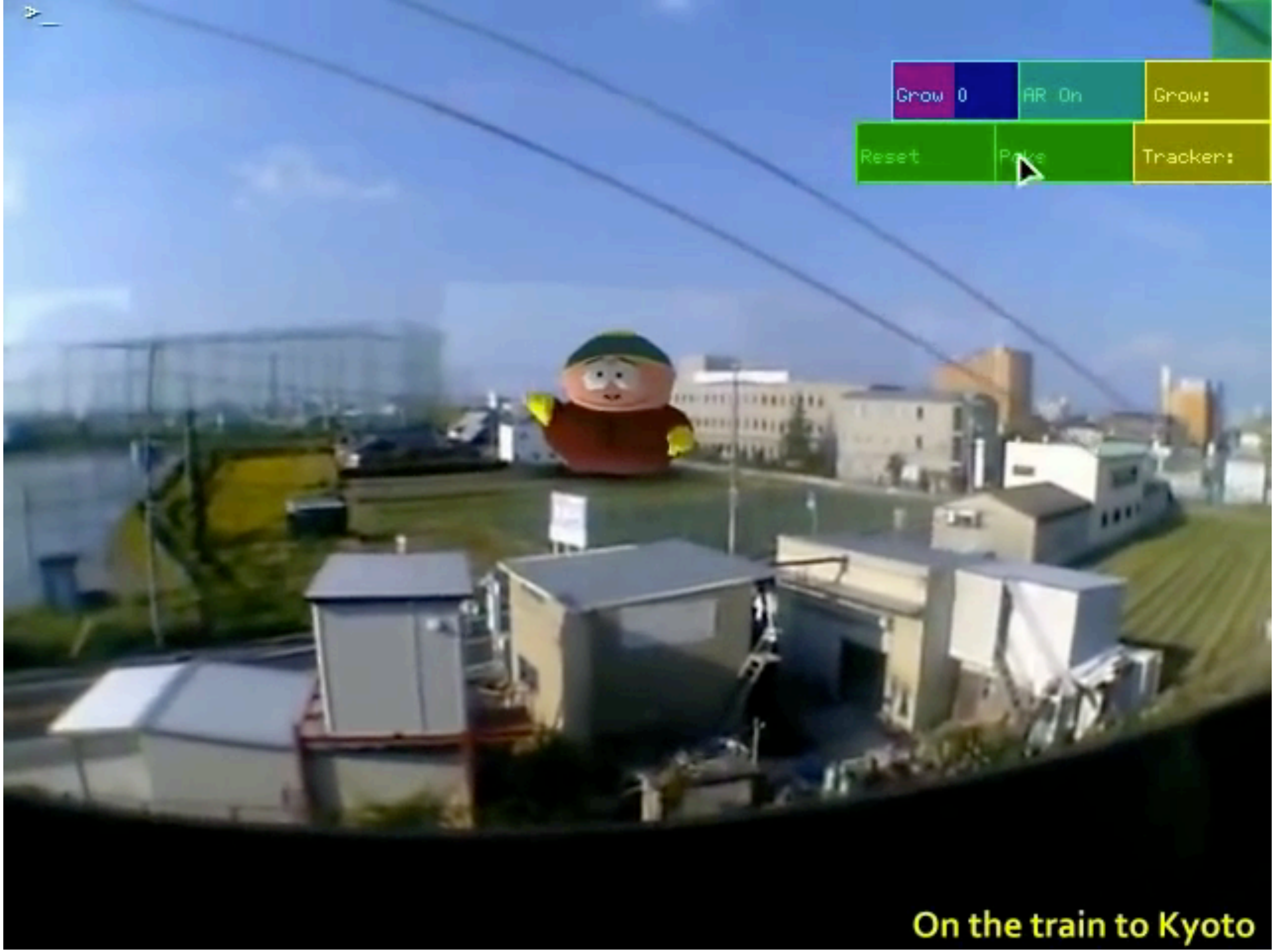
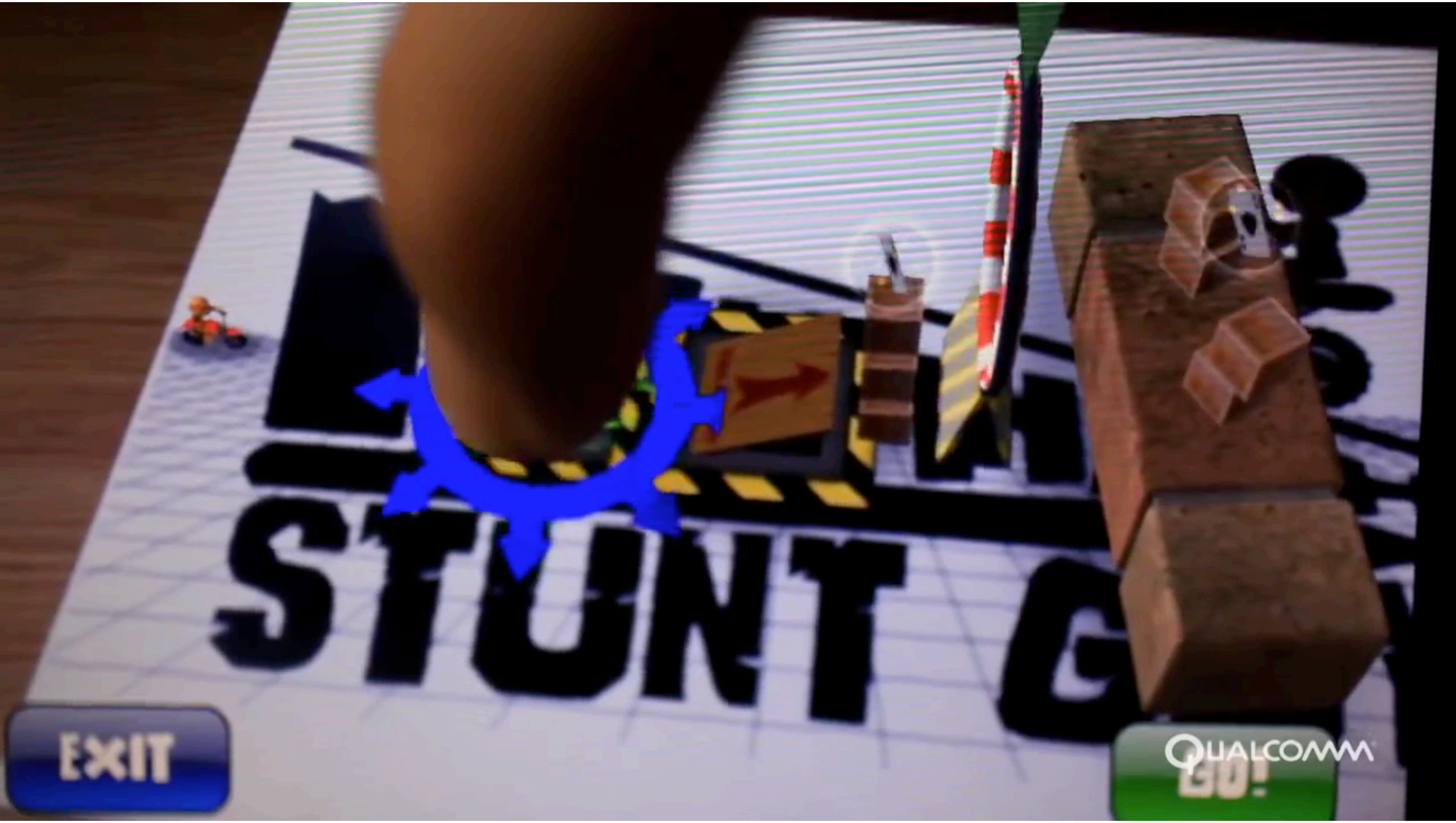


