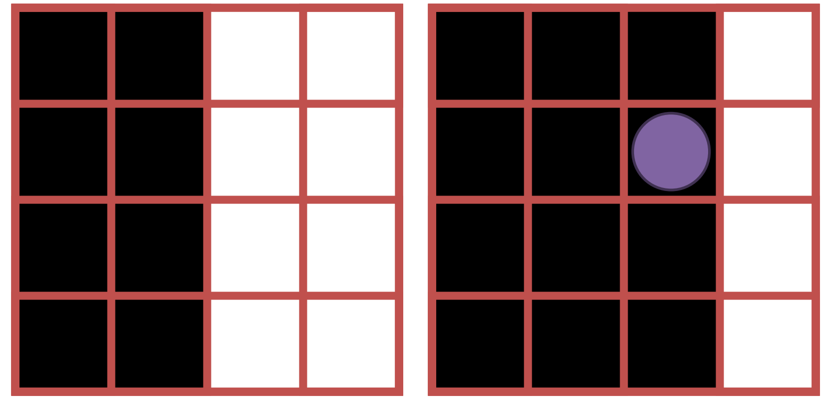
Slide Credit: S. Lazebnik

Brightness Constancy Example

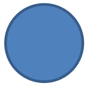
$$I_x u + I_y v + I_t = 0$$



t t+1

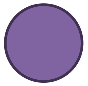


t t+1

@ 

It = 1-0 = 1
Iy = 0
Ix = 1-0 = 1

What's u?

@ 

It = 0-1 = -1
Iy = 0
Ix = 1-0 = 1

What's u?

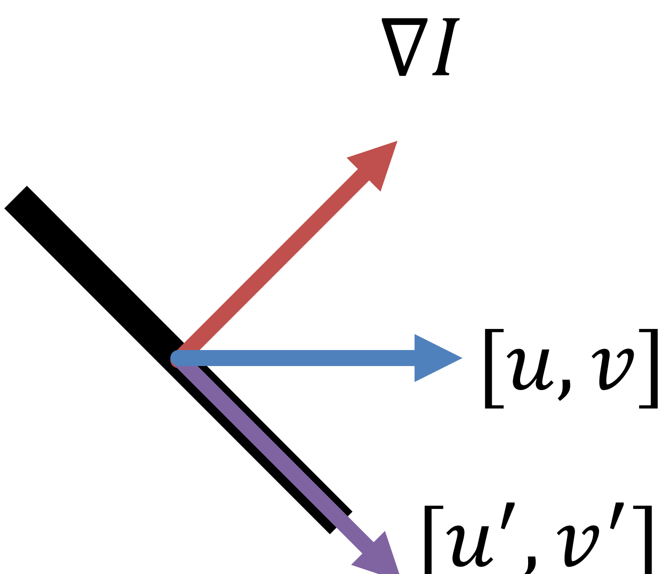
Optical Flow Equation

Have: $I_x u + I_y v + I_t = 0$ $\quad I_t + \nabla I \cdot [u, v] = 0$

How many equations and unknowns per pixel?

1 (single equation), 2 (u and v)

One nasty problem:



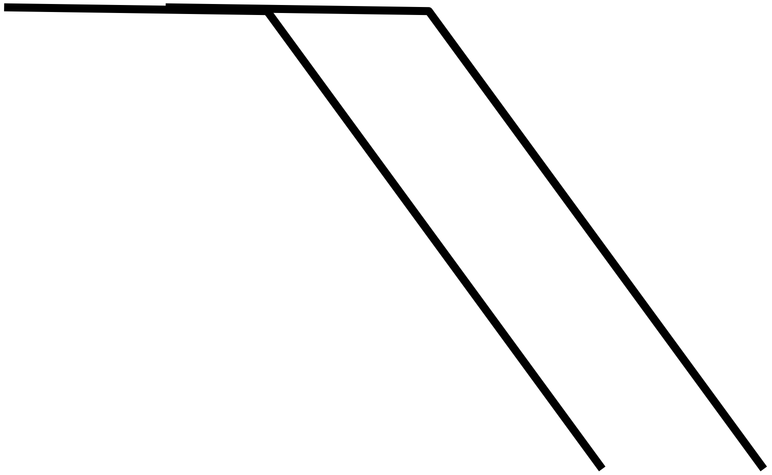
Suppose $\nabla I^T [u', v'] = 0$

$\mathrm{I_t} + \nabla I^T [u + u', v + v'] = 0$

Can only identify the motion along gradient and **not** motion perpendicular to it

