Motion and Optical Flow

ECE/CSE 576

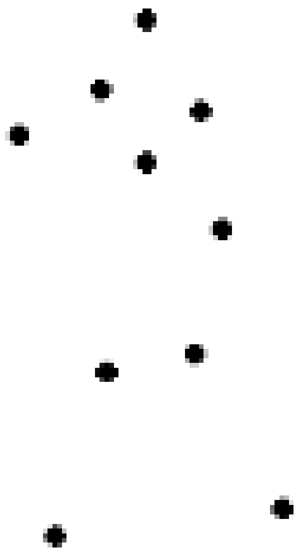
Linda Shapiro

We live in a moving world

- Perceiving, understanding and predicting motion is an important part of our daily lives

Motion and perceptual organization

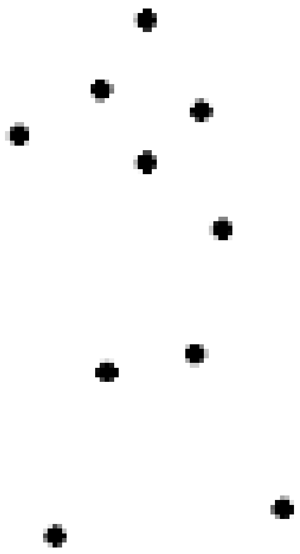
- Even “impoverished” motion data can evoke a strong percept



G. Johansson, “Visual Perception of Biological Motion and a Model For Its Analysis", *Perception and Psychophysics 14, 201-211, 1973.*

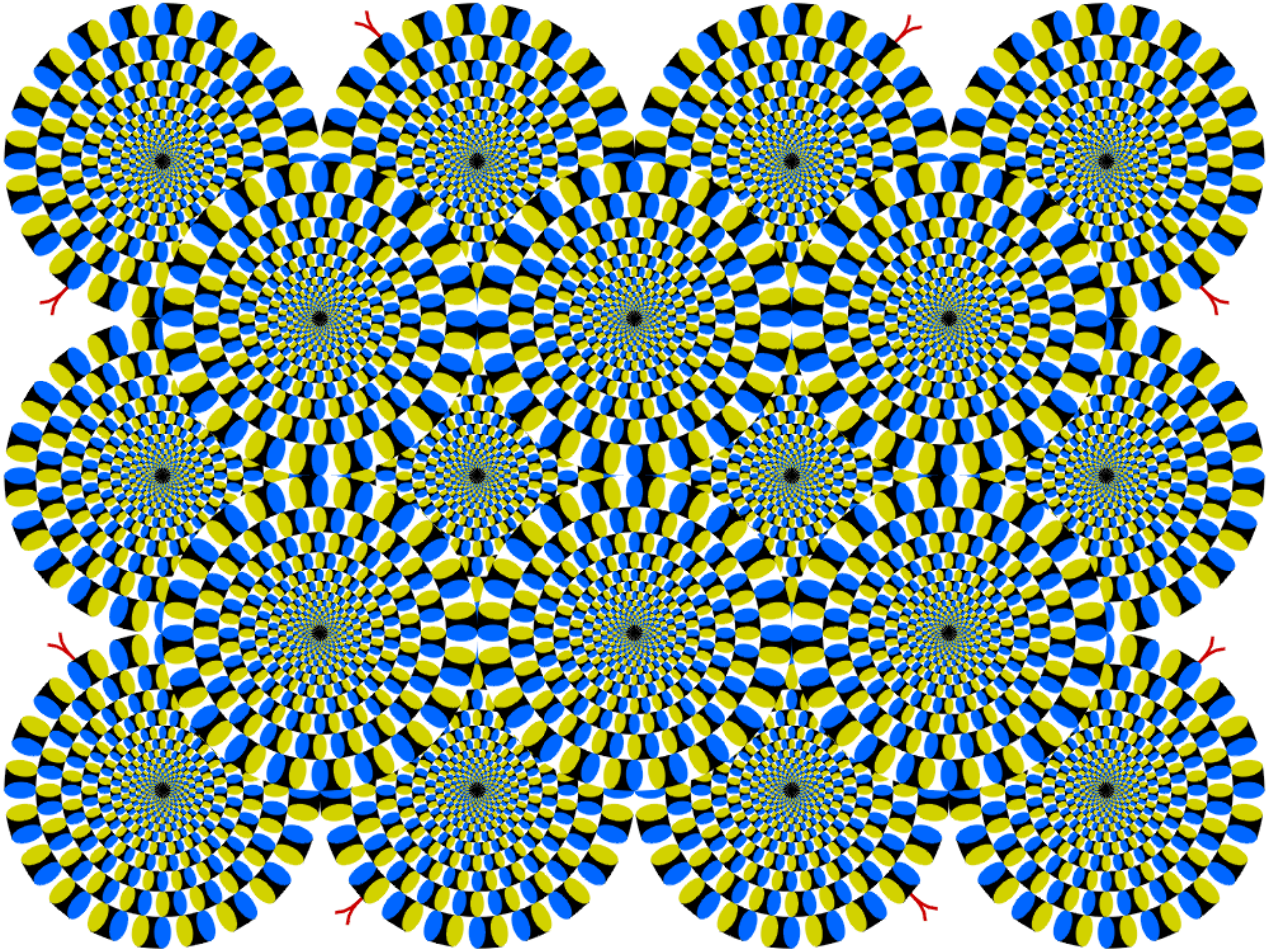
Motion and perceptual organization

- Even “impoverished” motion data can evoke a strong percept



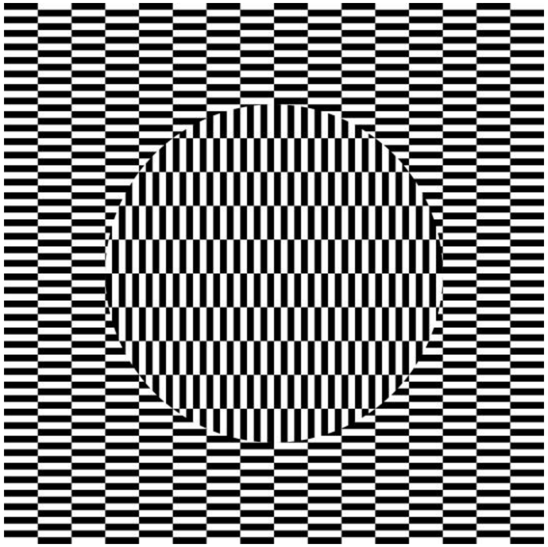
G. Johansson, “Visual Perception of Biological Motion and a Model For Its Analysis", *Perception and Psychophysics 14, 201-211, 1973.*

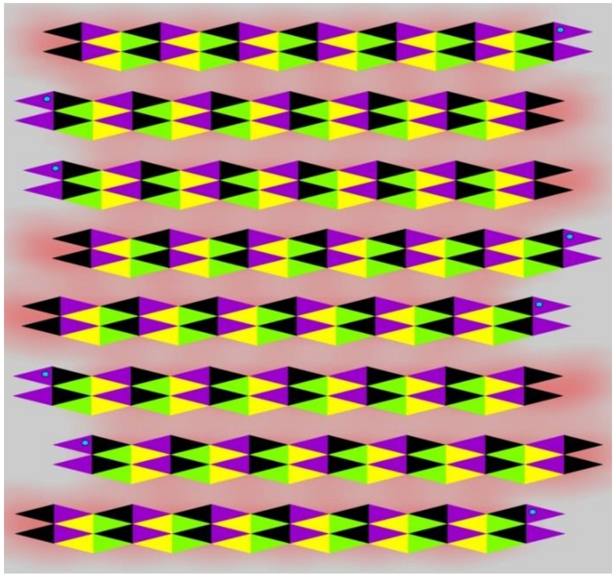
Seeing motion from a static picture?