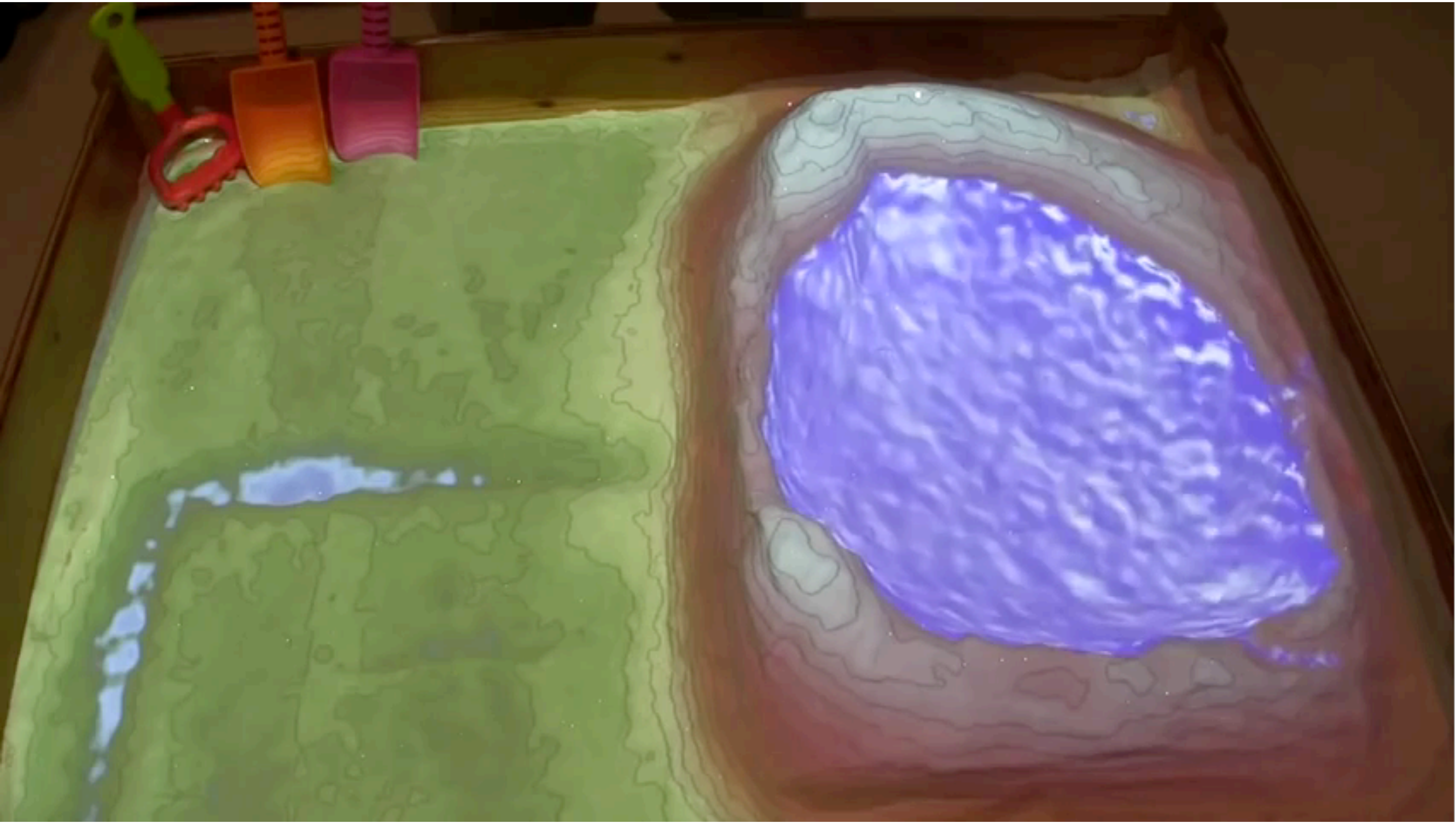
3D Reconstruction with Computer Vision

Meeting 14: Video and Tracking

CS 378 Fall 2014, UT Austin, Bryan Klingner, 2 October

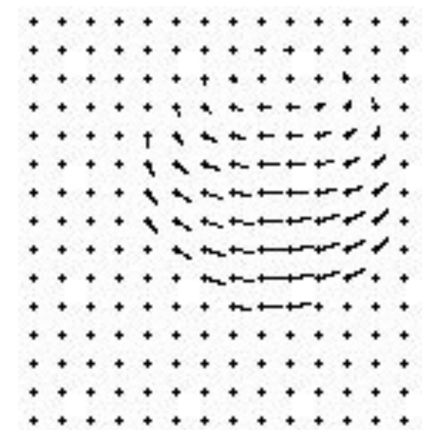
On the train to Kyoto

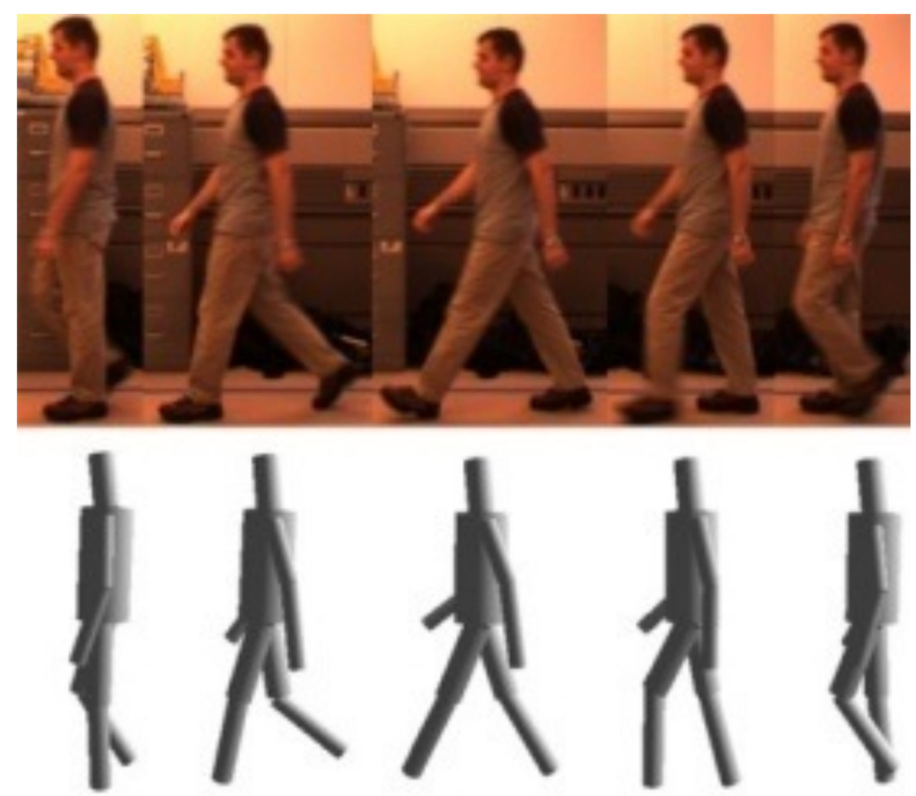

STUNT
EXIT
QUALCOMM
GO!