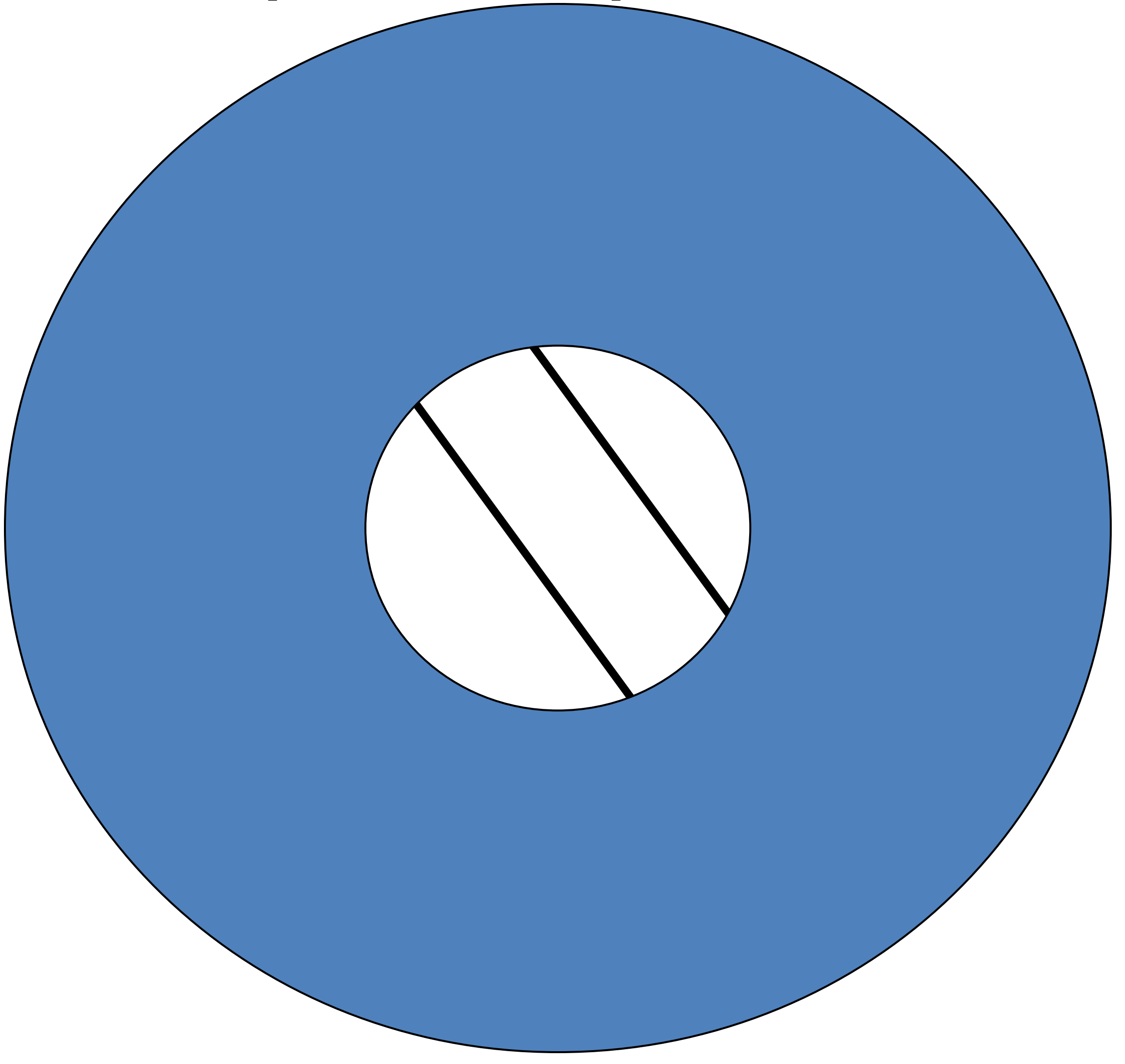
Adapted from S. Lazebnik slides

Aperture problem



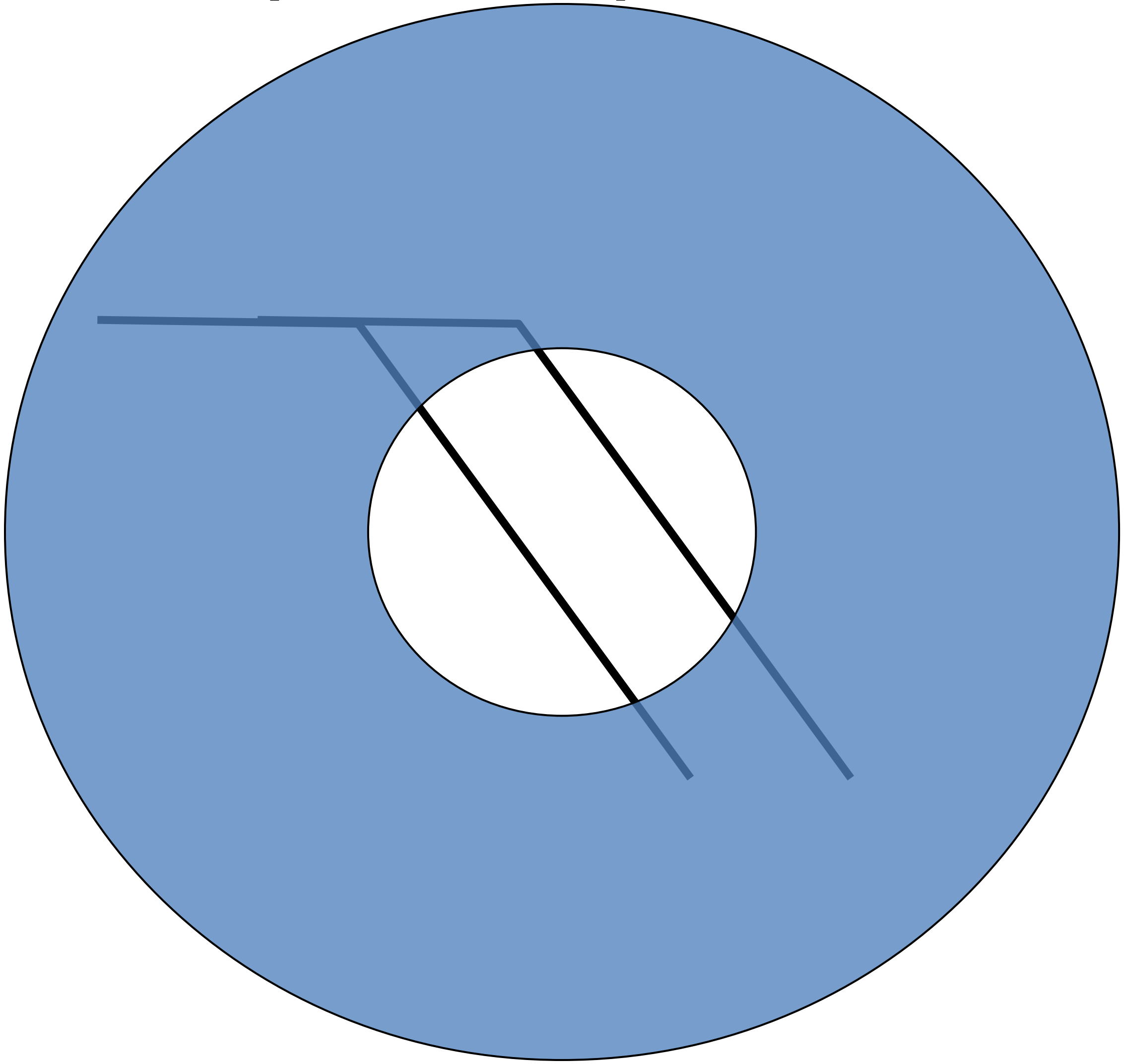
Slide credit: S. Lazebnik

Aperture problem



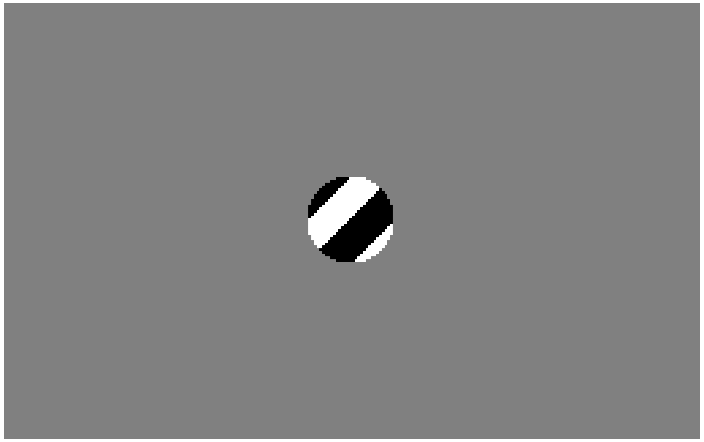
Slide credit: S. Lazebnik

Aperture problem



Slide credit: S. Lazebnik

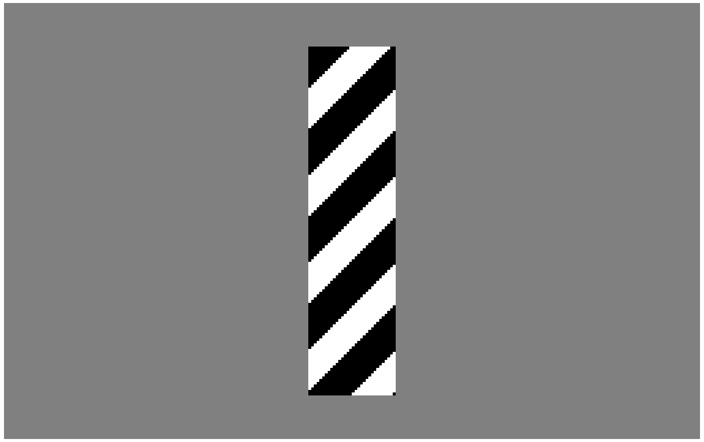
The barber pole illusion



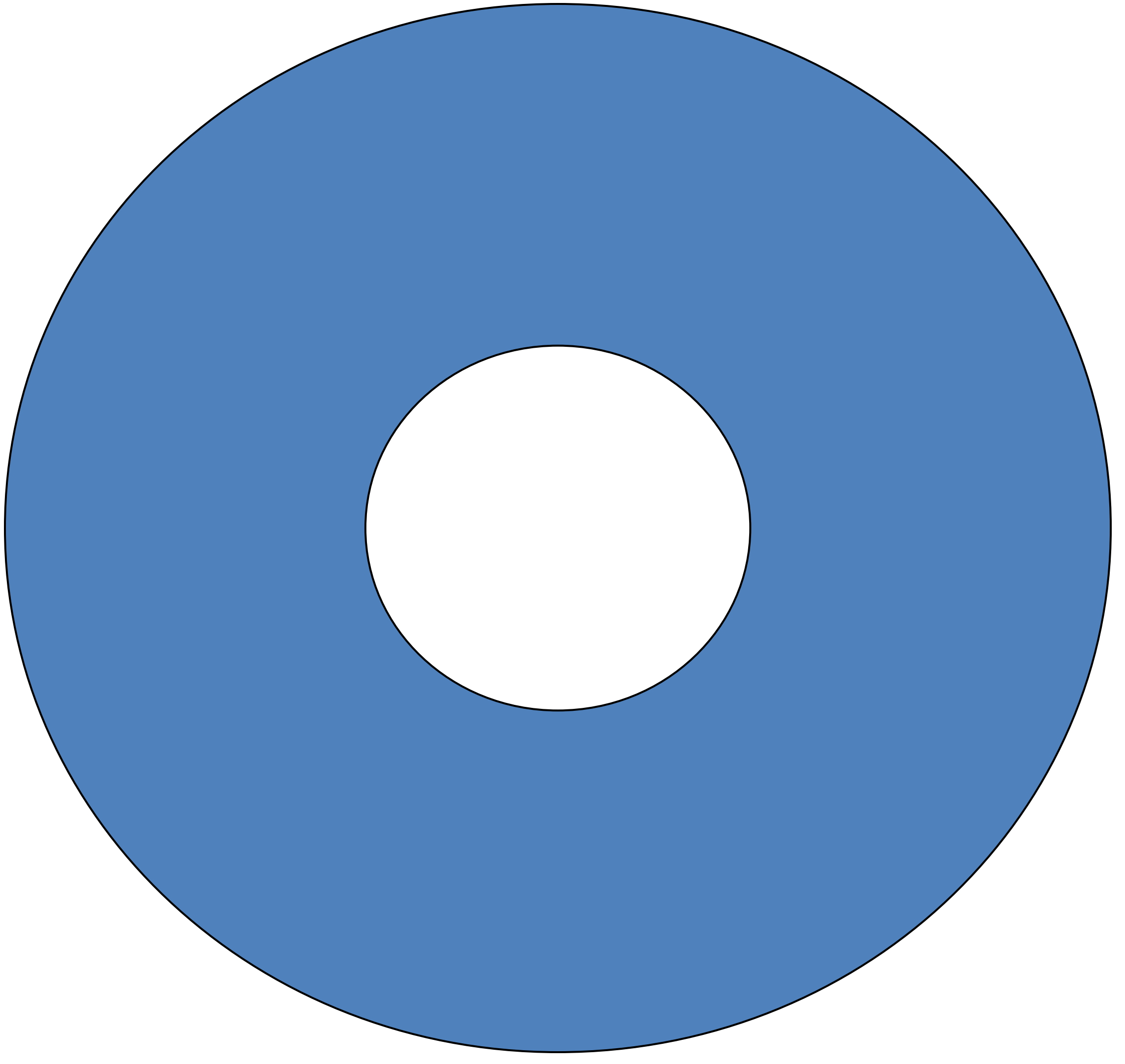
http://en.wikipedia.org/wiki/Barberpole_illusion

Source: S. Lazebnik

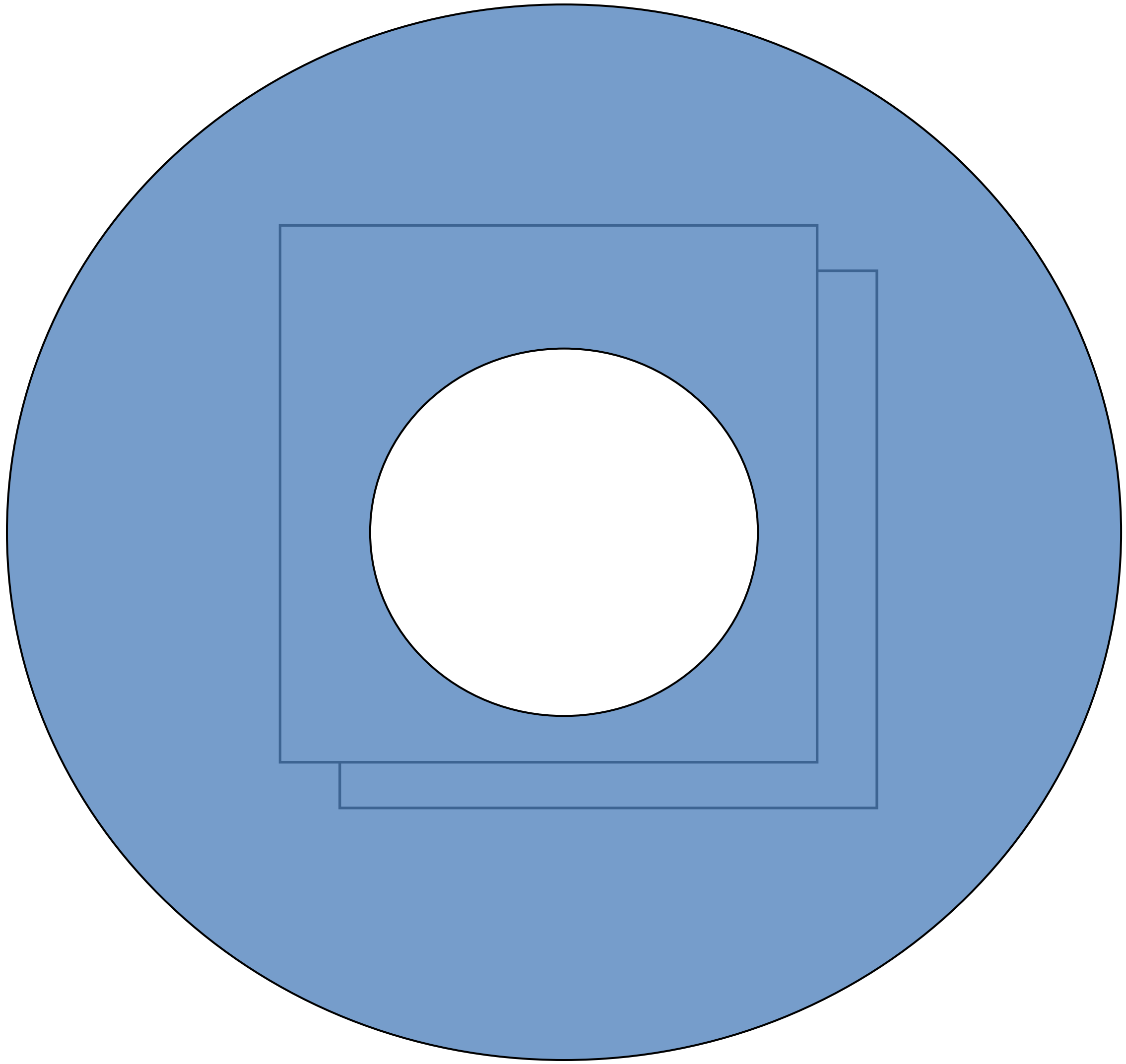
The barber pole illusion



Other Invisible Flow



Other Invisible Flow



Solving Ambiguity – Lucas Kanade

2 unknowns [u,v], 1 eqn per pixel
How do we get more equations?
Assume *spatial coherence*: pixel's neighbors have *move together* / have same [u,v]
5x5 window gives 25 new equations