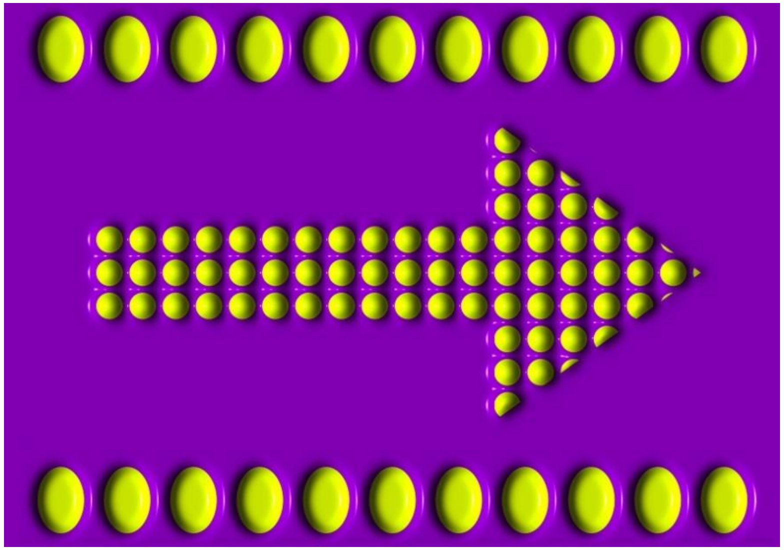


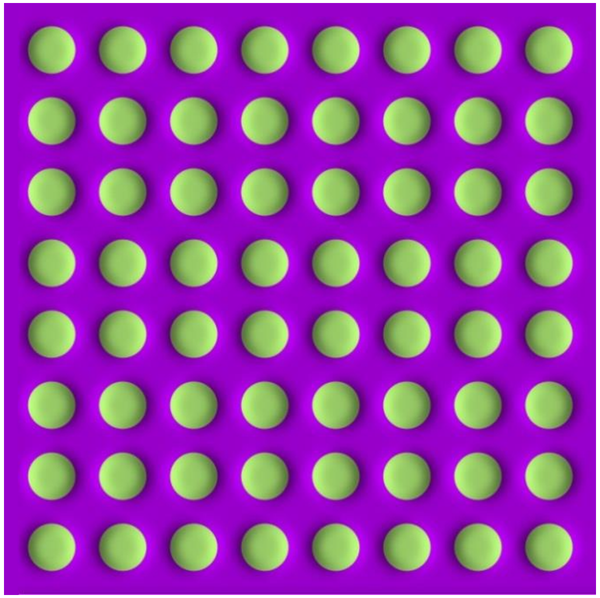
http://www.ritsumei.ac.jp/~akitaoka/index-e.html

More examples



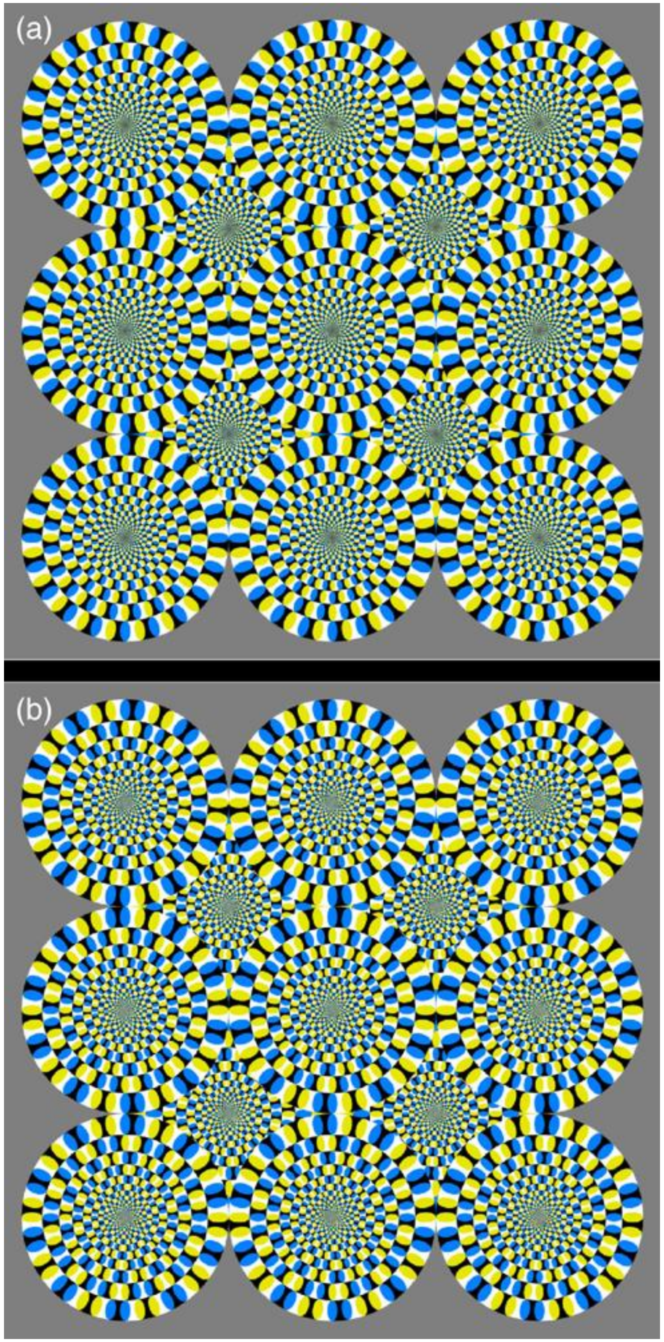






How is this possible?

- The true mechanism is yet to be revealed
- FMRI data suggest that illusion is related to some component of eye movements
- We don’t expect computer vision to “see” motion from these stimuli, yet



The cause of motion

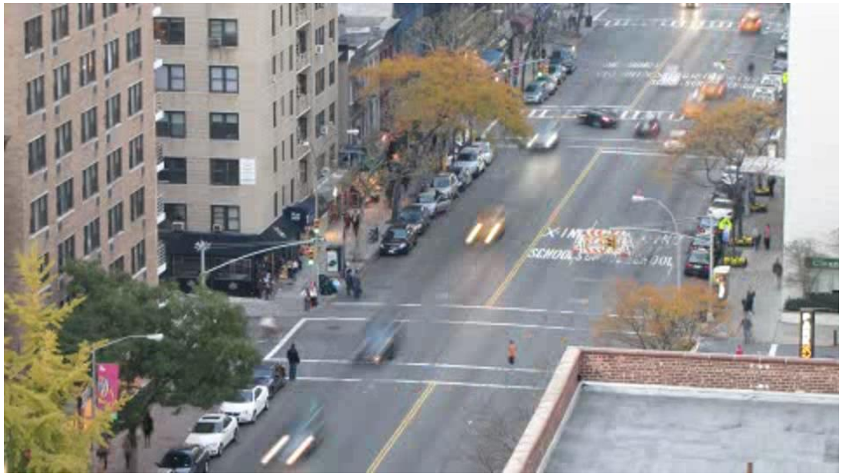
- Three factors in imaging process
  - Light
  - Object
  - Camera
- Varying either of them causes motion
  - Static camera, moving objects (surveillance)
  - Moving camera, static scene (3D capture)
  - Moving camera, moving scene (sports, movie)
  - Static camera, moving objects, moving light (time lapse)

Motion scenarios (priors)

Static camera, moving scene

Moving camera, static scene

Moving camera, moving scene



Static camera, moving scene, moving light

We still don’t touch these areas

How can we recover motion?