Motion and tracking

Tracking objects, video analysis, low level motion



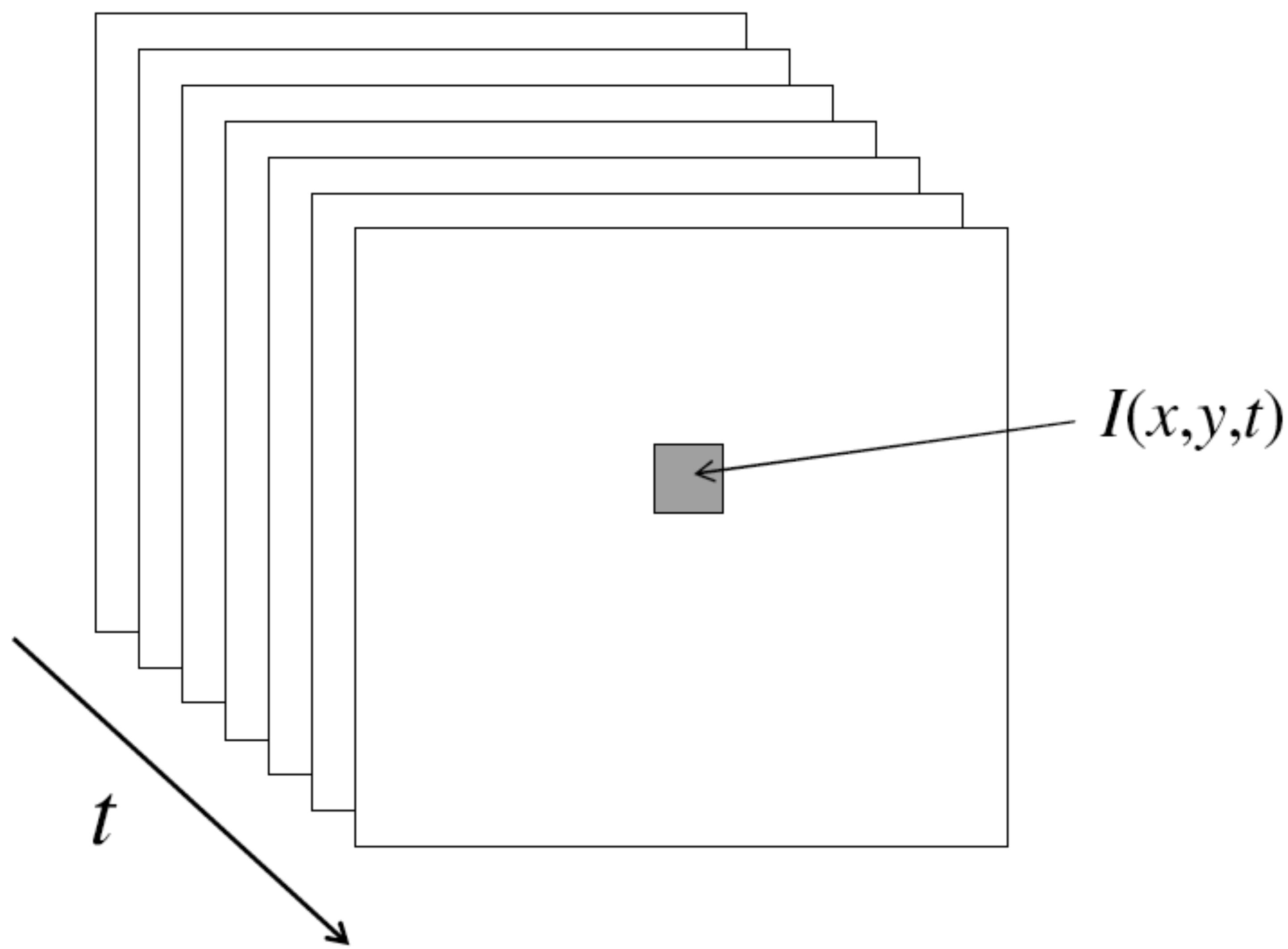


Tomas Izo

Slides by Steve Seitz, Kristen Grauman, others

Video

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

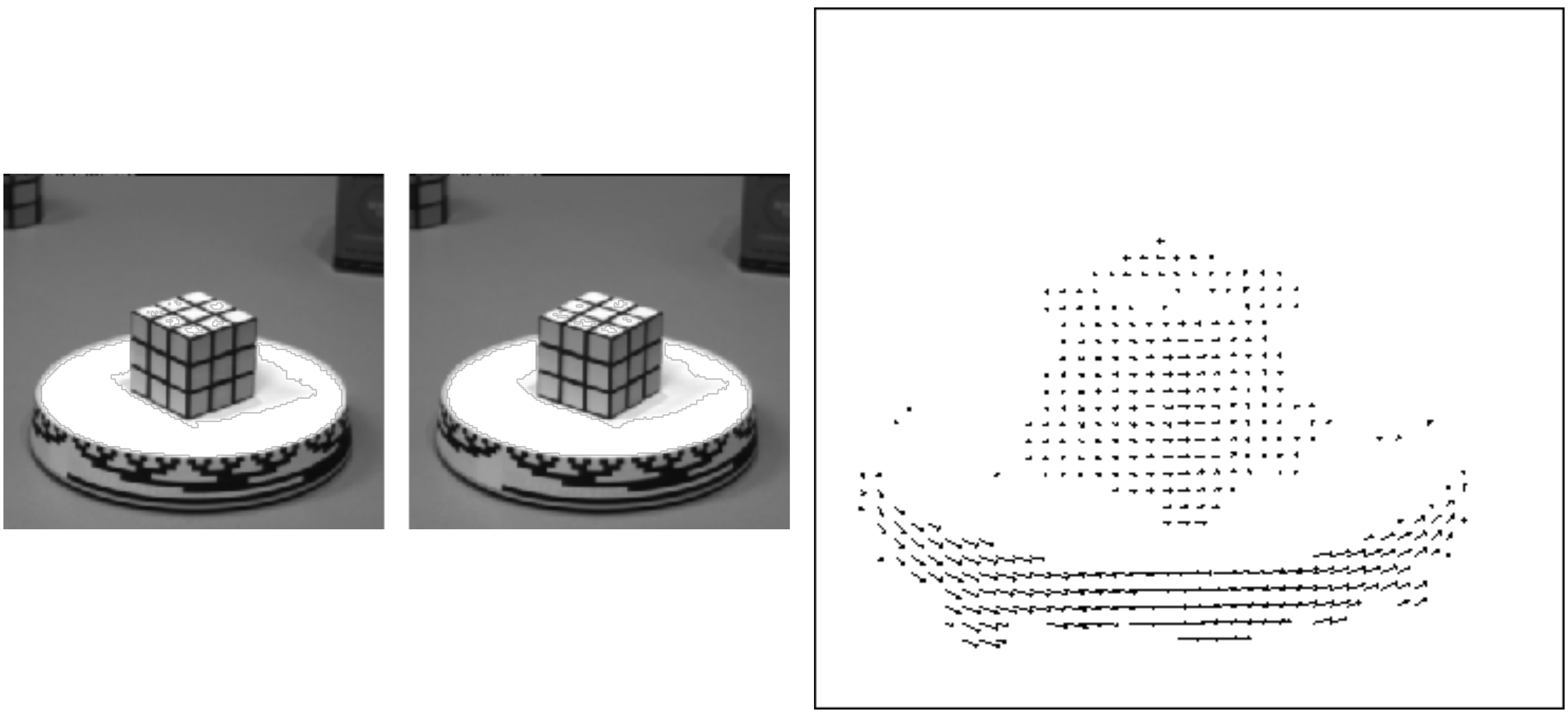


Uses of motion

- Estimating 3D structure
- Segmenting objects based on motion cues
- Learning dynamical models
- Recognizing events and activities
- Improving video quality (motion stabilization)