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

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

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 for [u,v]

$$\begin{bmatrix} I_x(p_1) & I_y(p_1) \\ \vdots & \vdots \\ I_x(p_{25}) & I_y(p_{25}) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(p_1) \\ \vdots \\ I_t(p_{25}) \end{bmatrix} \qquad \underset{25x2}{\boldsymbol{A}}\ \underset{2x1}{\boldsymbol{d}} = \underset{25x1}{\boldsymbol{b}}$$

What's the solution?

$$(\boldsymbol{A}^T\boldsymbol{A})\boldsymbol{d} = \boldsymbol{A}^T\boldsymbol{b} \quad \rightarrow \quad \boldsymbol{d} = (\boldsymbol{A}^T\boldsymbol{A})^{-1}\boldsymbol{A}^T\boldsymbol{b}$$

Intuitively, need to solve (sum over pixels in window)

$$\underbrace{\begin{bmatrix} \sum \boldsymbol{I}_x\boldsymbol{I}_x & \sum \boldsymbol{I}_x\boldsymbol{I}_y \\ \sum \boldsymbol{I}_x\boldsymbol{I}_y & \sum \boldsymbol{I}_y\boldsymbol{I}_y \end{bmatrix}}_{\boldsymbol{A}^T\boldsymbol{A}} \begin{bmatrix} u \\ v \end{bmatrix} = - \underbrace{\begin{bmatrix} \sum \boldsymbol{I}_x\boldsymbol{I}_t \\ \sum \boldsymbol{I}_y\boldsymbol{I}_t \end{bmatrix}}_{\boldsymbol{A}^T\boldsymbol{b}}$$

Adapted from S. Lazebnik slides

Solving for [u,v]

$$\begin{bmatrix} \sum \boldsymbol{I}_x \boldsymbol{I}_x & \sum \boldsymbol{I}_x \boldsymbol{I}_y \\ \sum \boldsymbol{I}_x \boldsymbol{I}_y & \sum \boldsymbol{I}_y \boldsymbol{I}_y \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} \sum \boldsymbol{I}_x \boldsymbol{I}_t \\ \sum \boldsymbol{I}_y \boldsymbol{I}_t \end{bmatrix}$$

$$\boldsymbol{A}^T \boldsymbol{A} \qquad\qquad\qquad \boldsymbol{A}^T \boldsymbol{b}$$

$\boldsymbol{A}^T \boldsymbol{A}$ is called Second-Moment Matrix

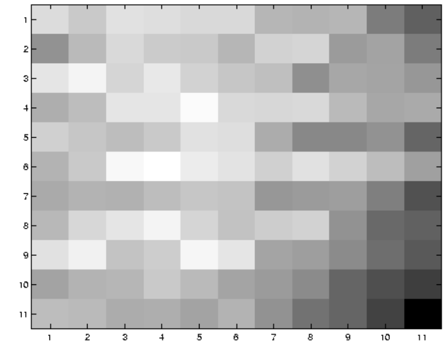
When can we find [u,v]?

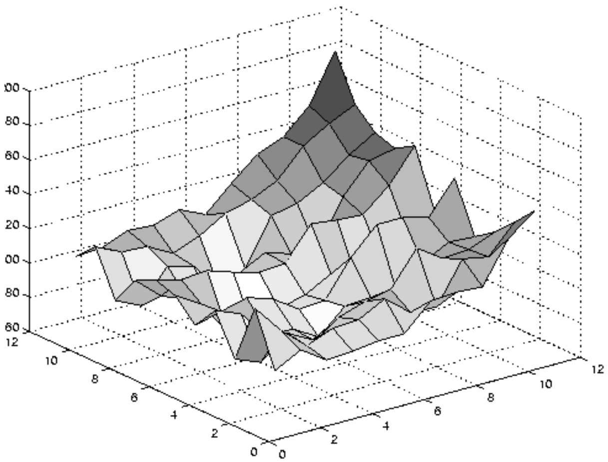
$\mathbf{A}^T\mathbf{A}$ invertible: can’t have constant image

$\mathbf{A}^T\mathbf{A}$ well-conditioned: $|\lambda_1|/ |\lambda_2|$ not large (edge)

Adapted from S. Lazebnik slides

Low texture region





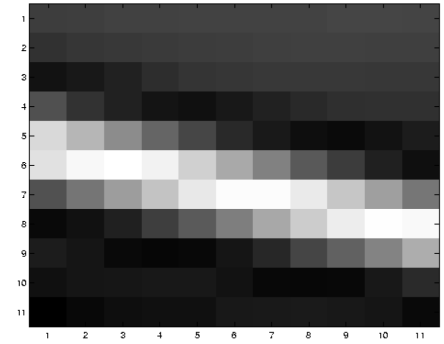
$$\begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix} = \sum \nabla \mathrm{I} (\nabla \mathrm{I})^{\mathrm{T}}$$

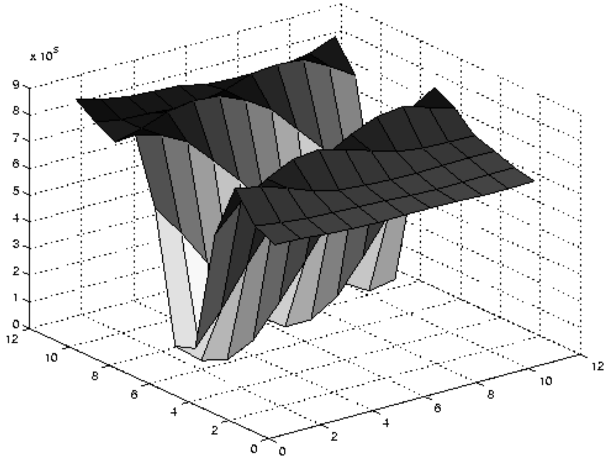
– gradients have small magnitude

– small λ_1, small λ_2

Slide credit: S. Lazebnik

Edge





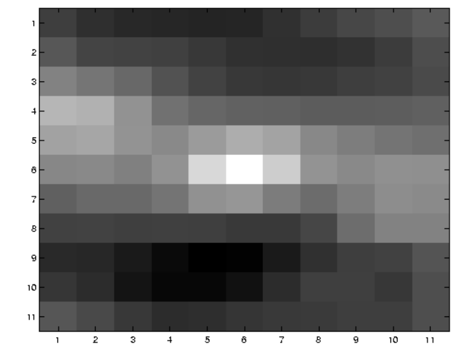
$$\begin{bmatrix} \sum I_x I_x & \sum I_x I_y \\ \sum I_x I_y & \sum I_y I_y \end{bmatrix} = \sum \nabla I (\nabla I)^T$$