Recovering motion

- Feature-tracking
  - Extract visual features (corners, textured areas) and "track" them over multiple frames

- Optical flow
  - Recover image motion at each pixel from spatio-temporal image brightness variations (optical flow)

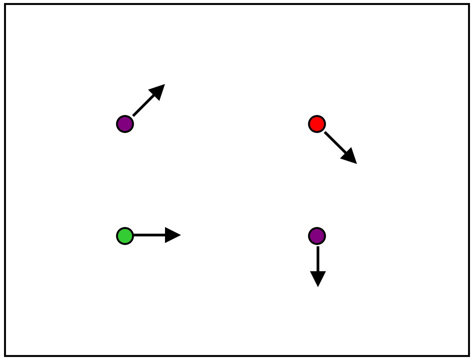
Two problems, one registration method

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.

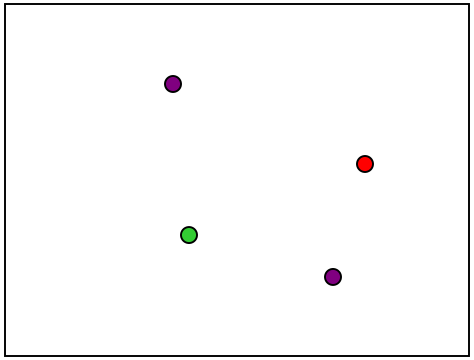
Feature tracking

- Challenges
  - Figure out which features can be tracked
  - Efficiently track across frames
  - Some points may change appearance over time (e.g., due to rotation, moving into shadows, etc.)
  - Drift: small errors can accumulate as appearance model is updated
  - Points may appear or disappear: need to be able to add/delete tracked points

Feature tracking



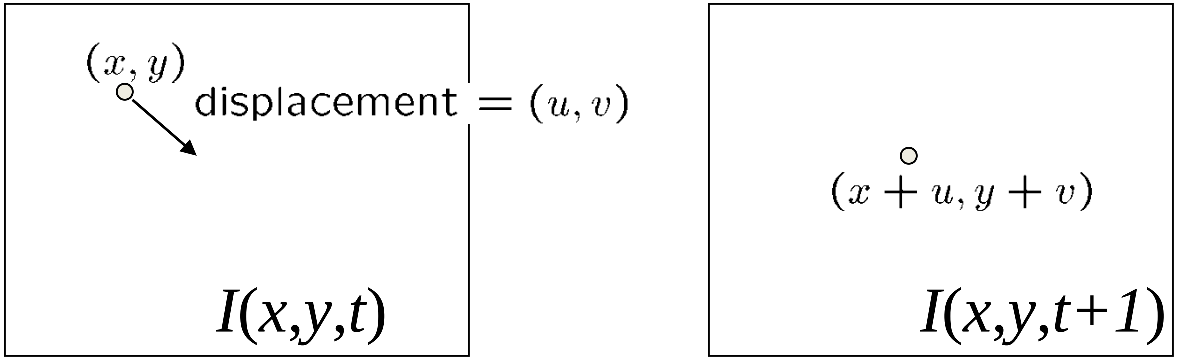
$I(x,y,t)$



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

- Given two subsequent frames, estimate the point translation
- Key assumptions of Lucas-Kanade Tracker
  - **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

The brightness constancy constraint



- Brightness Constancy Equation:

$$\boxed{I(x,y,t) = I(x+u,y+v,t+1)}$$

Take Taylor expansion of *I(x+u, y+v, t+1)* at *(x,y,t)* to linearize the right side:

Image derivative along x   Difference over frames

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

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

So: $I_x \cdot u + I_y \cdot v + I_t \approx 0$ $\quad \rightarrow \nabla \mathrm{I} \cdot [\mathrm{u} \;\; \mathrm{v}]^{\mathrm{T}} + \mathrm{I}_{\mathrm{t}} = 0$

The brightness constancy constraint

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

$$\nabla \mathrm{I} \cdot [\mathrm{u} \;\; \mathrm{v}]^{\mathrm{T}} + \mathrm{I}_{\mathrm{t}} = 0$$

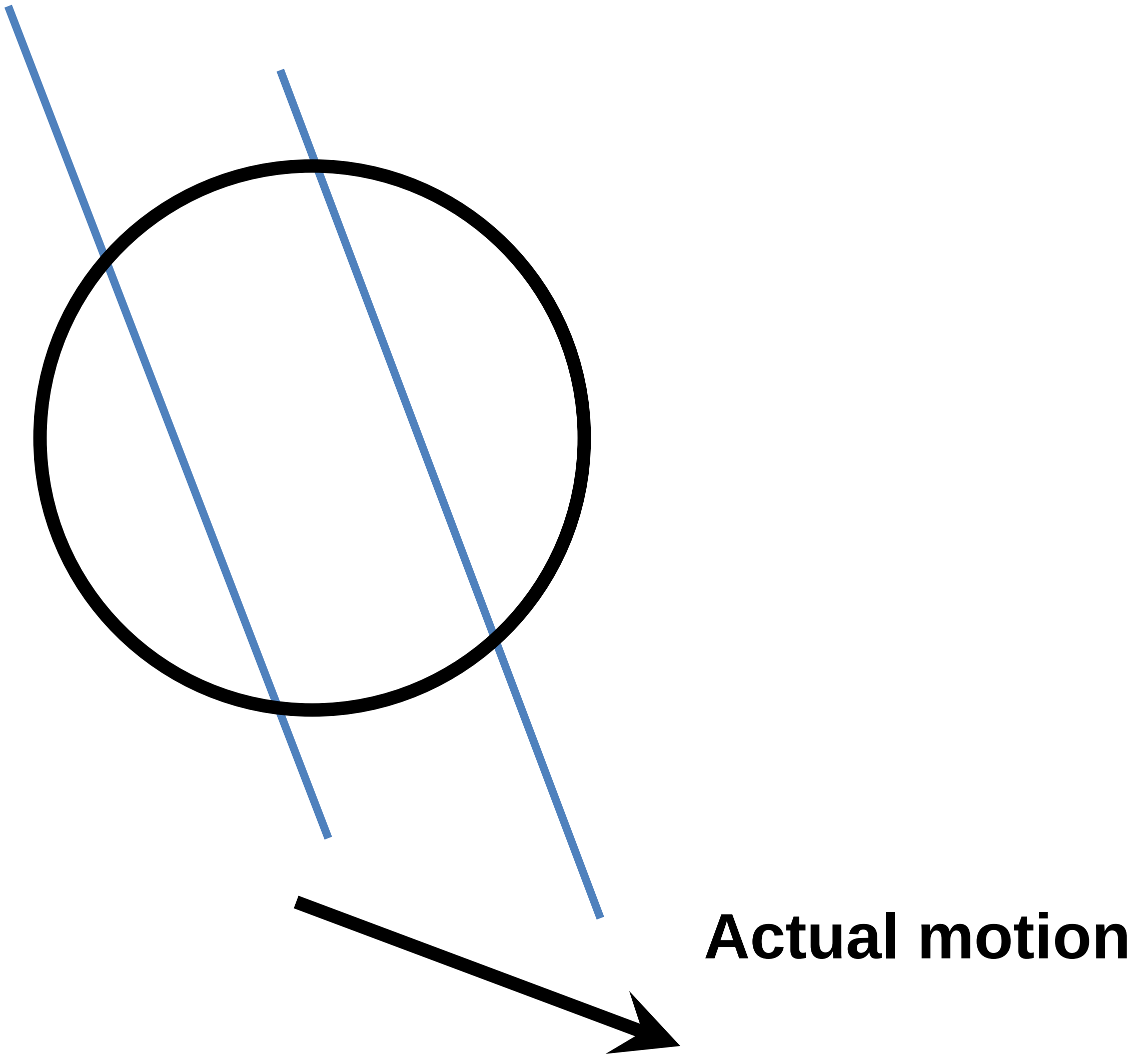
- How many equations and unknowns per pixel?
  - One equation (this is a scalar equation!), two unknowns (u,v)