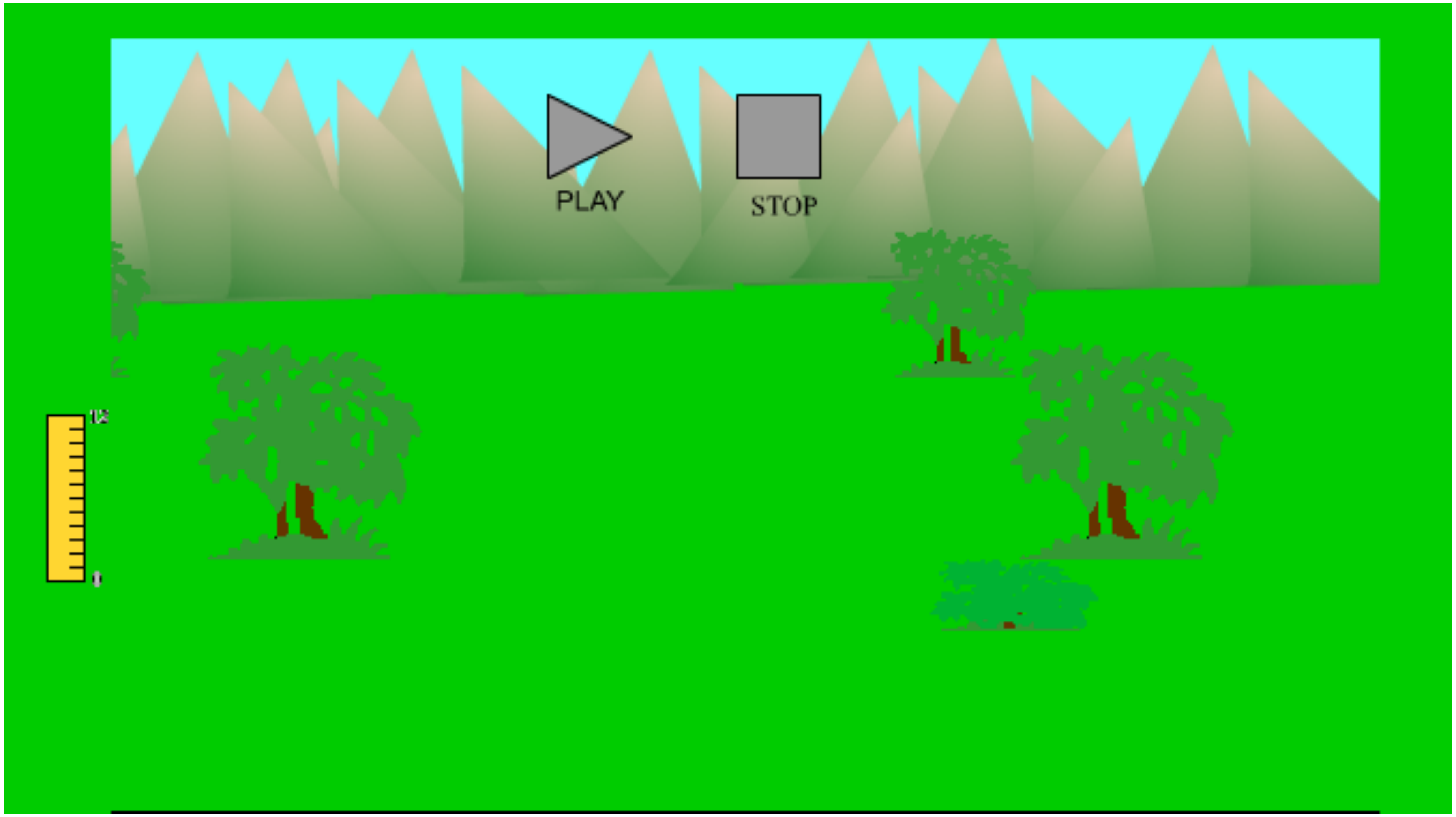
Motion field

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



Motion parallax

http://psych.hanover.edu/KRANTZ/MotionParallax/MotionParallax.html



Motion field + camera motion

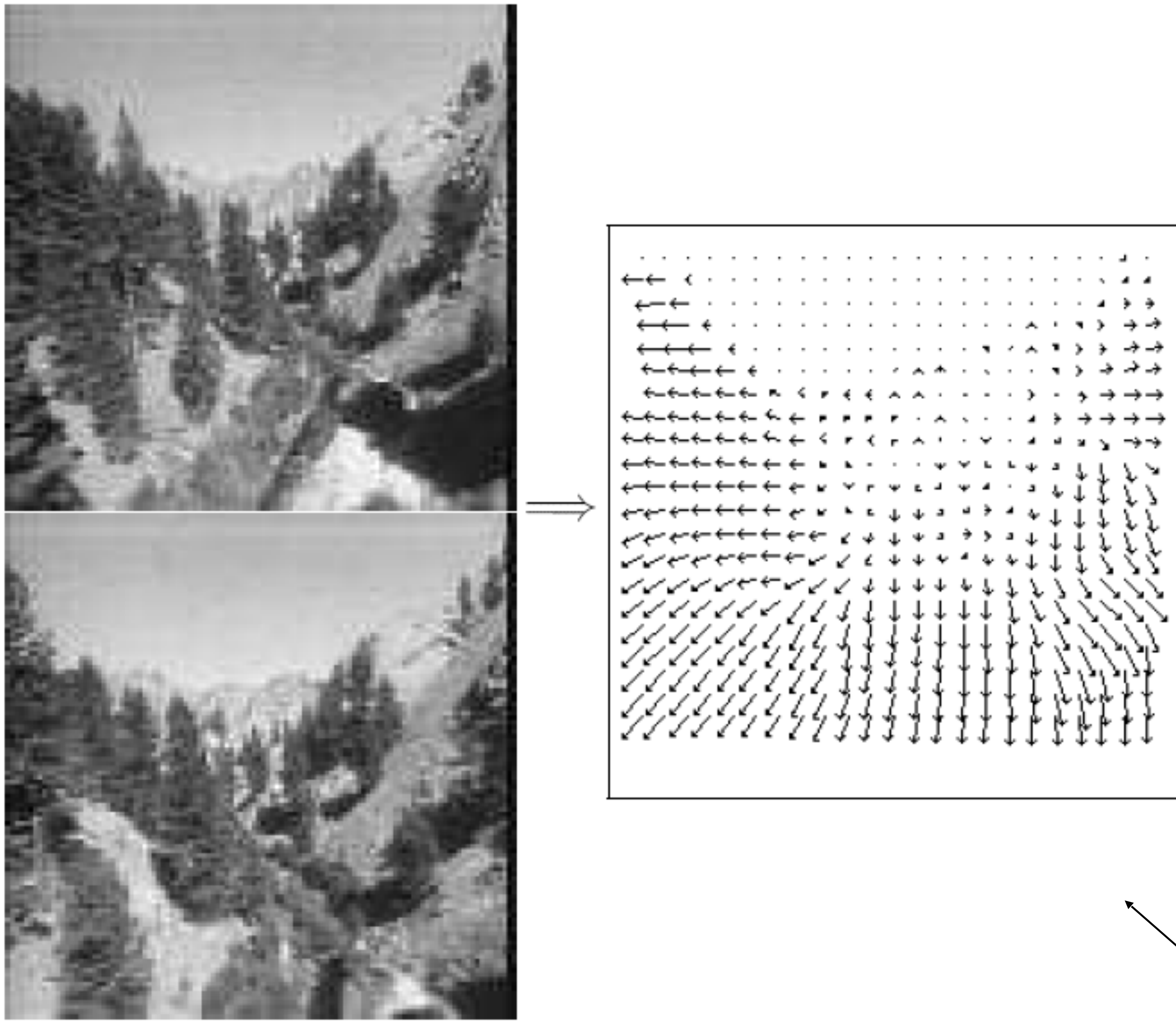


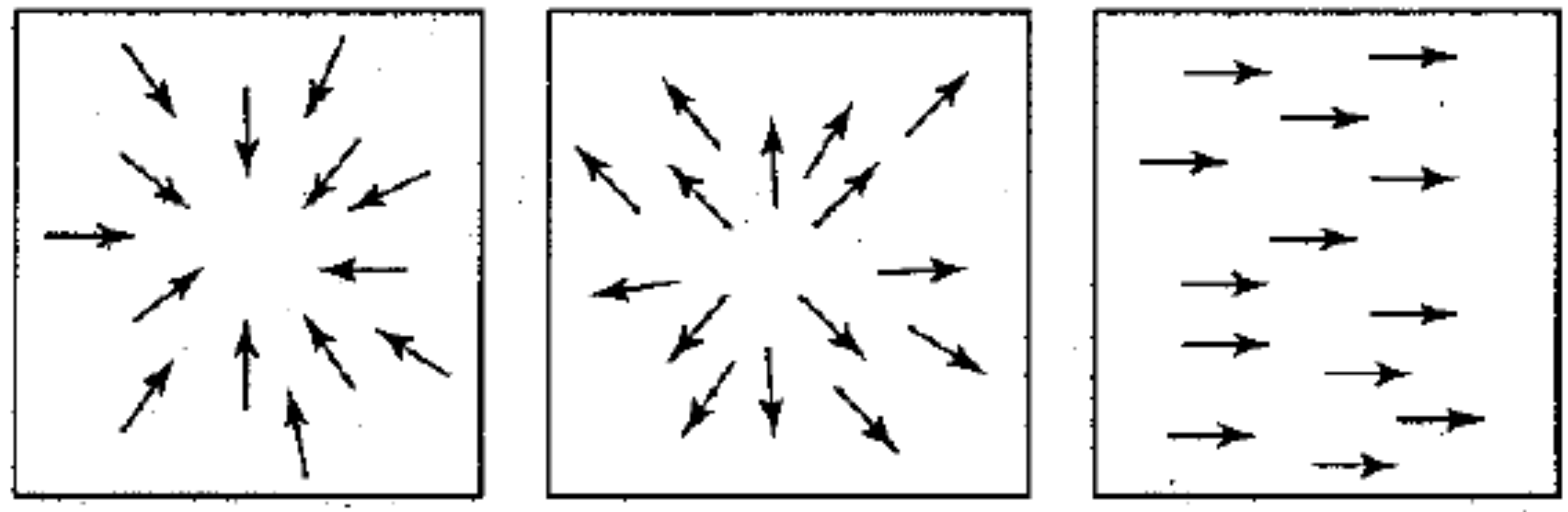
Figure 1.2: Two images taken from a helicopter flying through a canyon and the computed optical flow field.

Length of flow vectors inversely proportional to depth Z of 3d point

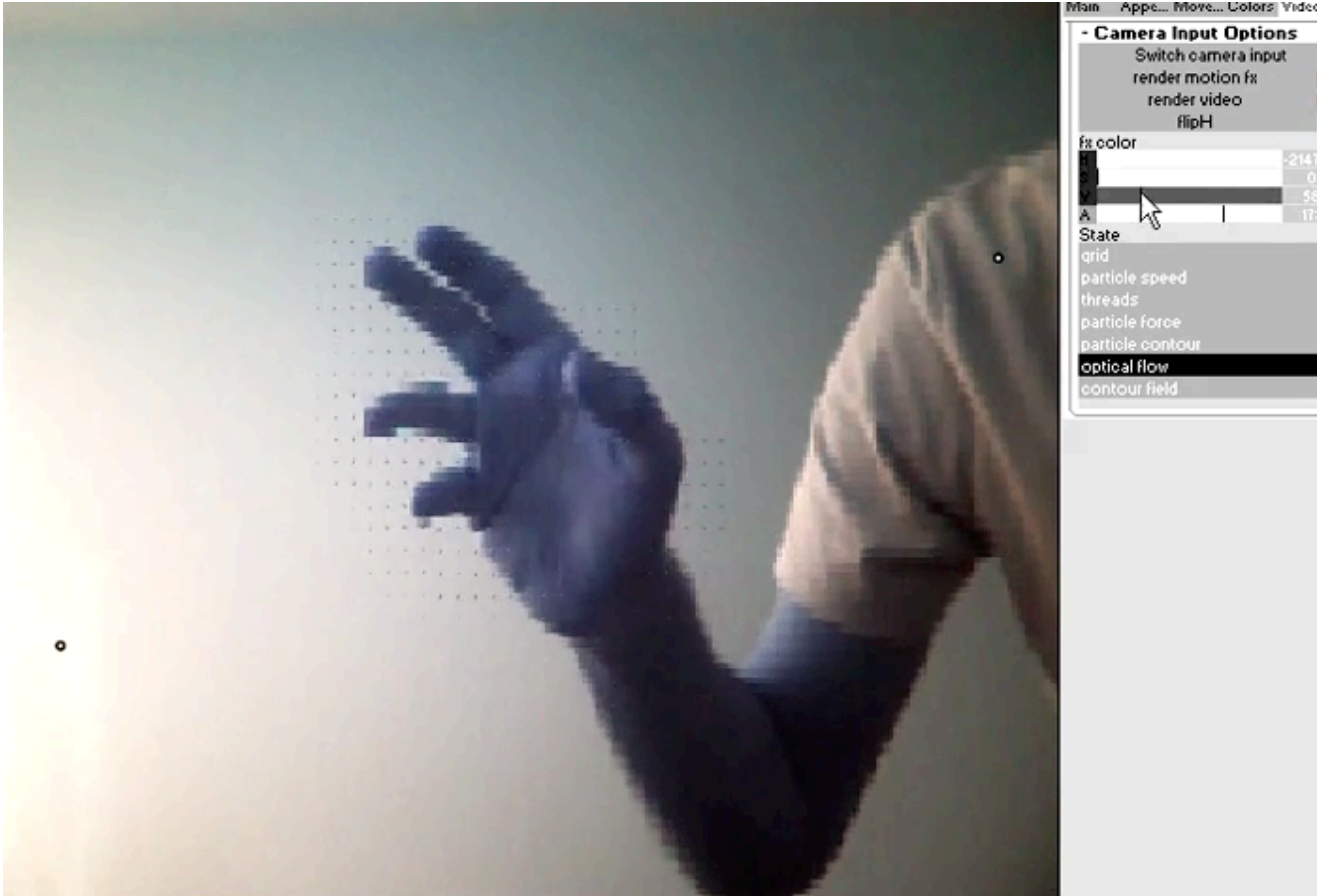
points closer to the camera move more quickly across the image plane