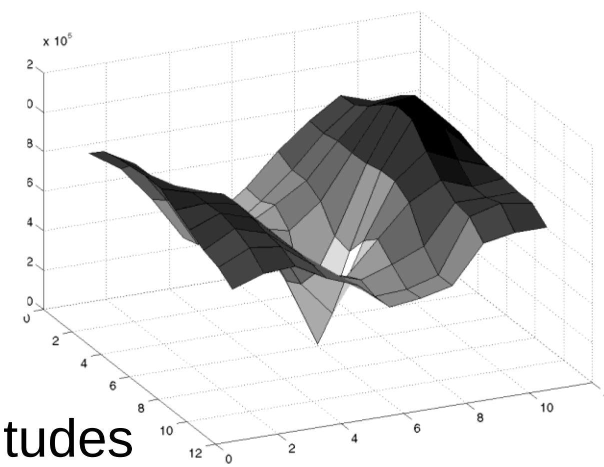
– large gradients, all the same

– large λ_1, small λ_2

Slide credit: S. Lazebnik

High texture region





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

– gradients are different, large magnitudes

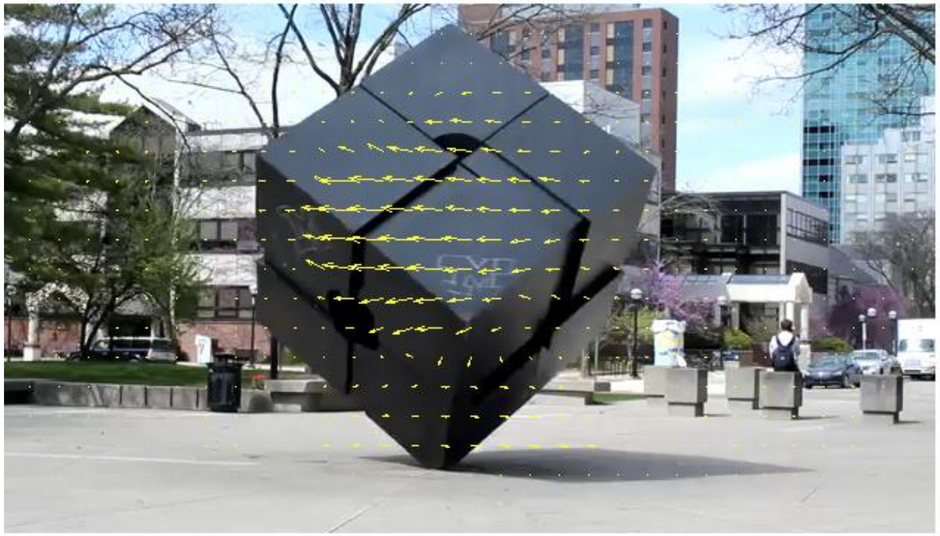
– large λ_1, large λ_2

Slide credit: S. Lazebnik

Lucas-Kanade flow example

Input frames

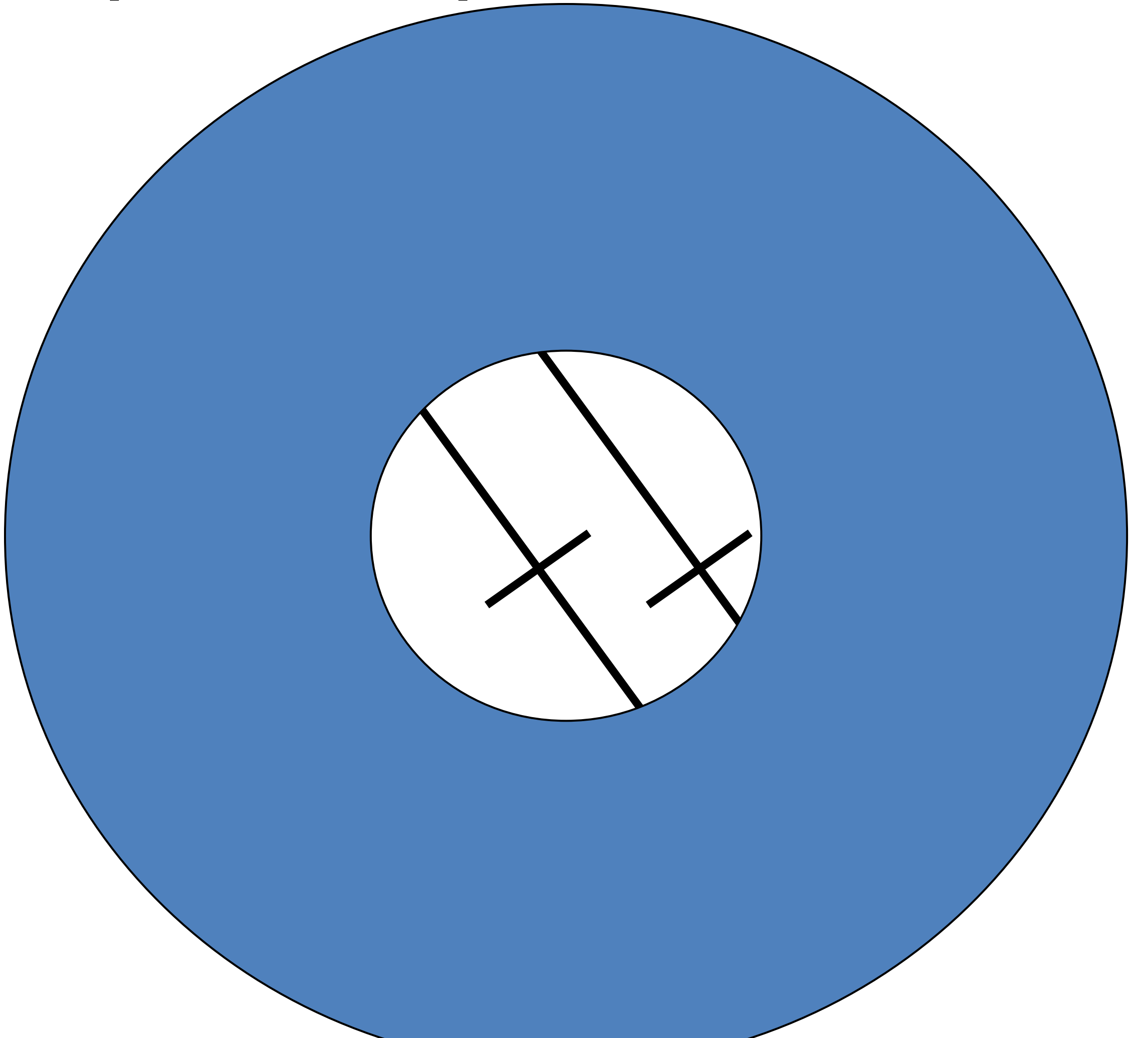
Output



Slide credit: S. Lazebnik

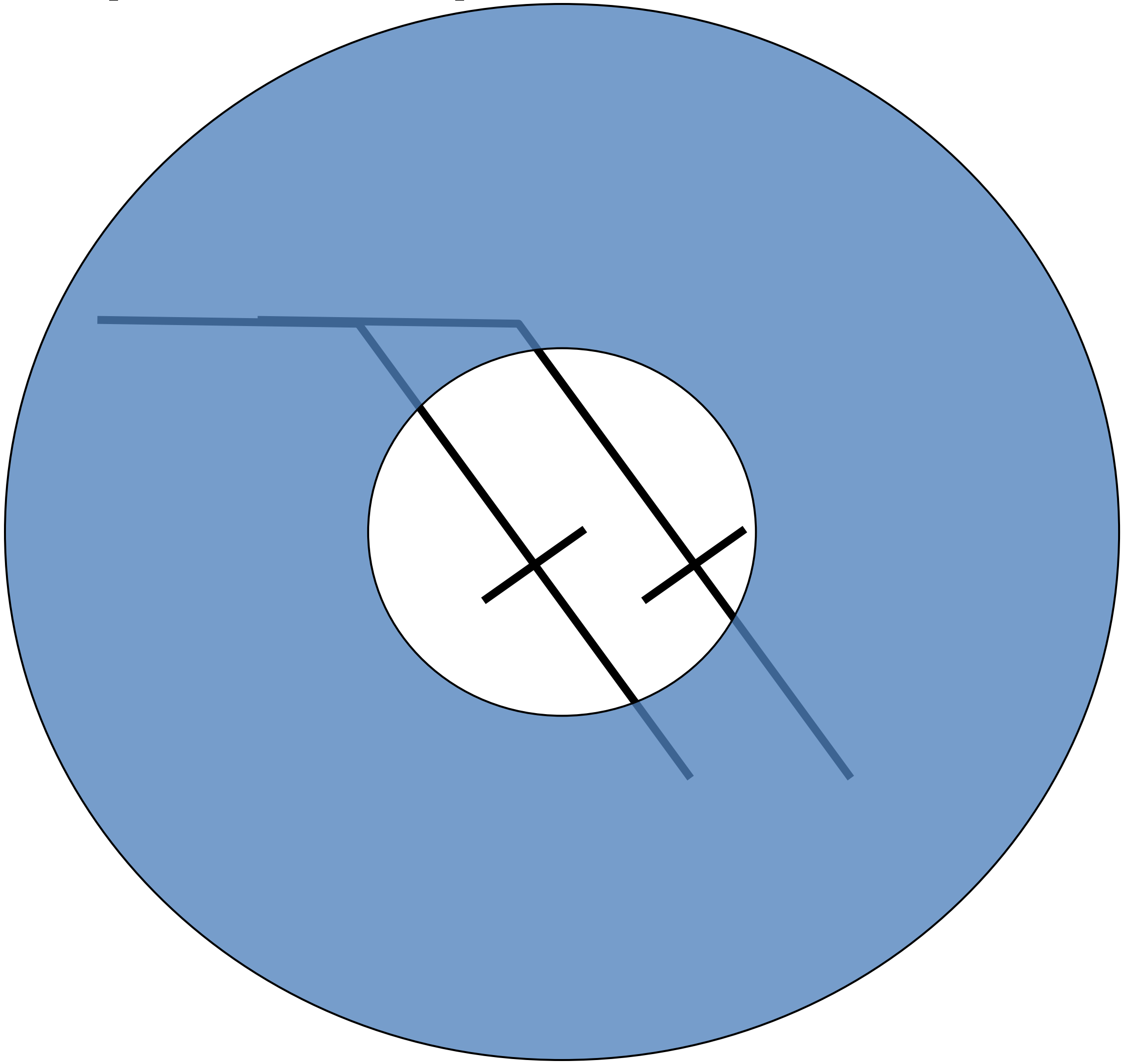
Source: [MATLAB Central File Exchange]