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

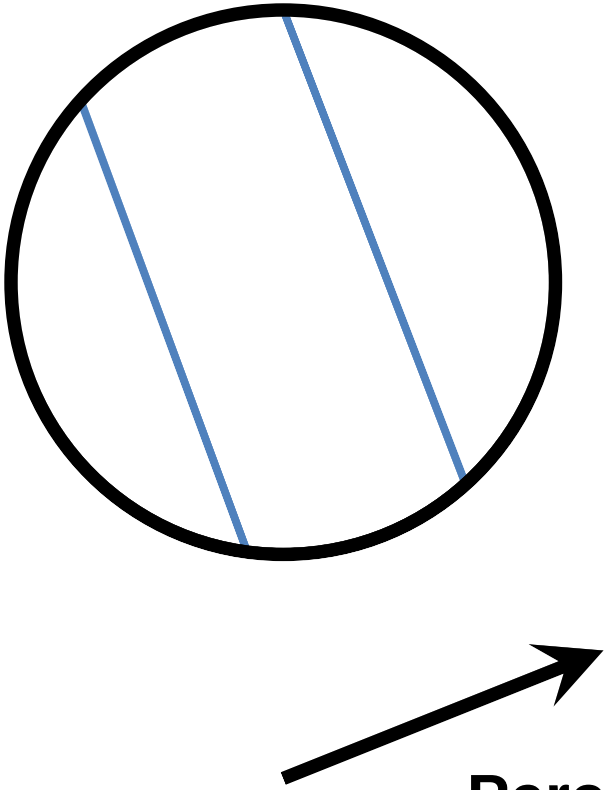
If (*u*, *v*) satisfies the equation,
so does (*u+u'*, *v+v'*) if

$$\nabla \mathrm{I} \cdot [\mathrm{u'} \;\; \mathrm{v'}]^{\mathrm{T}} = 0$$

The aperture problem

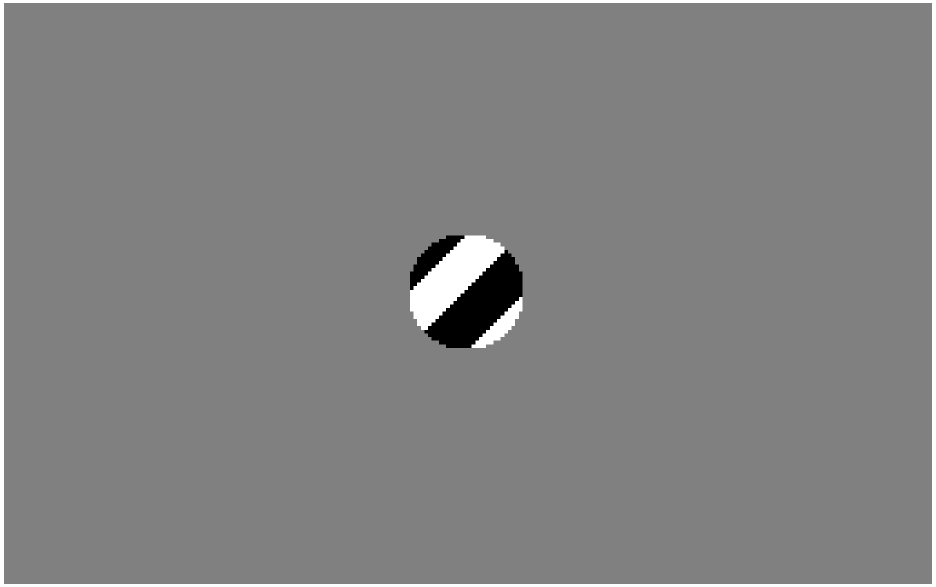


The aperture problem



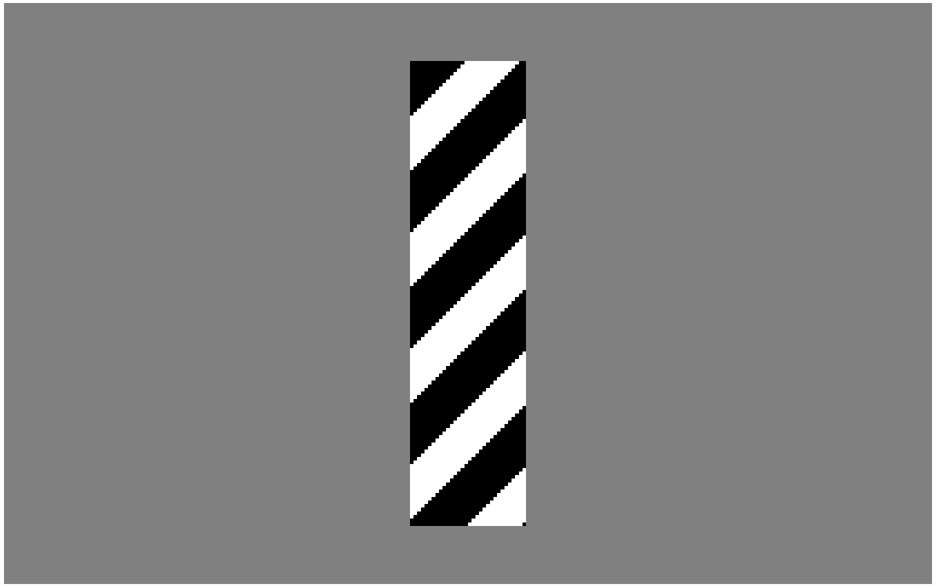
Perceived motion

The barber pole illusion



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

The barber pole illusion



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

Solving the ambiguity…

B. Lucas and T. Kanade. An iterative image registration technique with an application to stereo vision. In *Proceedings of th* *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

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

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

Solving the ambiguity…

- Least squares problem:

$$\begin{bmatrix} I_x(\mathrm{p}_1) & I_y(\mathrm{p}_1) \\ I_x(\mathrm{p}_2) & I_y(\mathrm{p}_2) \\ \vdots & \vdots \\ I_x(\mathrm{p}_{25}) & I_y(\mathrm{p}_{25}) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(\mathrm{p}_1) \\ I_t(\mathrm{p}_2) \\ \vdots \\ I_t(\mathrm{p}_{25}) \end{bmatrix} \qquad \underset{\text{25x2}}{A} \; \underset{\text{2x1}}{d} = \underset{\text{25x1}}{b}$$

Matching patches across images

- Overconstrained linear system

$$\begin{bmatrix} I_x(\mathrm{p}_1) & I_y(\mathrm{p}_1) \\ I_x(\mathrm{p}_2) & I_y(\mathrm{p}_2) \\ \vdots & \vdots \\ I_x(\mathrm{p}_{25}) & I_y(\mathrm{p}_{25}) \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} I_t(\mathrm{p}_1) \\ I_t(\mathrm{p}_2) \\ \vdots \\ I_t(\mathrm{p}_{25}) \end{bmatrix} \qquad \underset{25\text{x}2}{A} \ \underset{2\text{x}1}{d} = \underset{25\text{x}1}{b}$$