Figure from Michael Black, Ph.D. Thesis

Motion field + camera motion



Zoom out **Zoom in** **Pan right to left**


Main
Appe...
Move...
Colors
Video
- Camera Input Options
Switch camera input
render motion fx
render video
flipH
fx color
-21474
0
58
A
173
State
grid
particle speed
threads
particle force
particle contour
optical flow
contour field

Motion estimation techniques

- Direct methods
  - Directly recover image motion at each pixel from spatio-temporal image brightness variations
  - Dense motion fields, but sensitive to appearance variations
  - Suitable for video and when image motion is small
- Feature-based methods
  - Extract visual features (corners, textured areas) and track them over multiple frames
  - Sparse motion fields, but more robust tracking
  - Suitable when image motion is large (10s of pixels)

Optical flow

- Definition: optical flow is the ***apparent*** motion of brightness patterns in the image
- Ideally, optical flow would be the same as the motion field
- Have to be careful: apparent motion can be caused by lighting changes without any actual motion

Apparent motion != motion field

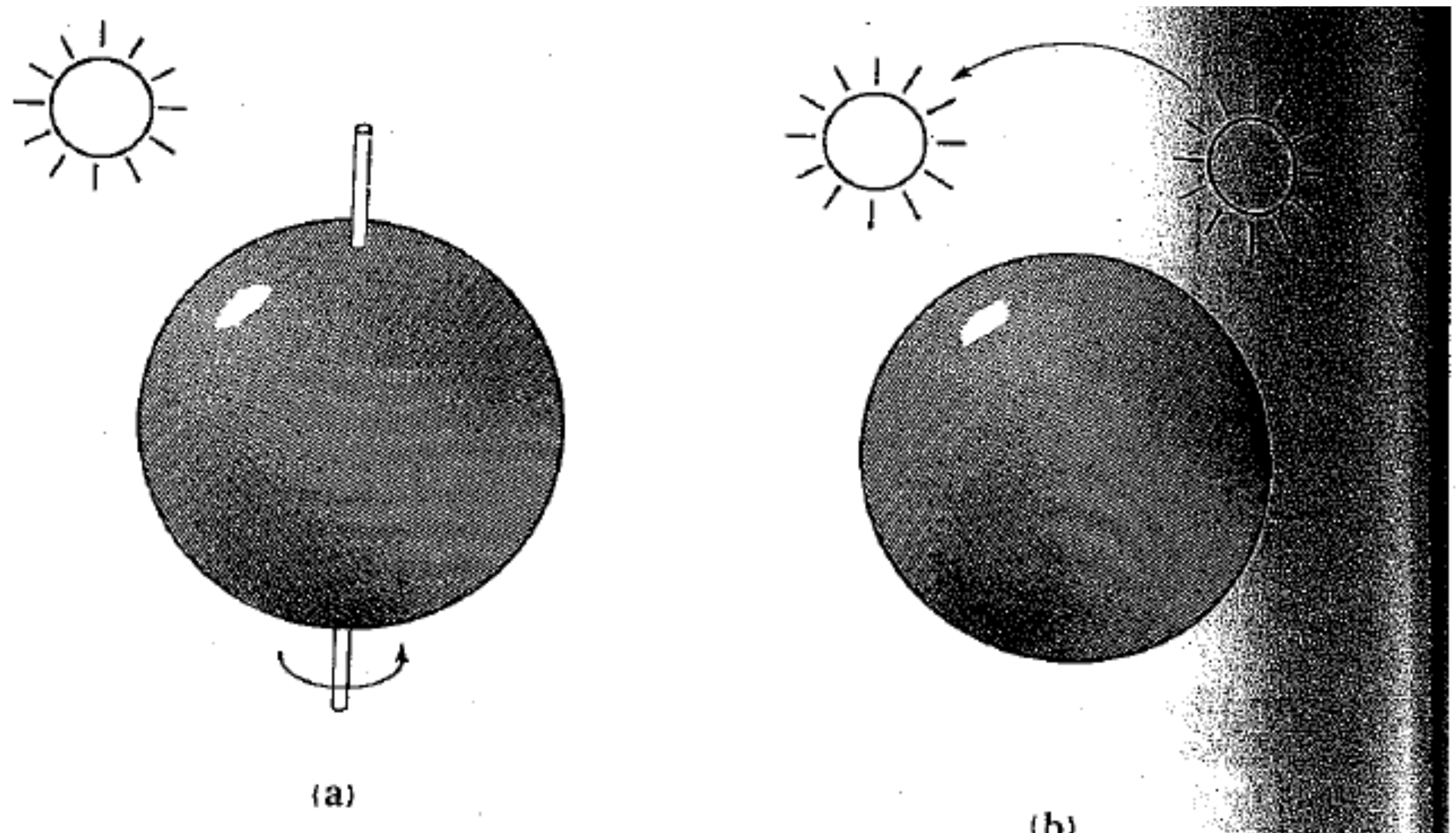
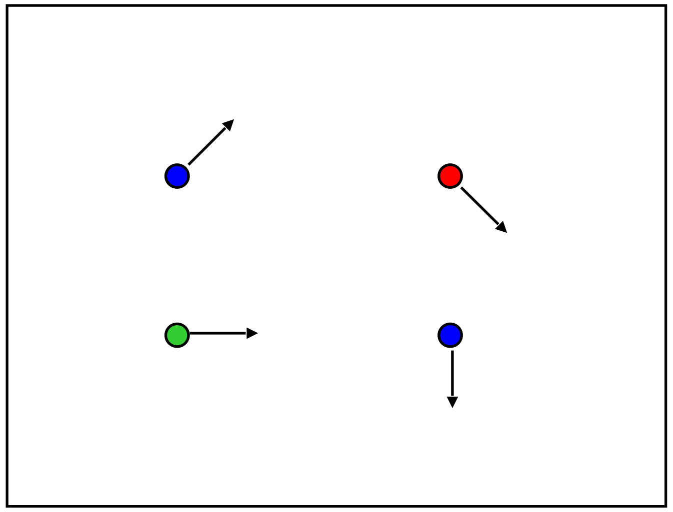


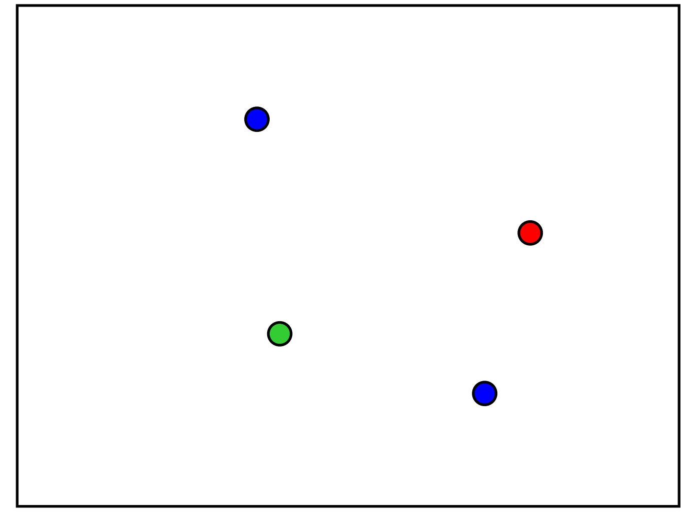
Figure 12-2. The optical flow is not always equal to the motion field. In (a) a smooth sphere is rotating under constant illumination—the image does not change, yet the motion field is nonzero. In (b) a fixed sphere is illuminated by a moving source—the shading in the image changes, yet the motion field is zero.

Figure from Horn book

Problem definition: optical flow



$H(x, y)$



$I(x, y)$

How to estimate pixel motion from image H to image I?