Aperture problem Take 2



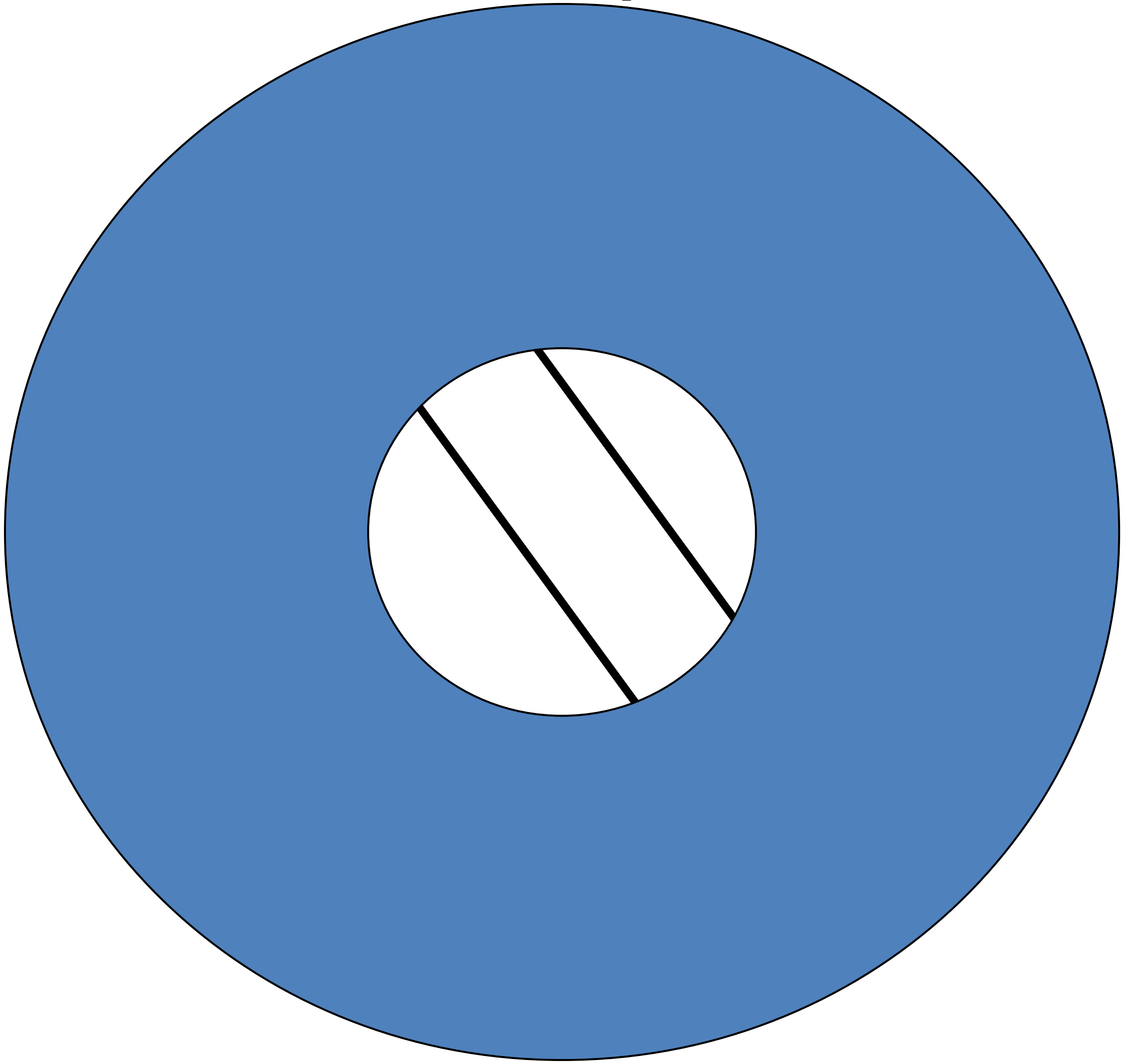
Slide credit: S. Lazebnik

Aperture problem Take 2



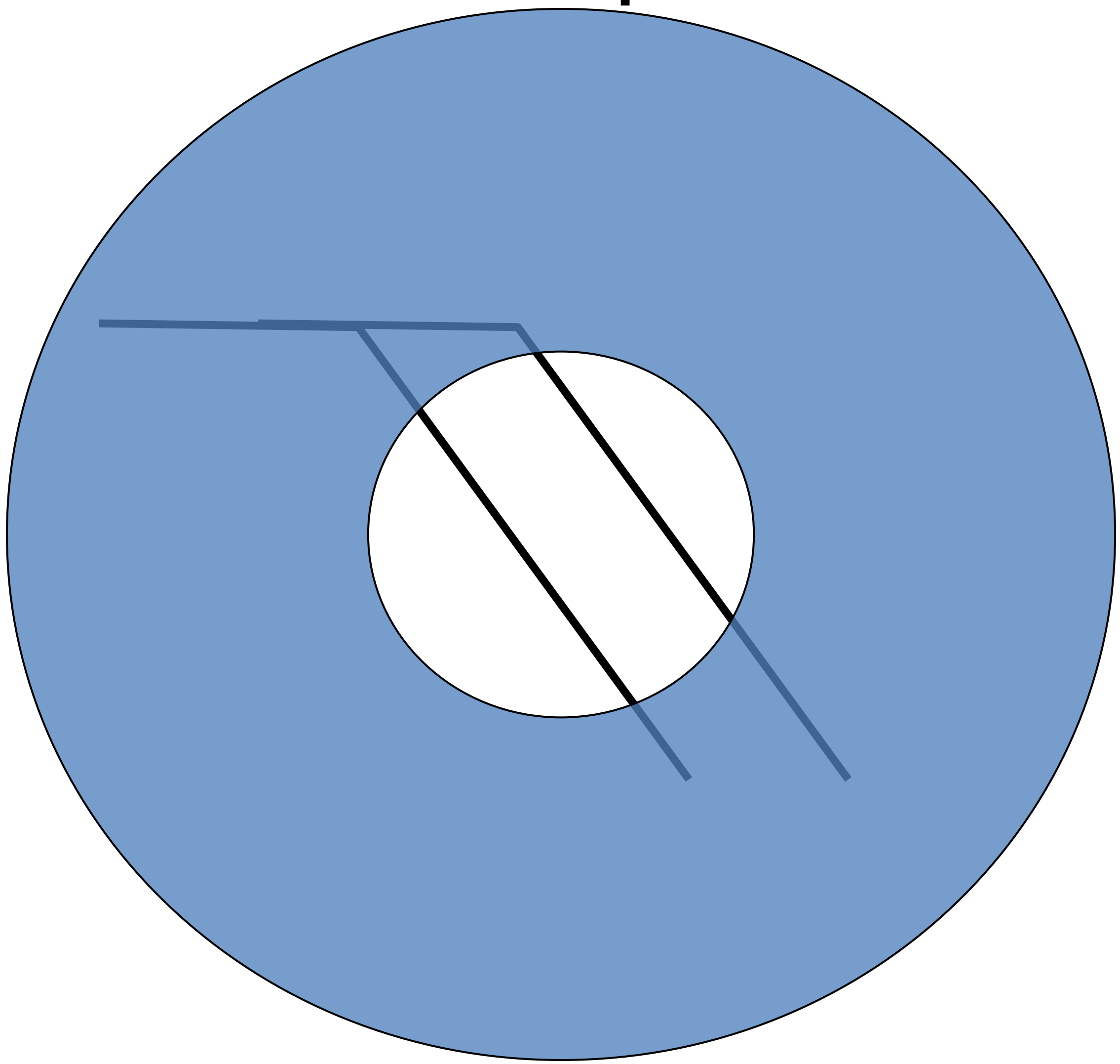
Slide credit: S. Lazebnik

For Comparison



Slide credit: S. Lazebnik

For Comparison



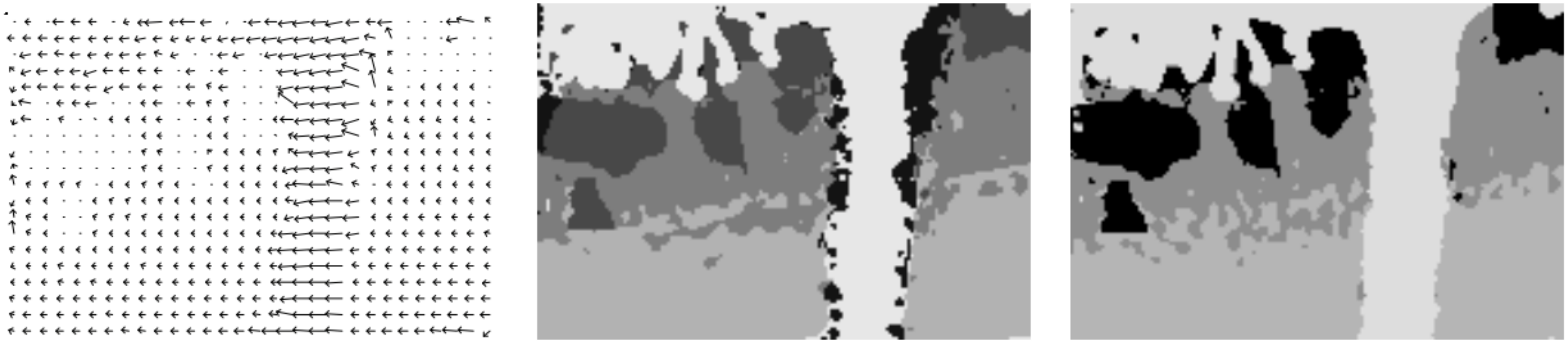
Slide credit: S. Lazebnik

So How Does This Fail?

- Point doesn’t move like neighbors:
  - **Why would this happen?**
  - Figure out which points move together, then come back and fix.

So How Does This Fail?

- Point doesn't move like neighbors:
  - **Why would this happen?**
  - Figure out which points move together, then come back and fix



(a) (b) (c)

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

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

So How Does This Fail?

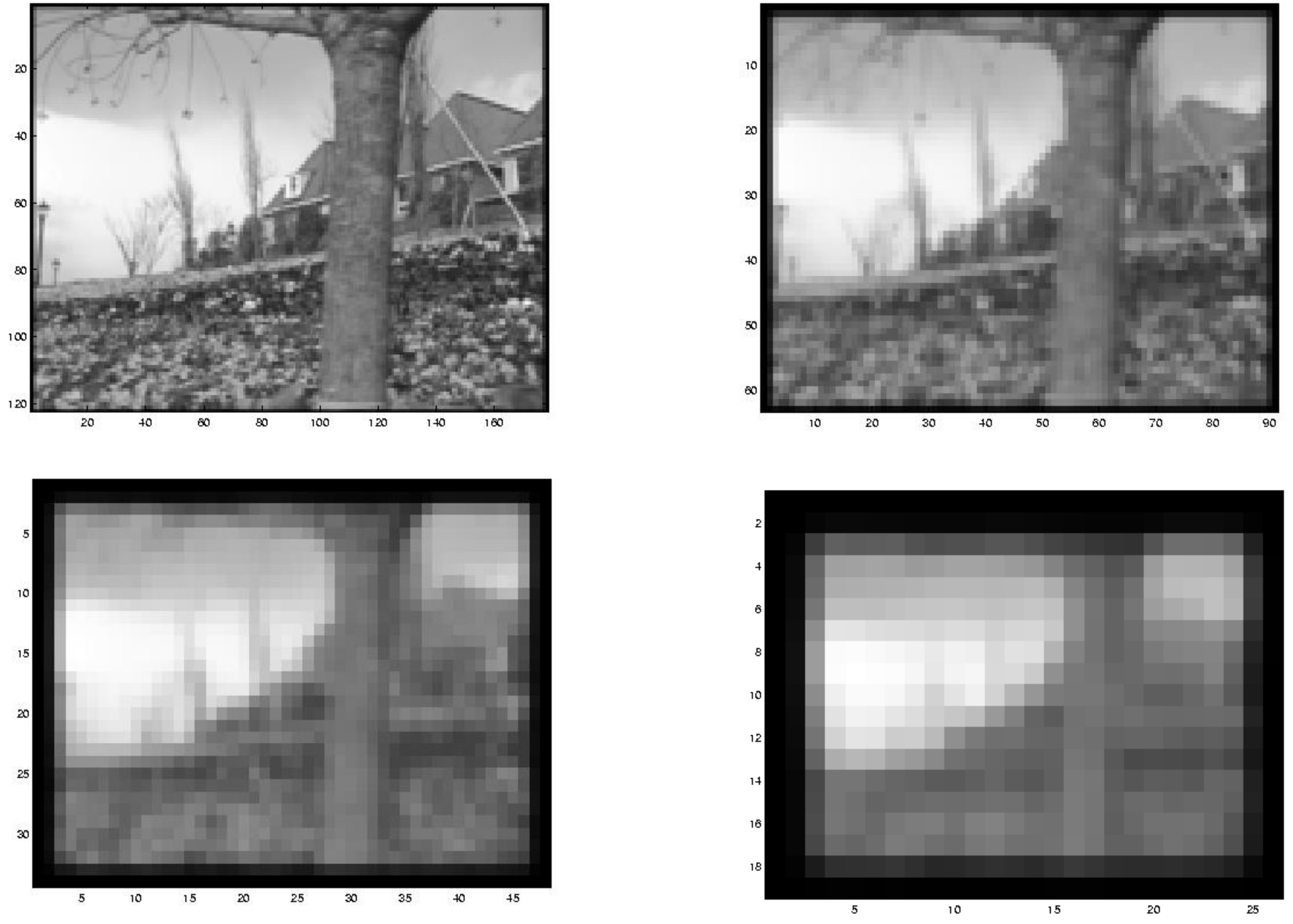
- Point doesn't move like neighbors:
  - **Why would this happen?**
  - Figure out which points move together, then come back and fix.
- Brightness constancy isn't true
  - **Why would this happen?**
  - Solution: other form of matching (e.g.interest points)
- Taylor series is bad approximation
  - **Why would this happen?**
  - Solution: Make your pixels big

Revisiting small motions

- Is this motion small enough?
  - Probably not—it's much larger than one pixel
  - How might we solve this problem?