Least squares solution for *d* given by

$$\boxed{(A^T A)\ d = A^T b}$$

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

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

Conditions for solvability

Optimal (u, v) satisfies Lucas-Kanade equation

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

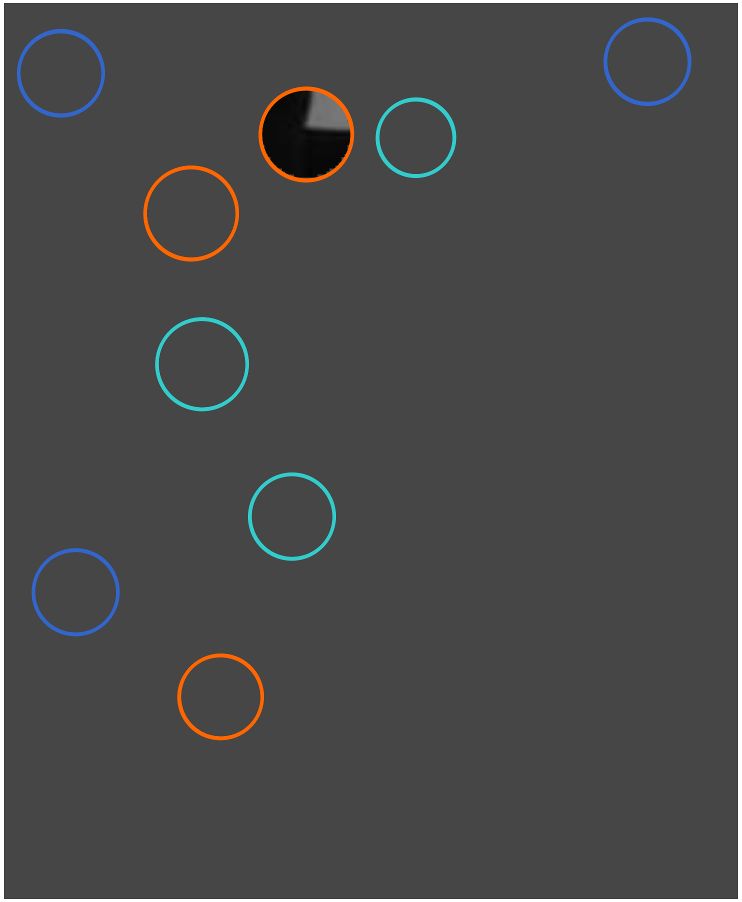
When is this solvable? I.e., what are good points to track?

- **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 you of anything?