- Solve pixel correspondence problem
  - given a pixel in H, look for nearby pixels of the same color in I

Key assumptions

- **color constancy**: a point in H looks the same in I
  - For grayscale images, this is **brightness constancy**
- **small motion**: points do not move very far

This is called the **optical flow** problem

Steve Seitz

Brightness constancy

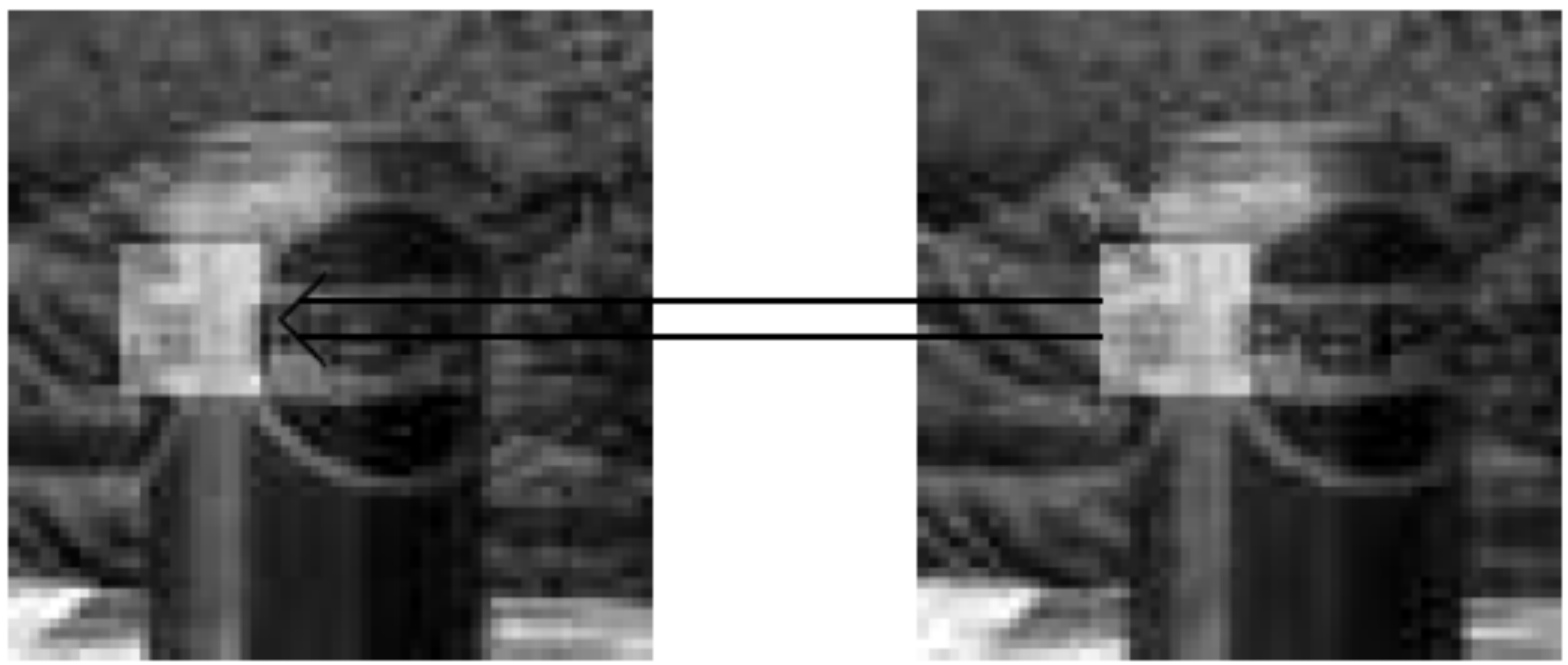
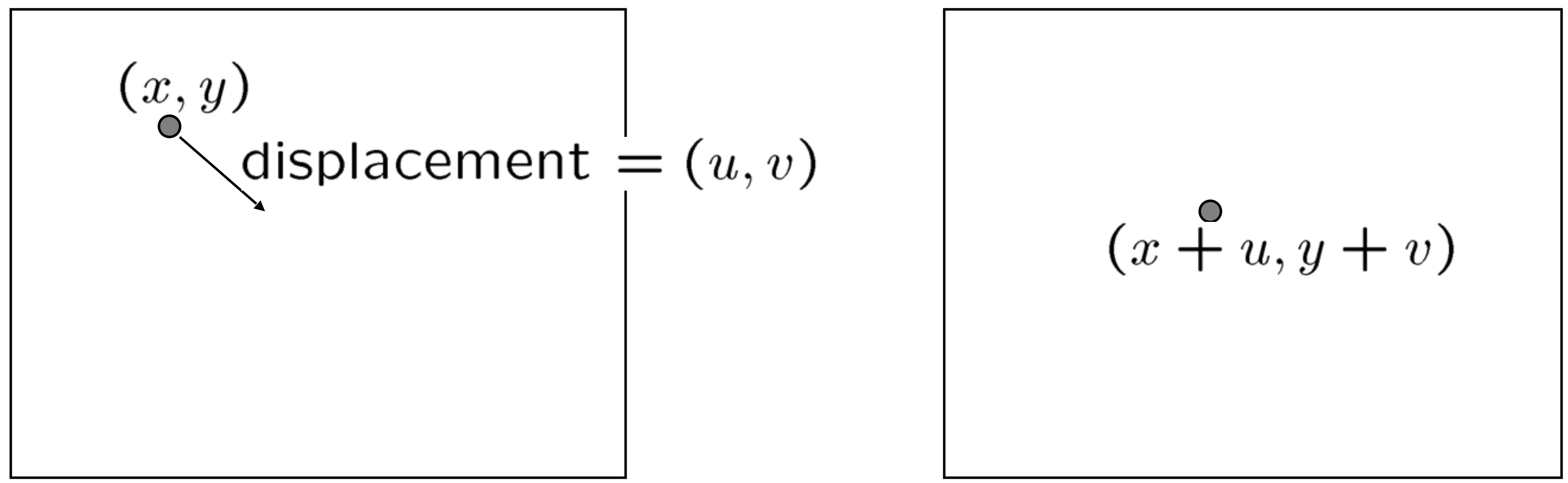


Figure 1.5: Data conservation assumption. The highlighted region in the right image looks roughly the same as the region in the left image, despite the fact that it has moved.

Figure by Michael Black

Optical flow constraints (grayscale images)



$H(x, y)$ $I(x, y)$

Let's look at these constraints more closely

- brightness constancy: Q: what's the equation?

$$H(x, y) = I(x + u, y + v)$$

- small motion:

$$I(x+u, y+v) = I(x, y) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v + \text{higher order terms}$$
$$\approx I(x, y) + \frac{\partial I}{\partial x}u + \frac{\partial I}{\partial y}v$$

Steve Seitz

Optical flow equation

Combining these two equations

shorthand: $I_x = \frac{\partial I}{\partial x}$

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

Steve Seitz

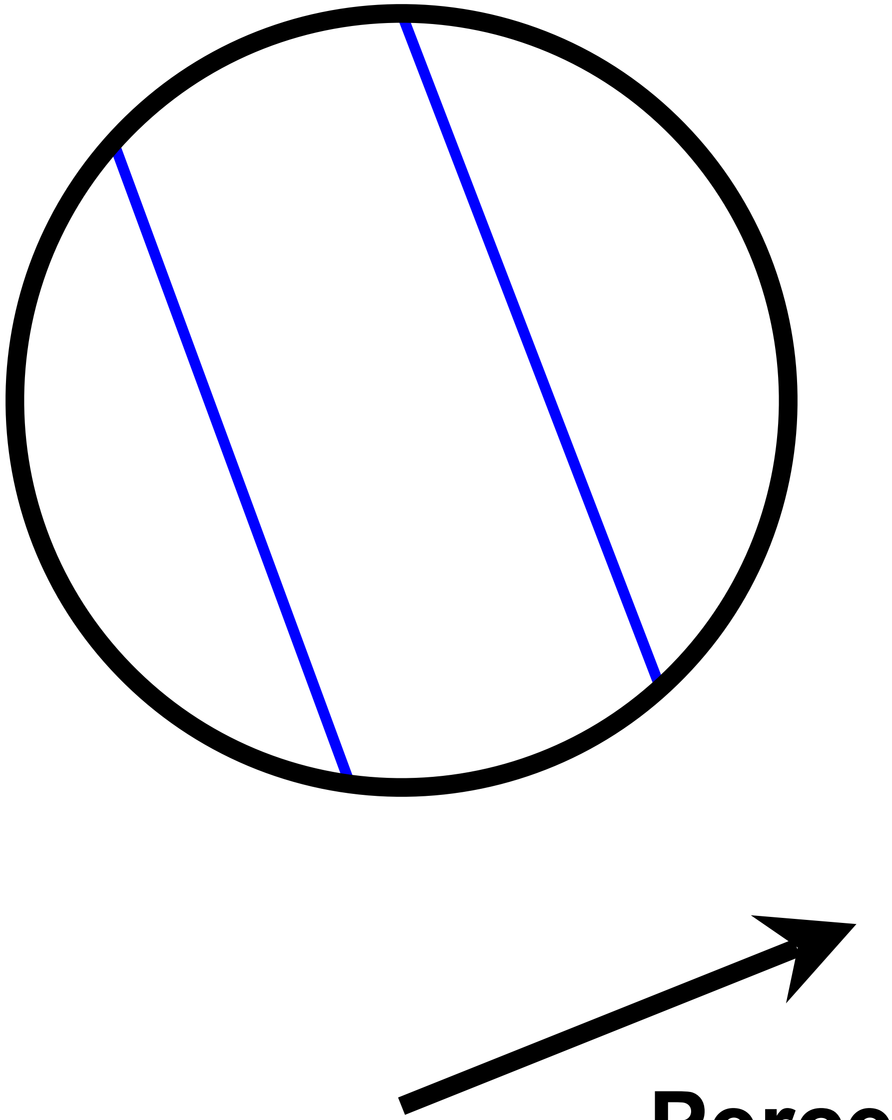
Optical flow equation

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

Q: how many unknowns and equations per pixel?