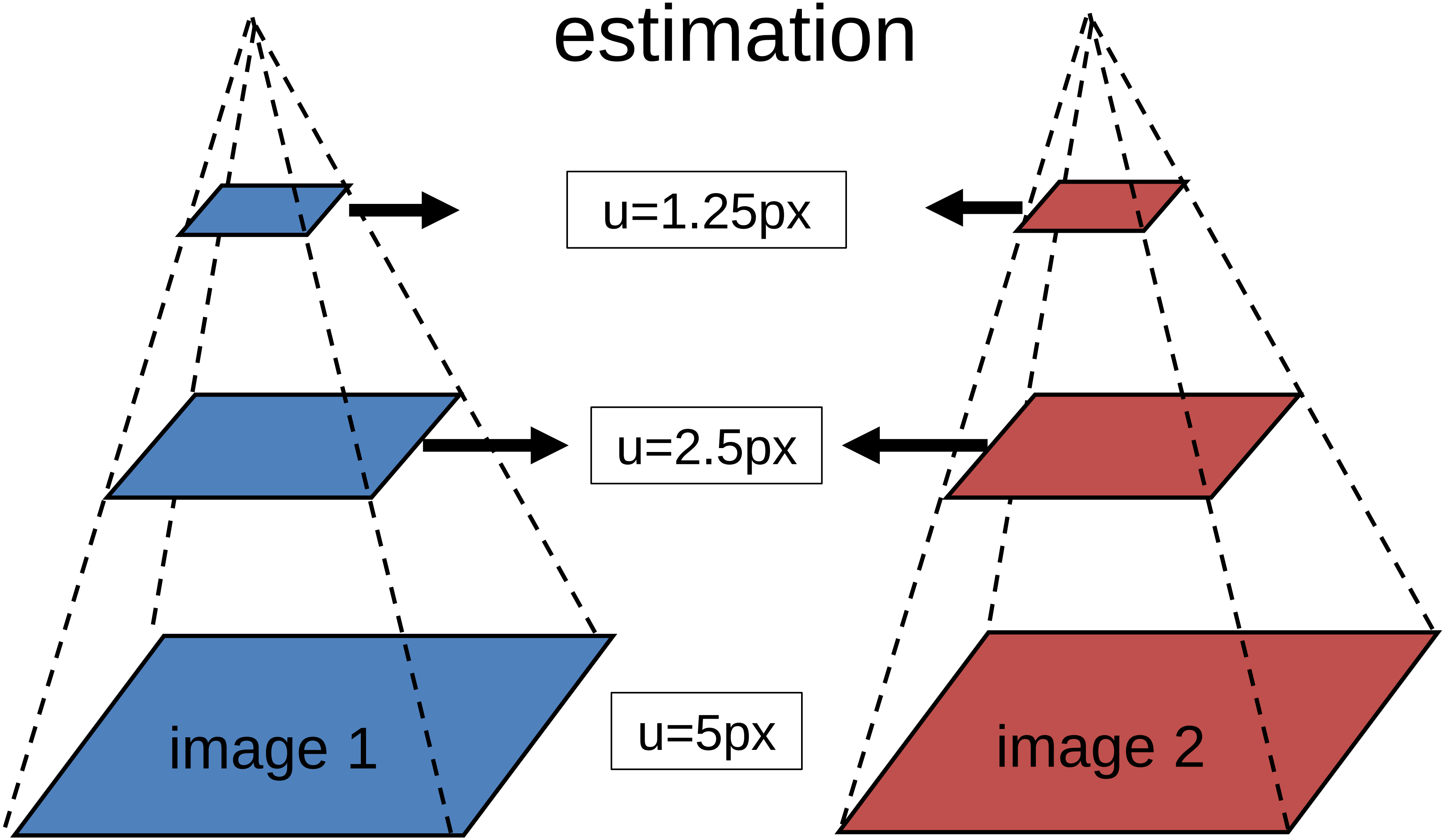
Slide credit: S. Lazebnik

Reduce the resolution!