Criteria for Harris corner detector

Aperture problem

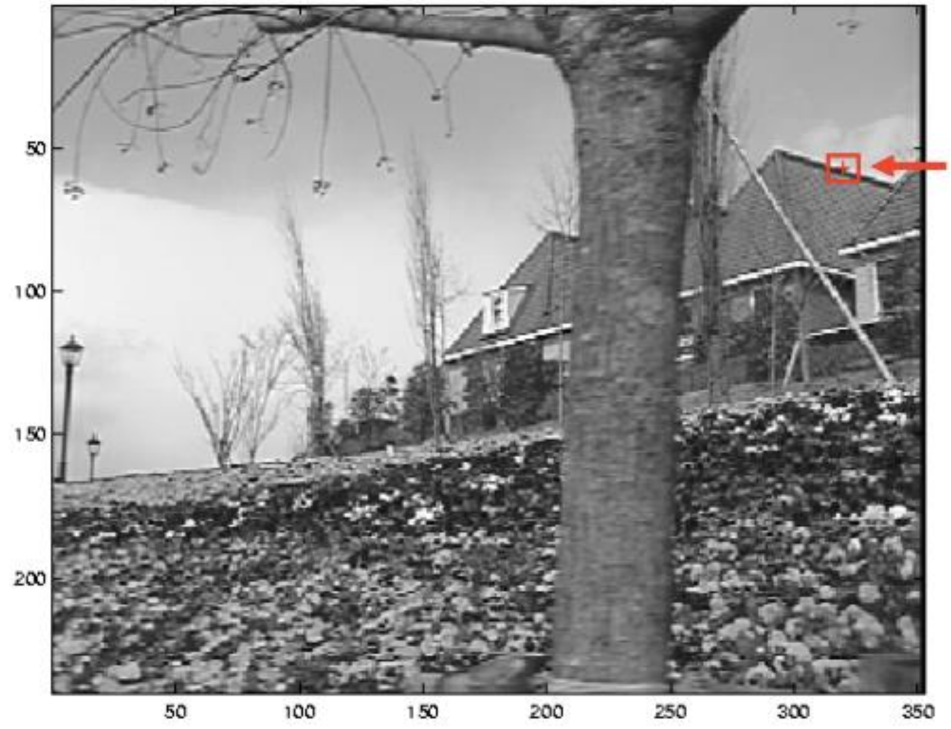


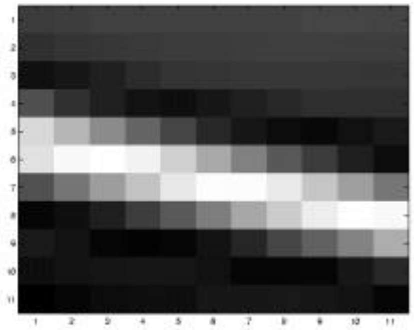
Corners

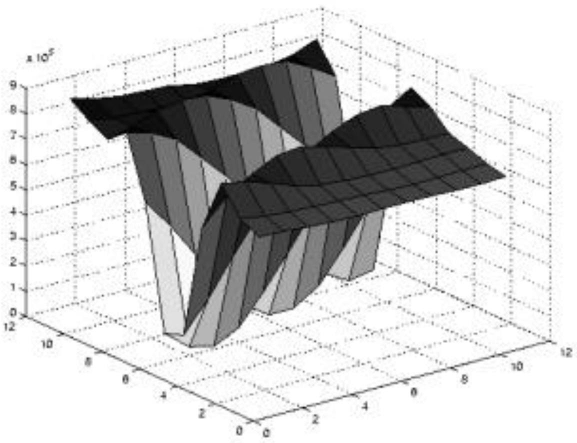
Lines

Flat regions

Edge



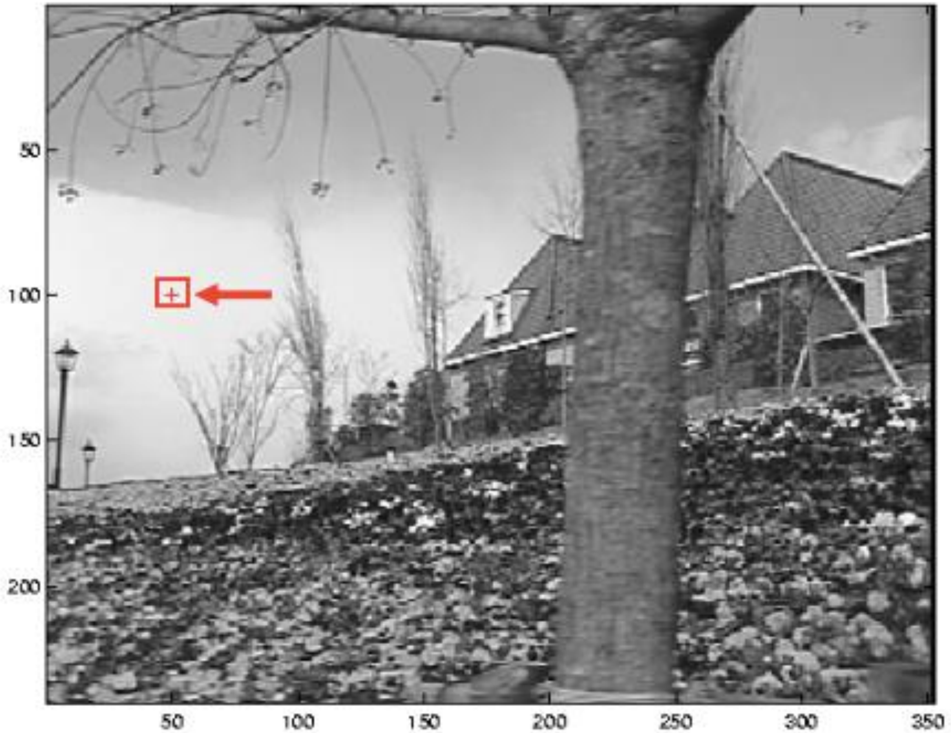


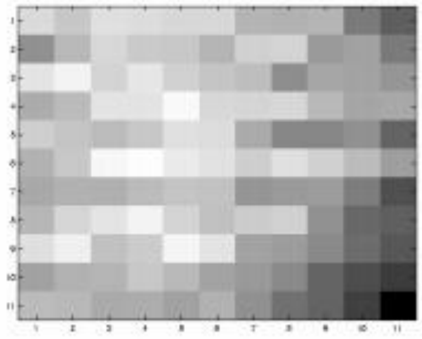


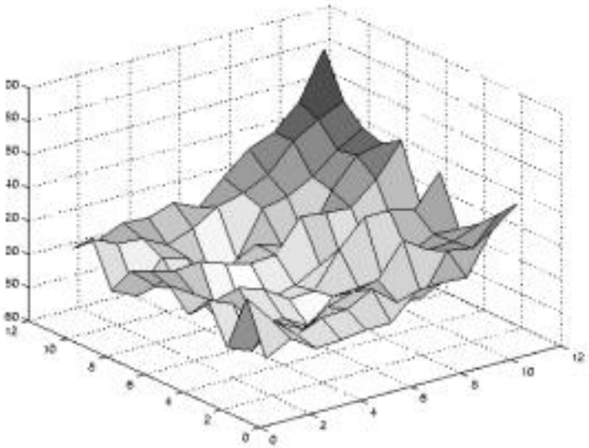
$$\sum \nabla I (\nabla I)^T$$

- large gradients, all the same
- large λ_1, small λ_2

Low Texture Region





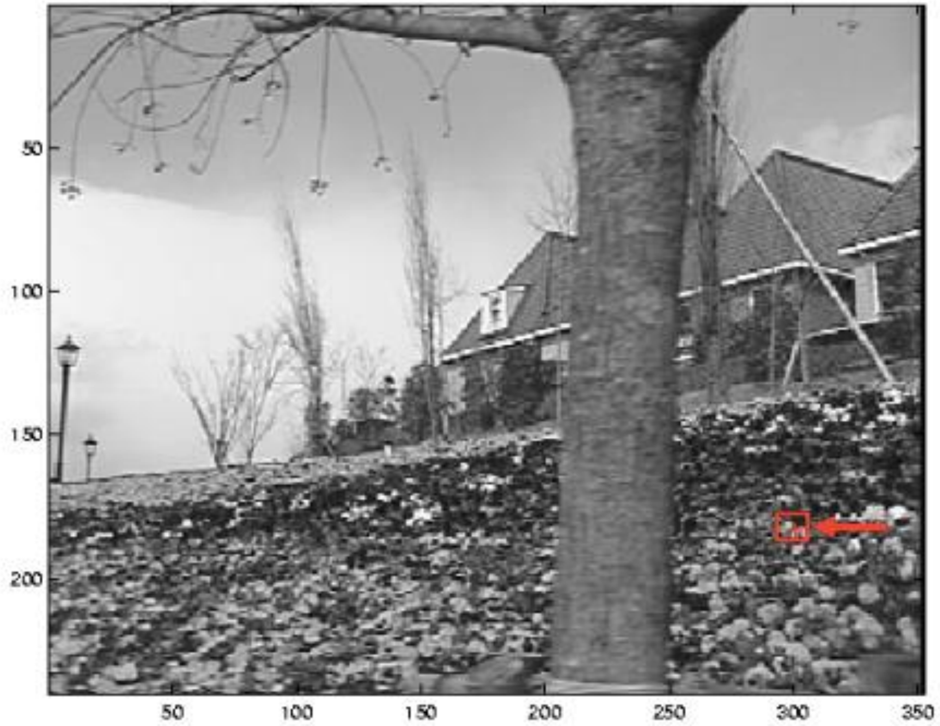


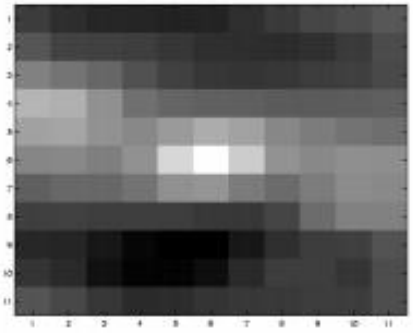
$$\sum \nabla I (\nabla I)^T$$

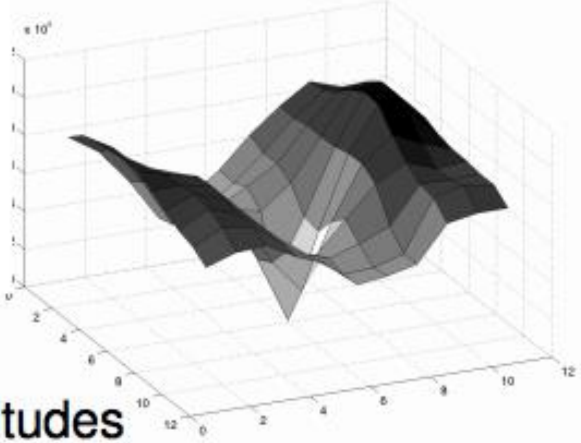
– gradients have small magnitude

– small λ_1, small λ_2

High Texture Region







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

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

Errors in Lukas-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

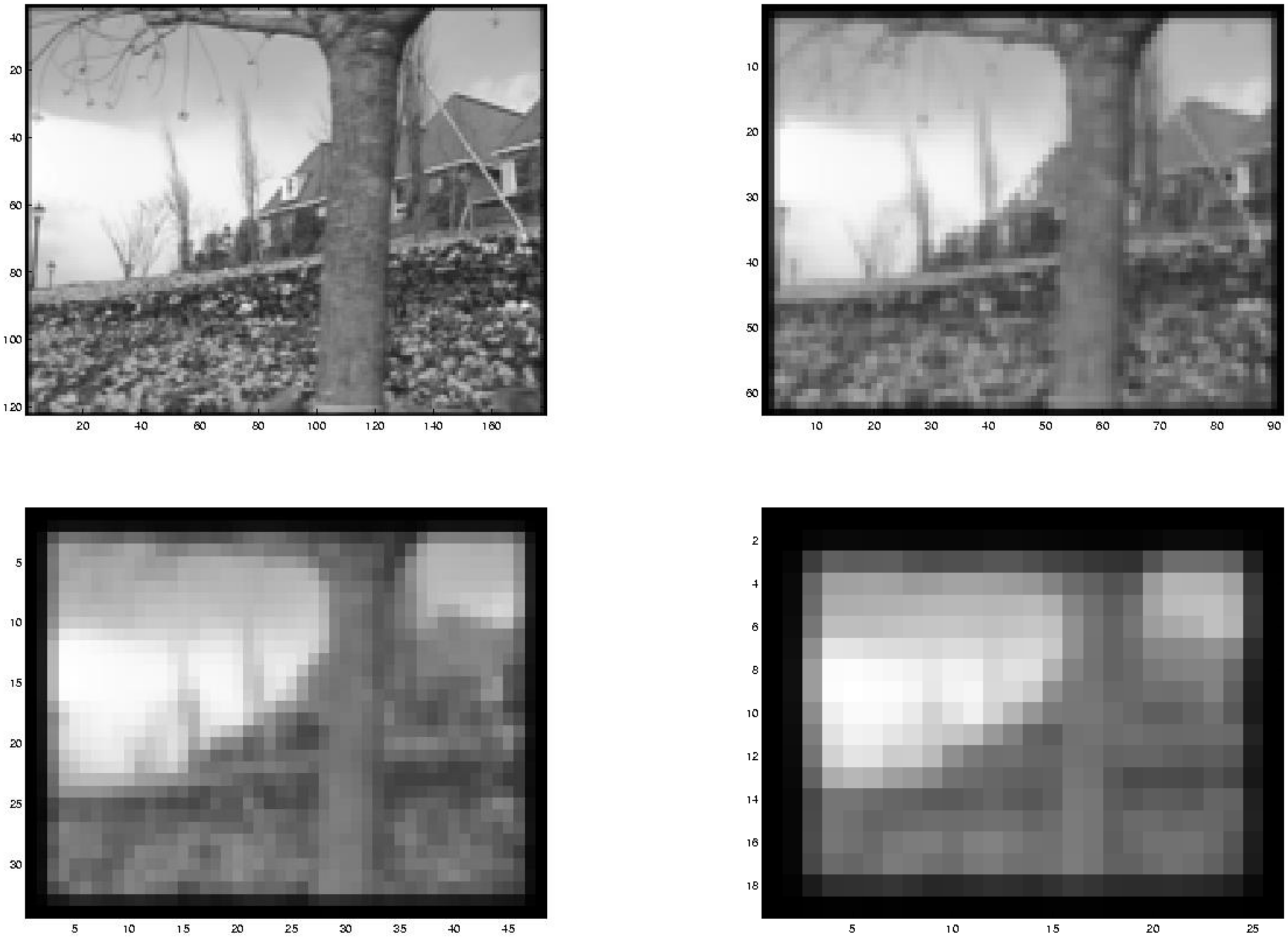
When our assumptions are violated

- Brightness constancy is **not** satisfied
- The motion is **not** small
- A point does **not** move like its neighbors
  - window size is too large
  - what is the ideal window size?