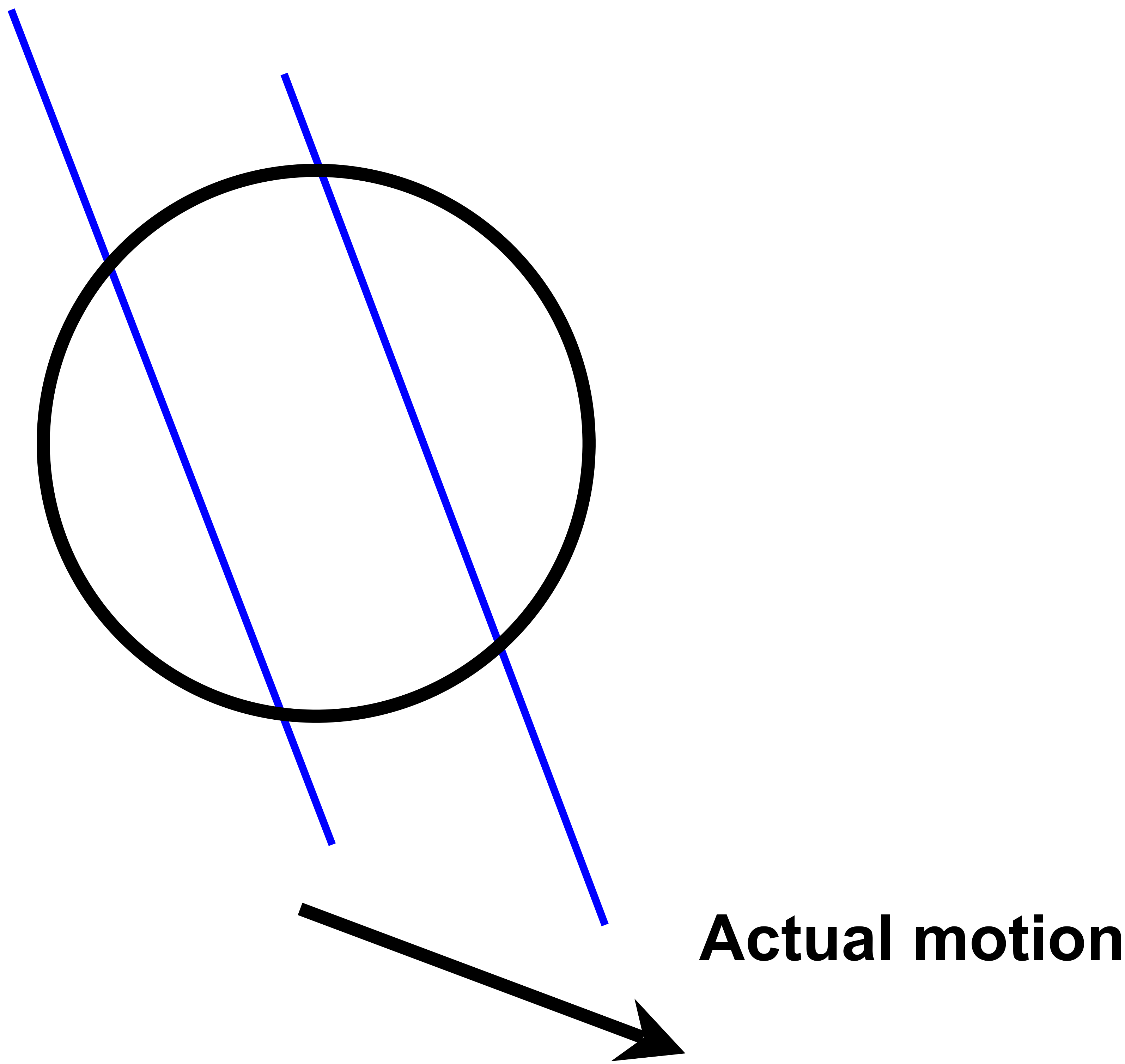
Intuitively, what does this ambiguity mean?

The aperture problem

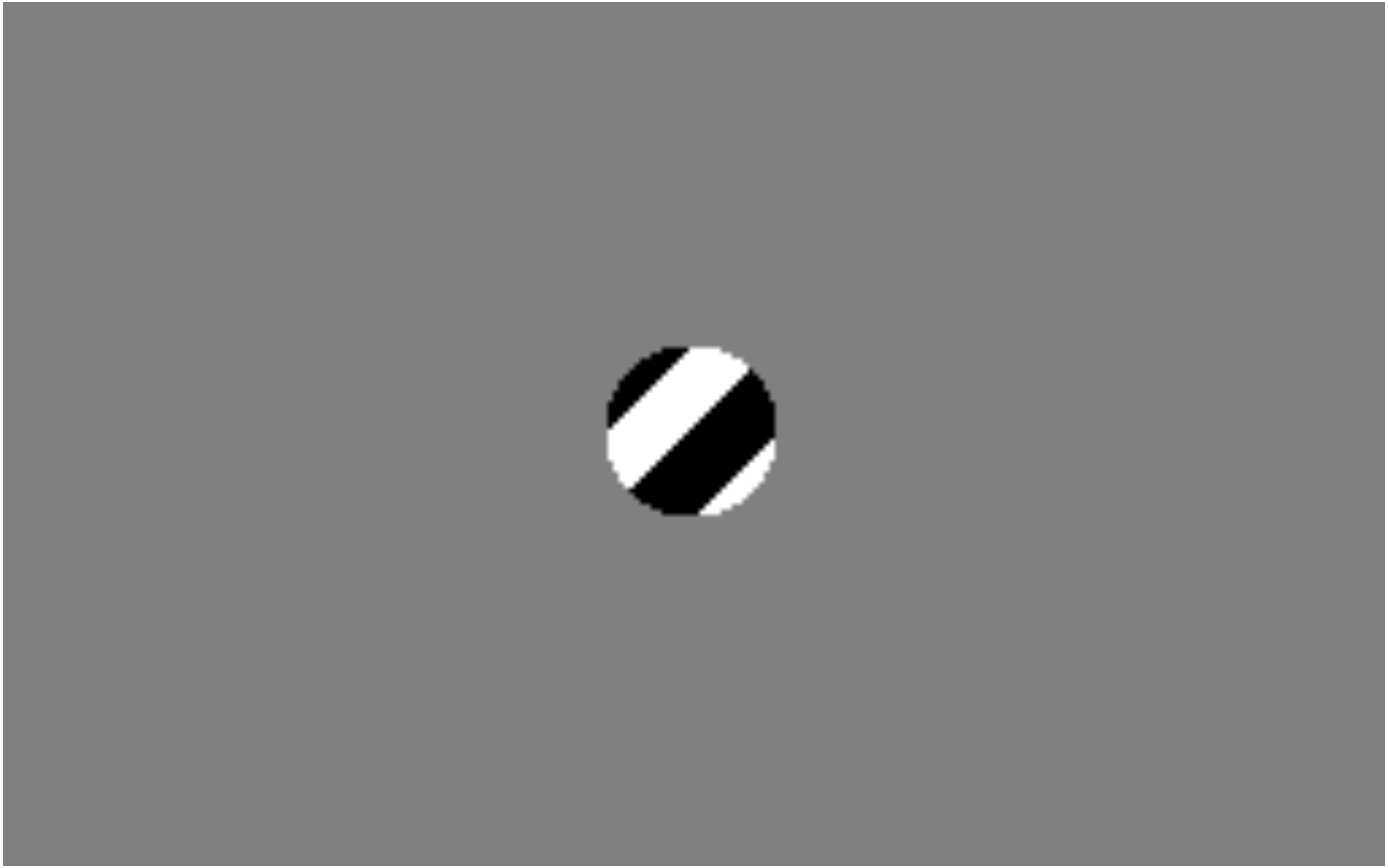


Perceived motion

The aperture problem



The barber pole illusion



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

The barber pole illusion

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

Solving the aperture problem (grayscale image)

- How to get more equations for a pixel?
- **Spatial coherence constraint:** pretend the pixel's neighbors have the same (u,v)