Slide credit: S. Lazebnik

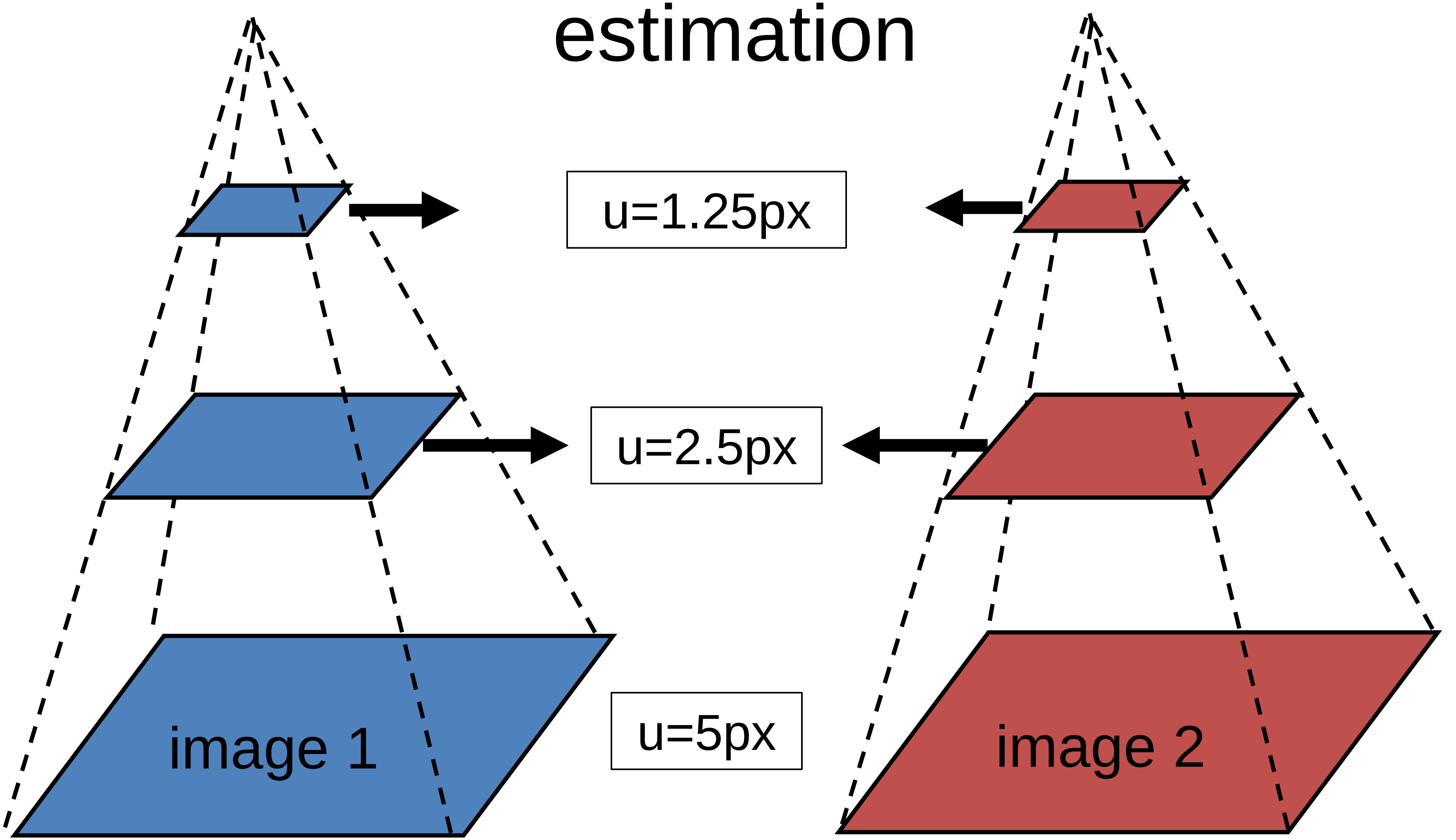
Coarse-to-fine optical flow estimation



Typically called Gaussian Pyramid

Slide credit: S. Lazebnik

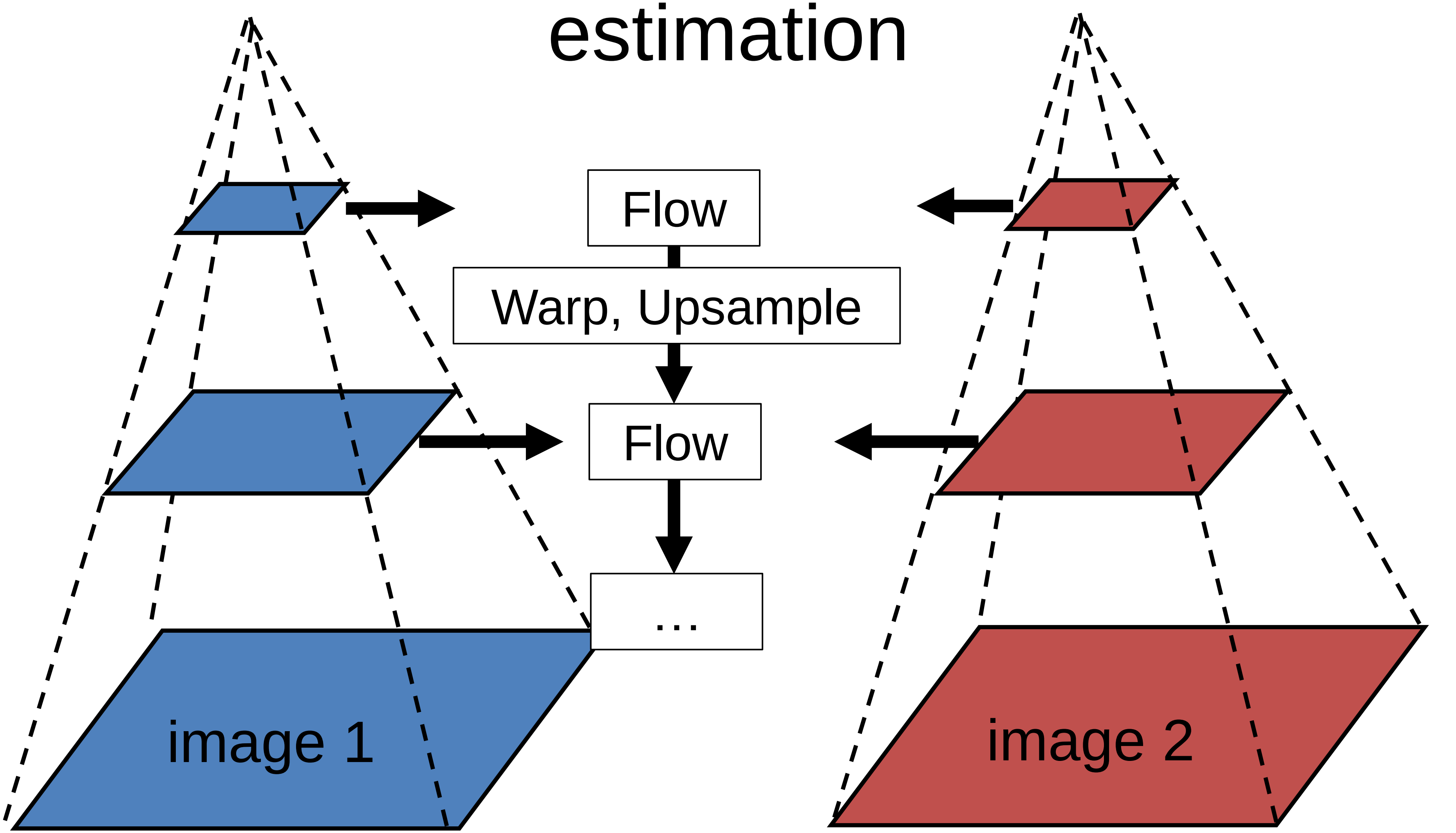
Coarse-to-fine optical flow estimation



Do we start at bottom or top to align?