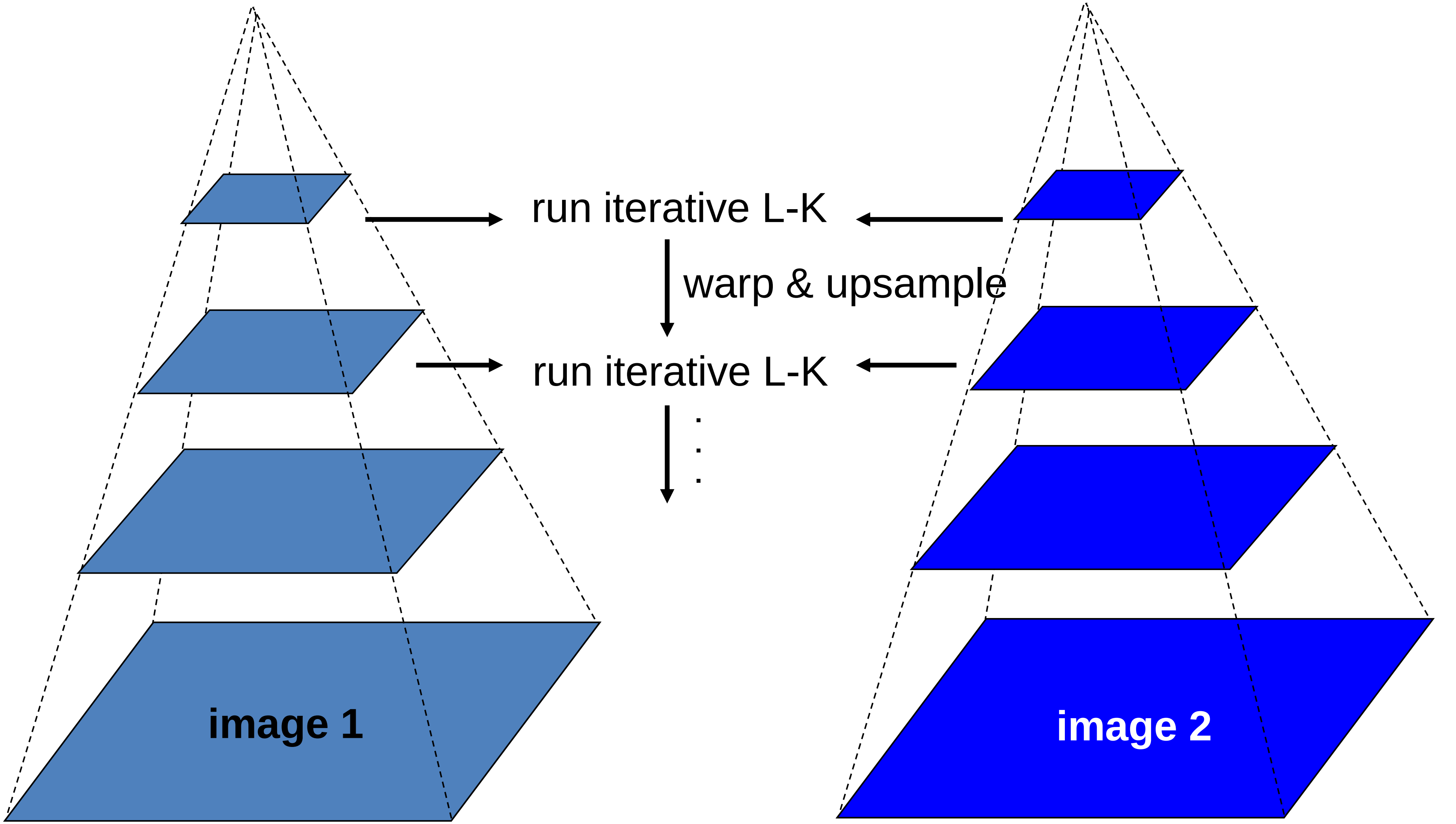
Revisiting the small motion assumption

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

Reduce the resolution!



Coarse-to-fine optical flow estimation



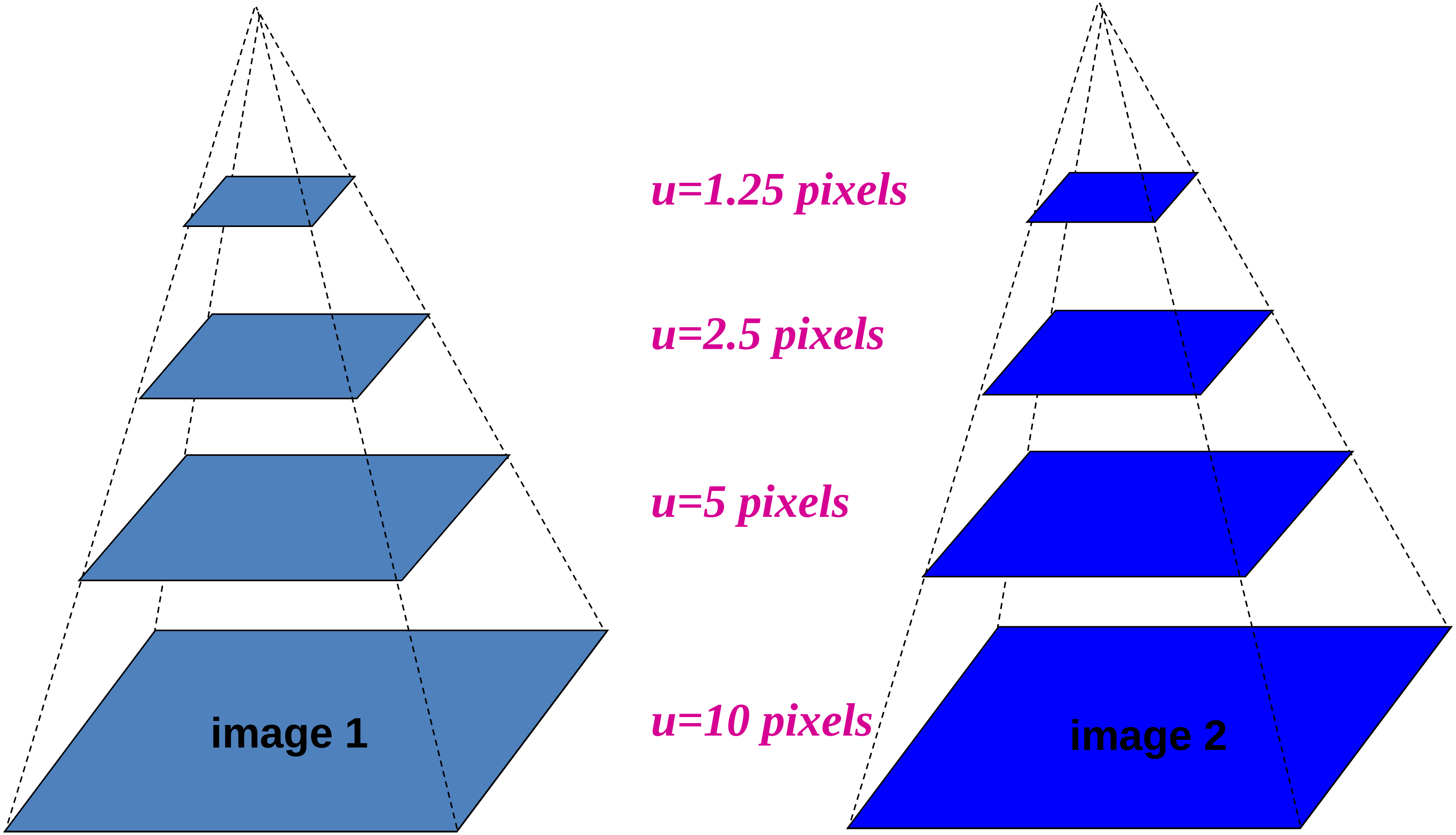
Gaussian pyramid of image 1 (t)