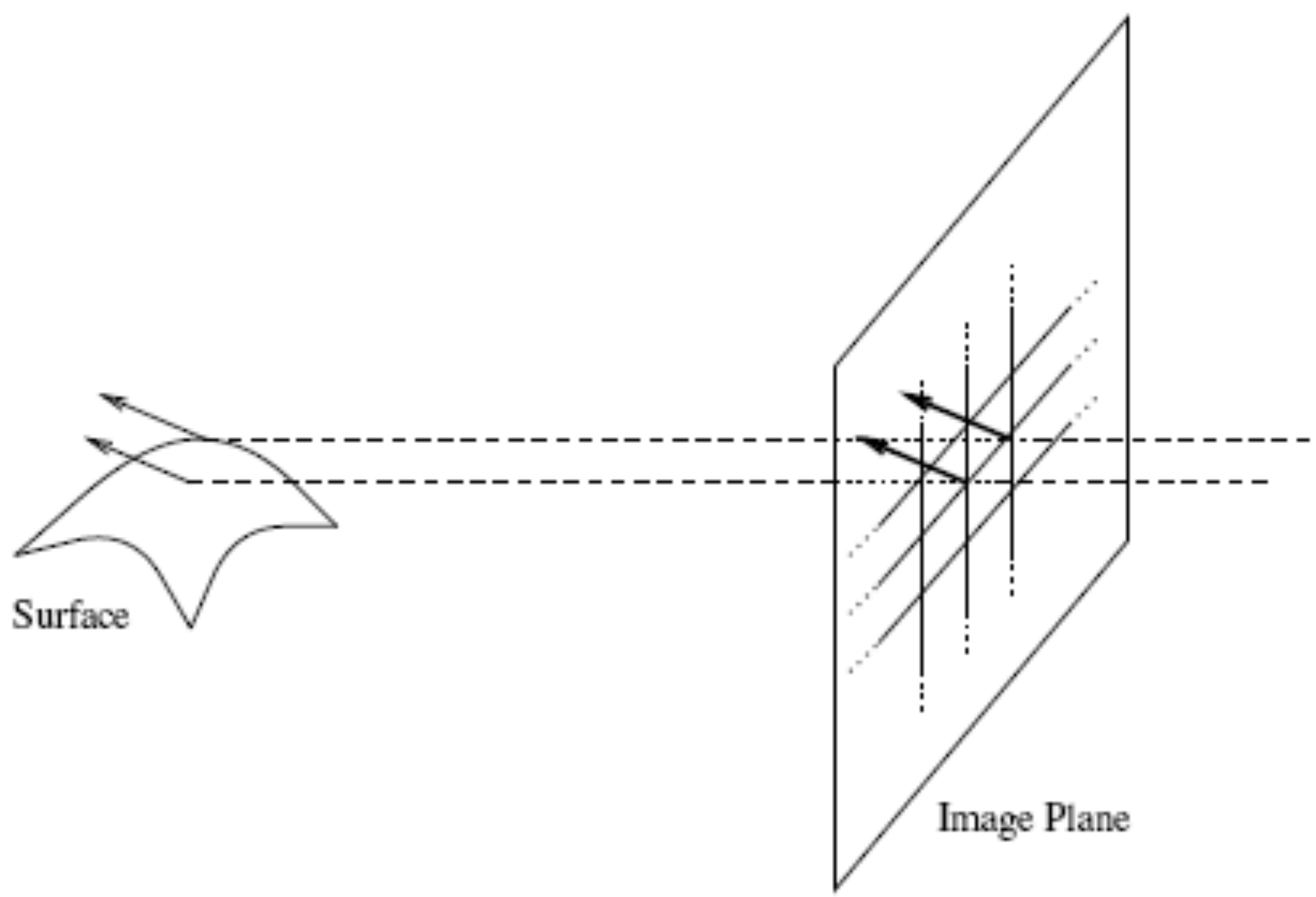


Figure 1.7: Spatial coherence assumption. Neighboring points in the image are assumed to belong to the same surface in the scene.

Figure by Michael Black

Solving the aperture problem (grayscale image)

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

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

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

$$\underset{25\text{x}2}{A} \ \ \underset{2\text{x}1}{d} = \underset{25\text{x}1}{b}$$

Steve Seitz

Solving the aperture problem

Prob: we have more equations than unknowns

$$\underset{25\text{x}2}{A}\ \underset{2\text{x}1}{d} = \underset{25\text{x}1}{b} \longrightarrow \text{minimize } \|Ad - b\|^2$$

Solution: solve least squares problem

- minimum least squares solution given by solution (in d) of:

$$\underset{2\text{x}2}{(A^T A)}\ \underset{2\text{x}1}{d} = \underset{2\text{x}1}{A^T b}$$

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

- The summations are over all pixels in the K x K window
- This technique was first proposed by Lucas & Kanade (1981)

Conditions for solvability

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

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

When is this solvable?

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

Slide by Steve Seitz, UW

Edge

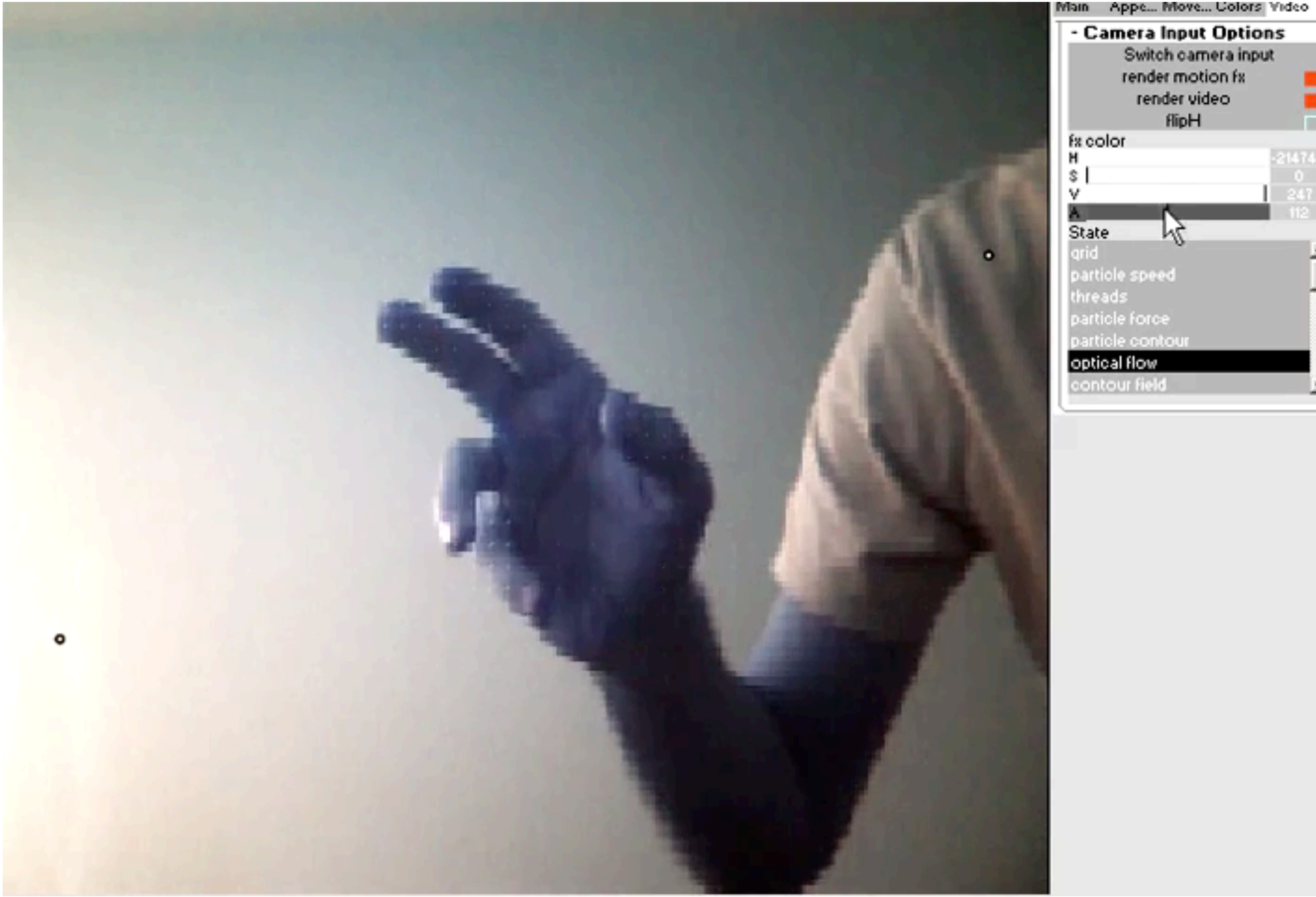
– gradients very large or very small
– large λ_1, small λ_2

Low-texture region

– gradients have small magnitude
– small λ_1, small λ_2

High-texture region

– gradients are different, large magnitudes
– large λ_1, large λ_2


Main
Appe... Move... Colors Video
- Camera Input Options
Switch camera input
render motion fx
render video
flipH
fx color
H
S
V
A
State
grid
particle speed
threads
particle force
particle contour
optical flow
contour field

Example use of optical flow: facial animation

http://www.fxguide.com/article333.html

Example use of optical flow: Motion Paint

Use optical flow to track brush strokes, in order to animate them to follow underlying scene motion.

http://www.fxguide.com/article333.html

Motion vs. Stereo: Similarities

- Both involve solving
  - Correspondence: disparities, motion vectors
  - Reconstruction

Motion vs. Stereo: Differences

- **Motion**:
  - Uses velocity: consecutive frames must be close to get good approximate time derivative
  - 3d movement between camera and scene not necessarily single 3d rigid transformation
- **Whereas with stereo**:
  - Could have any disparity value
  - View pair separated by a single 3d transformation

Coming up

Background subtraction, activity recognition, tracking

Project 2: Stereo

- **Due today!**
- **Code reviews due this Friday, 17 Oct**
- **Project 3 out in one week: 21 Oct**
- **Start thinking about final project groups and topics!**