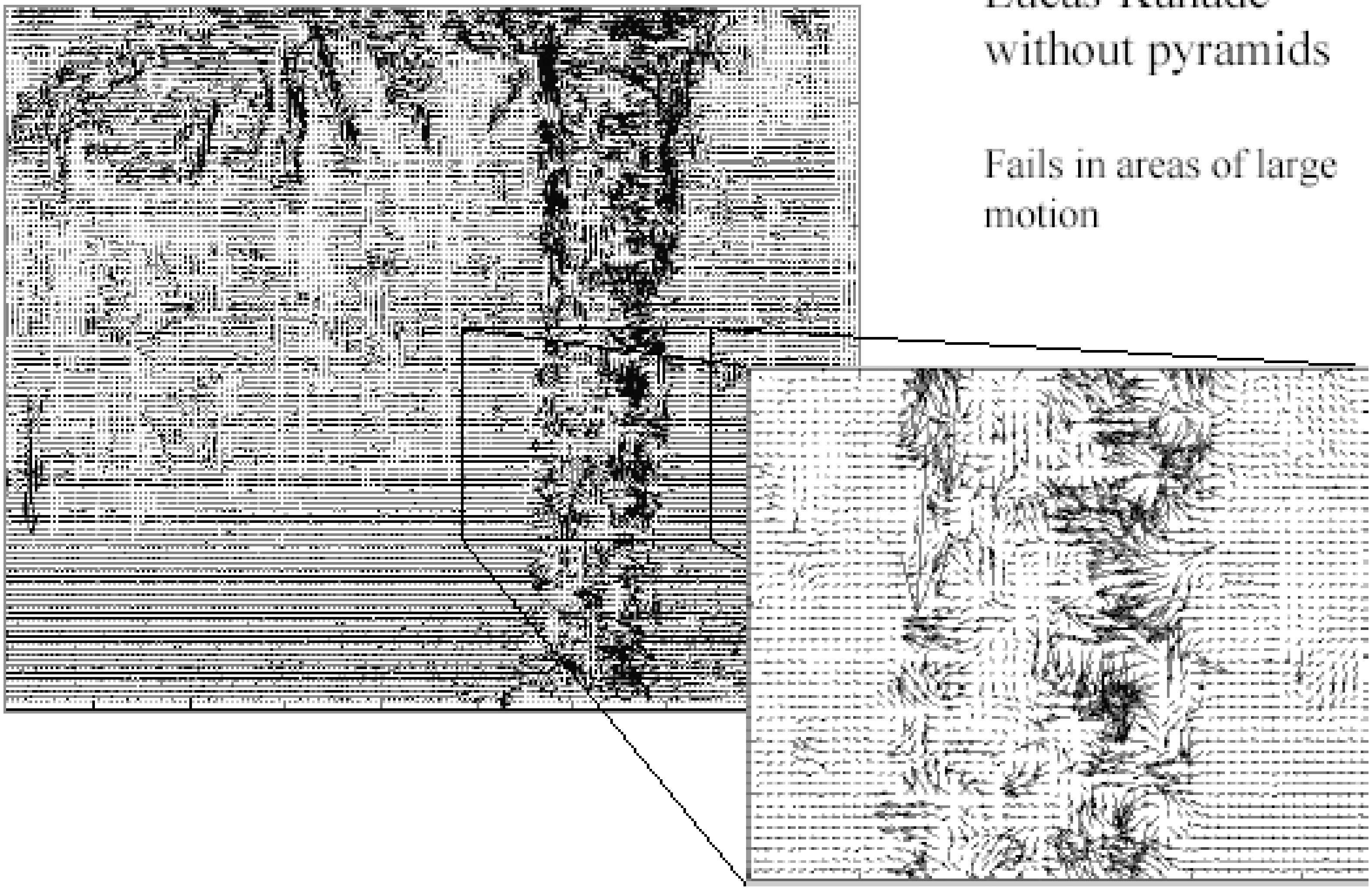
Slide credit: S. Lazebnik

Coarse-to-fine optical flow estimation



Slide credit: S. Lazebnik

Optical Flow Results

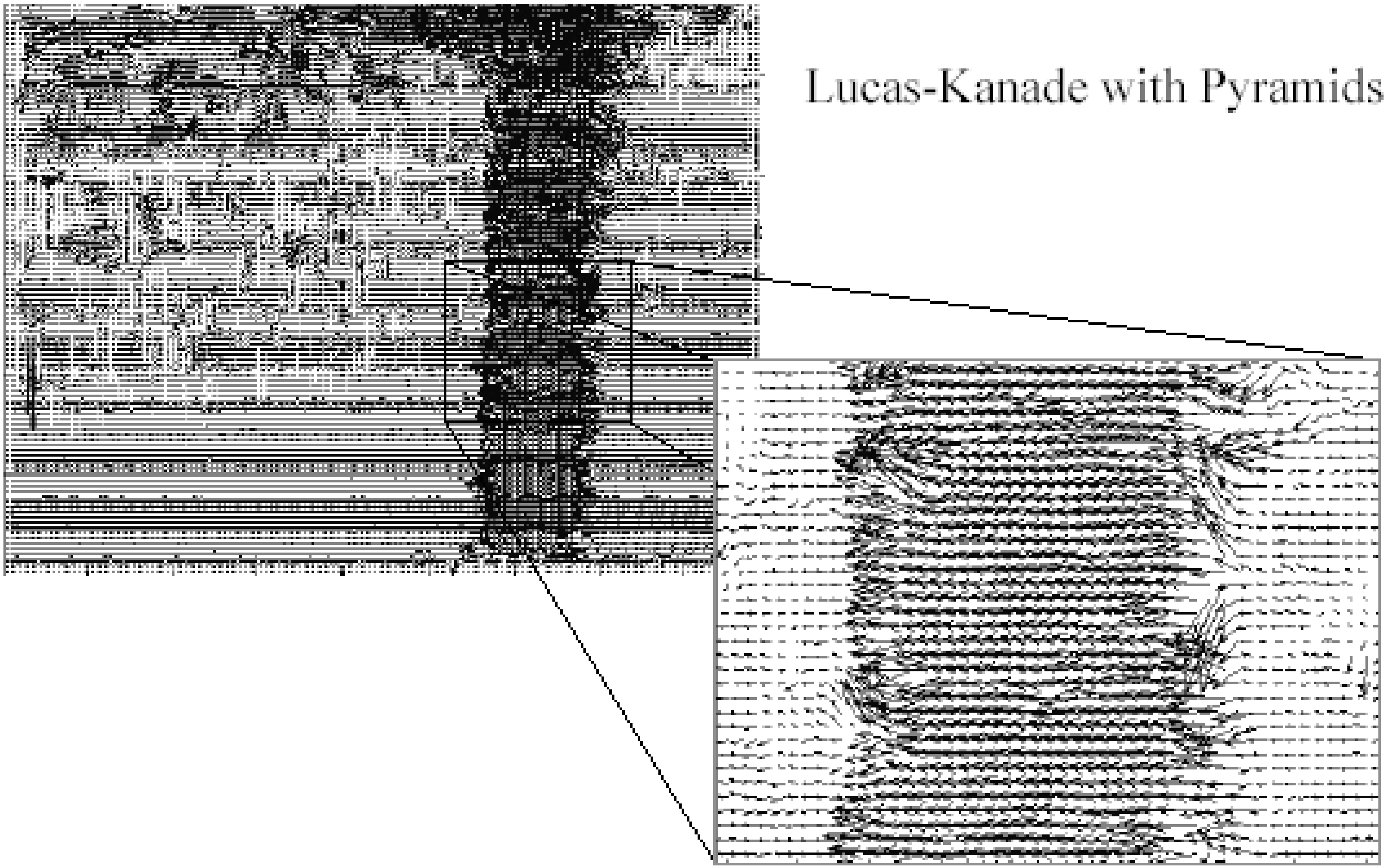


Lucas-Kanade without pyramids

Fails in areas of large motion

Slide credit: K. Hassan-Shafique

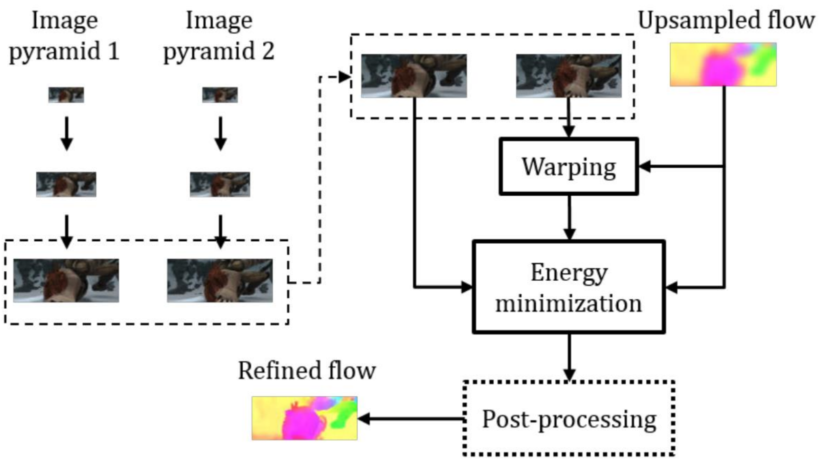
Optical Flow Results



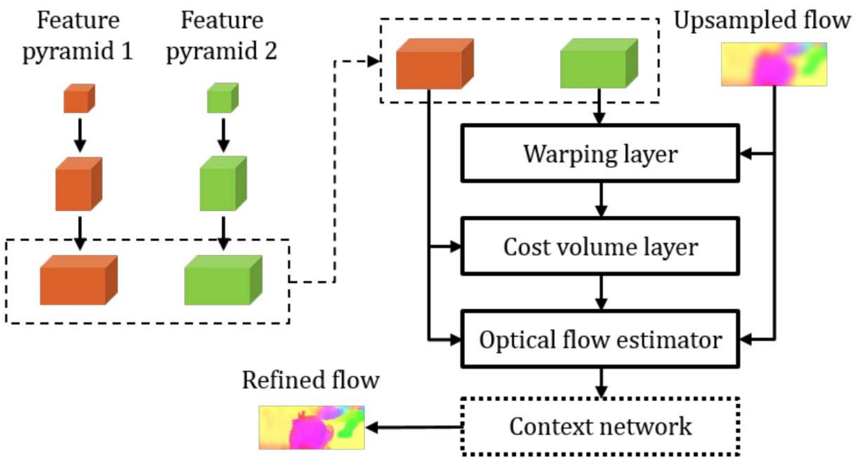
Slide credit: K. Hassan-Shafique

Flow CNNs

Match CNN features instead of pixels!



Traditional coarse-to-fine flow

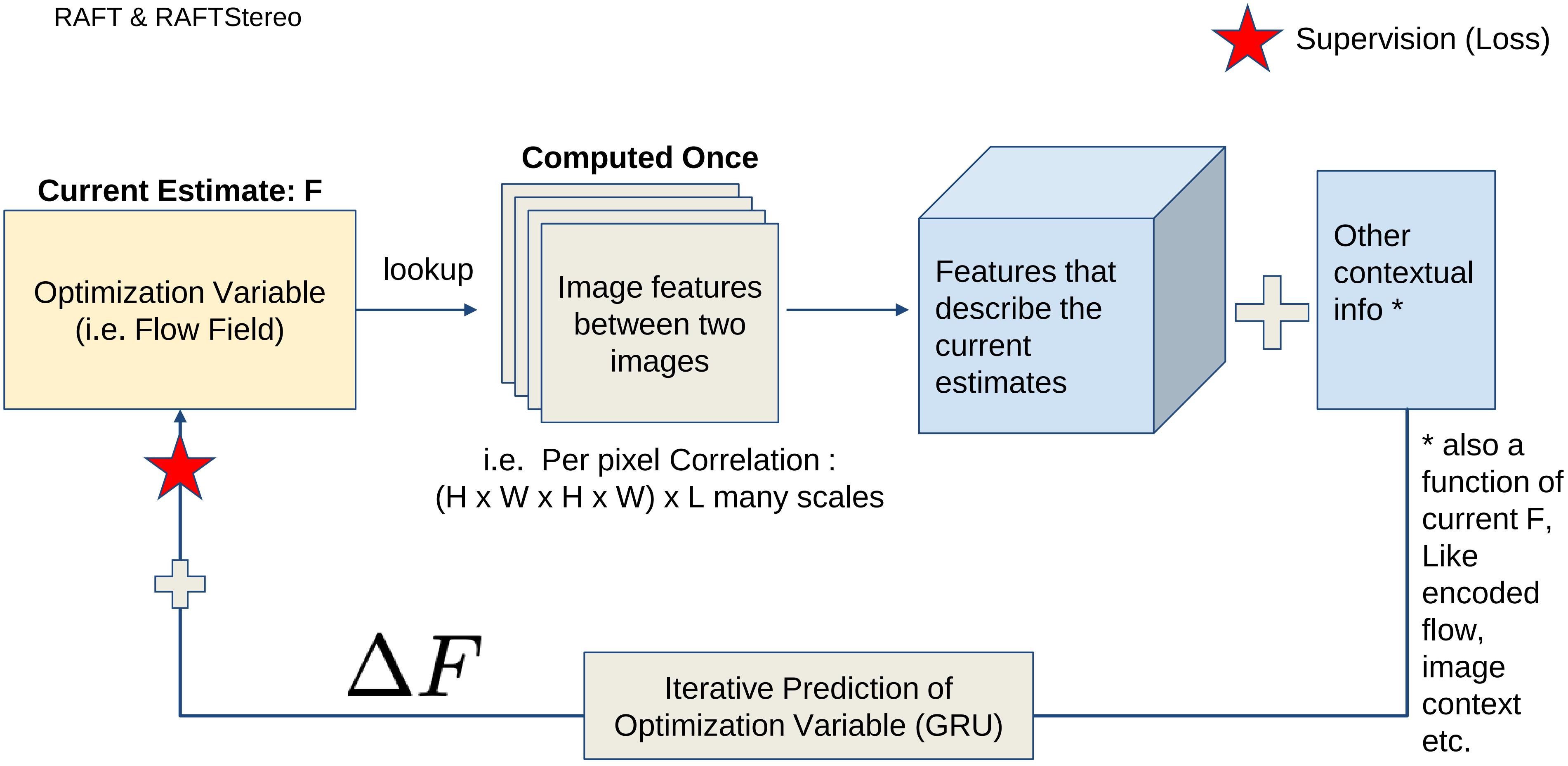


PWC-net

[Sun et al., "PWC-Net", 2018]

RAFT: Feature Look up based on current (flow) estimates

RAFT & RAFTStereo



Training Data

Flying Chairs Dataset

Fischer et al. 2015. https://arxiv.org/abs/1504.06852

Training Data

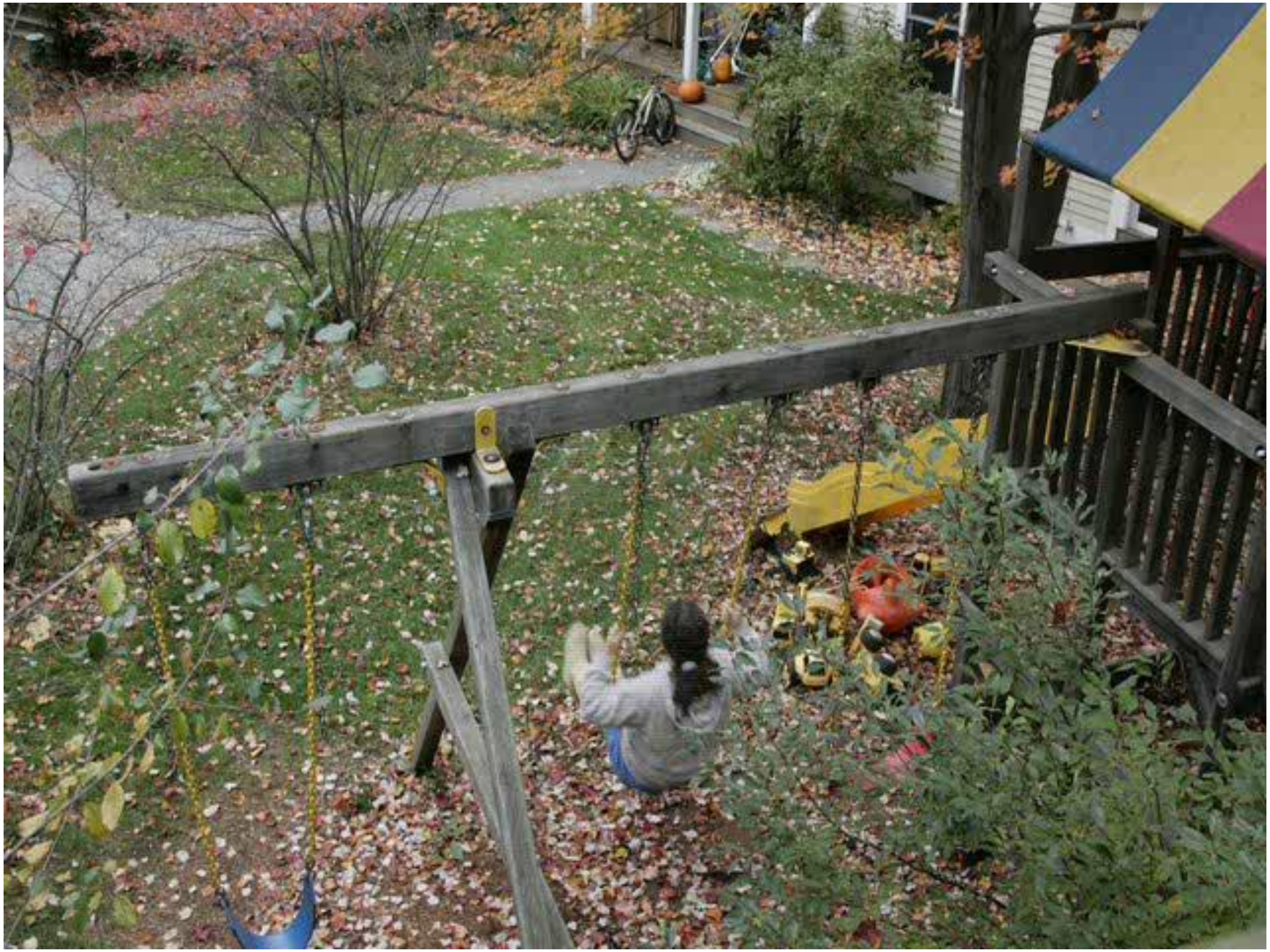
Are we ready for Autonomous Driving? The KITTI Vision Benchmark Suite. Geiger et al. CVPR 2012.

Cute Application: Motion Magnification

Idea: take flow, magnify it

Example credit: C. Liu

Motion Magnification



Example credit: C. Liu

Motion Magnification

Example credit: C. Liu