Gaussian pyramid of image 2 (t+1)

A Few Details

- Top Level
  - Apply L-K to get a flow field representing the flow from the first frame to the second frame.
  - Apply this flow field to warp the first frame toward the second frame.
  - Rerun L-K on the new warped image to get a flow field from it to the second frame.
  - Repeat till convergence.
- Next Level
  - Upsample the flow field to the next level as the first guess of the flow at that level.
  - Apply this flow field to warp the first frame toward the second frame.
  - Rerun L-K and warping till convergence as above.
- Etc.

Coarse-to-fine optical flow estimation

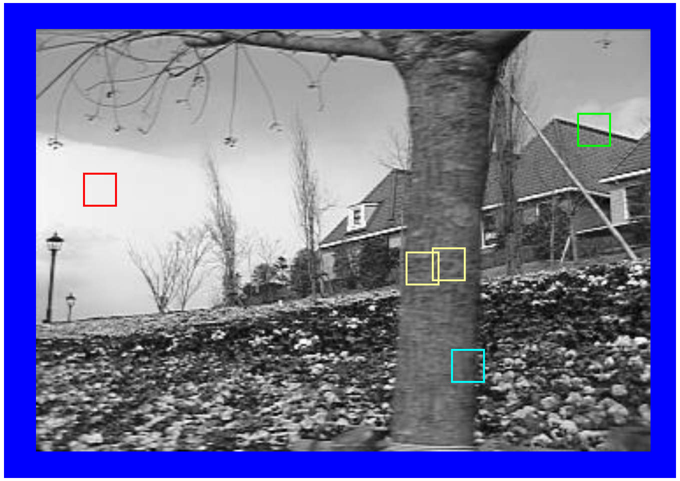


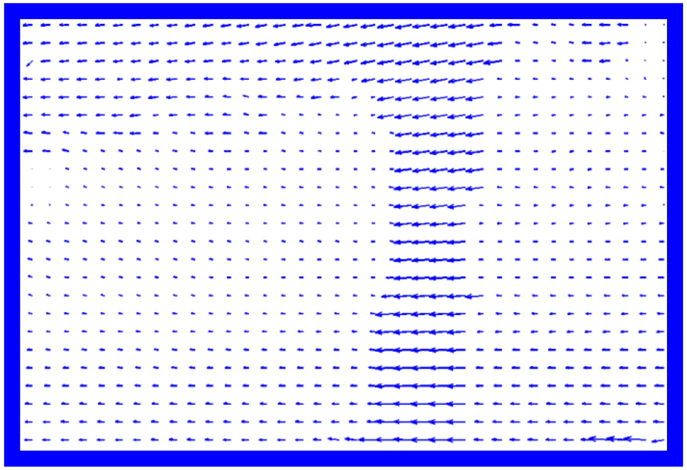
Gaussian pyramid of image 1

Gaussian pyramid of image 2

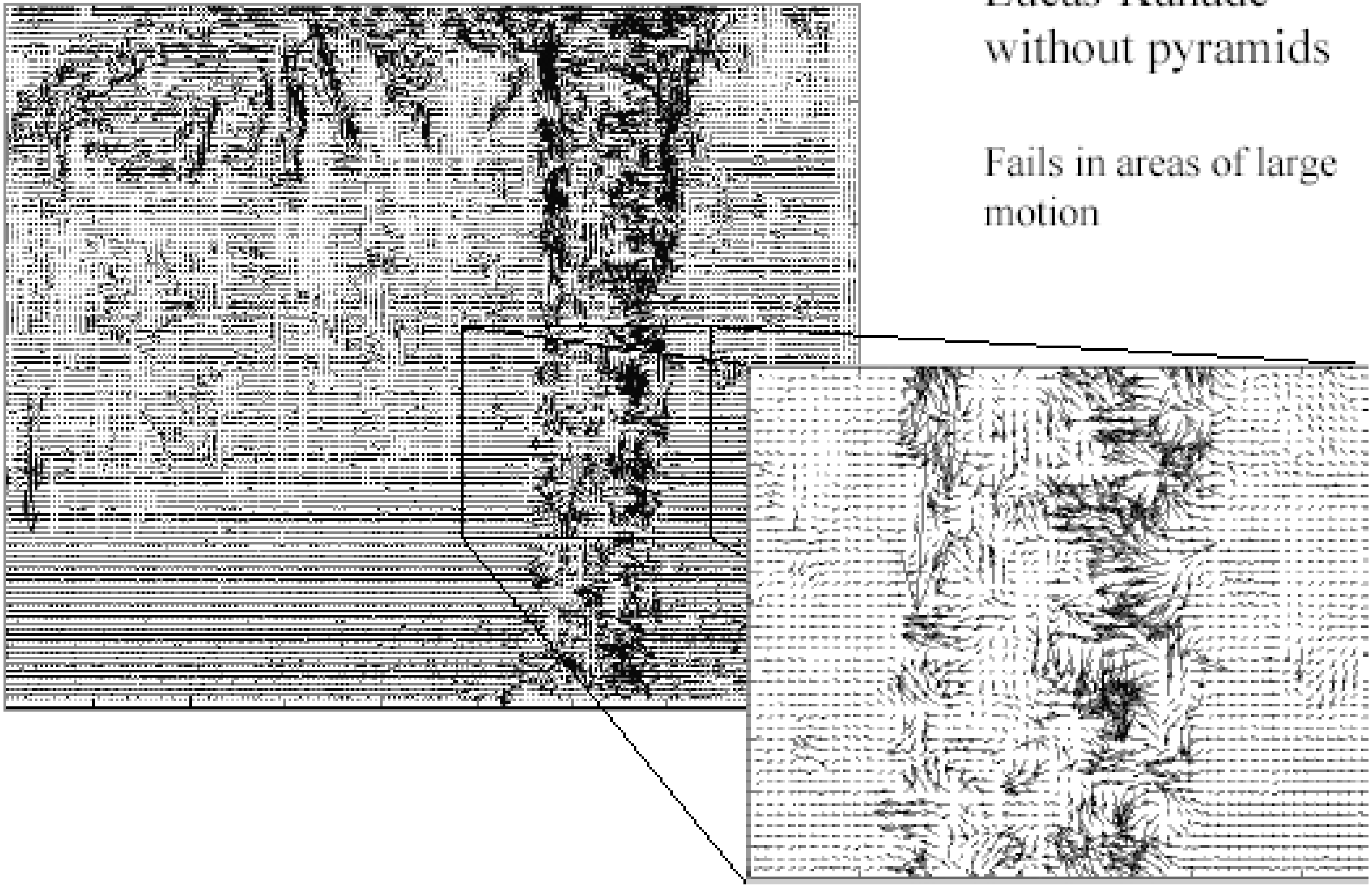
The Flower Garden Video

What should the optical flow be?





Optical Flow Results

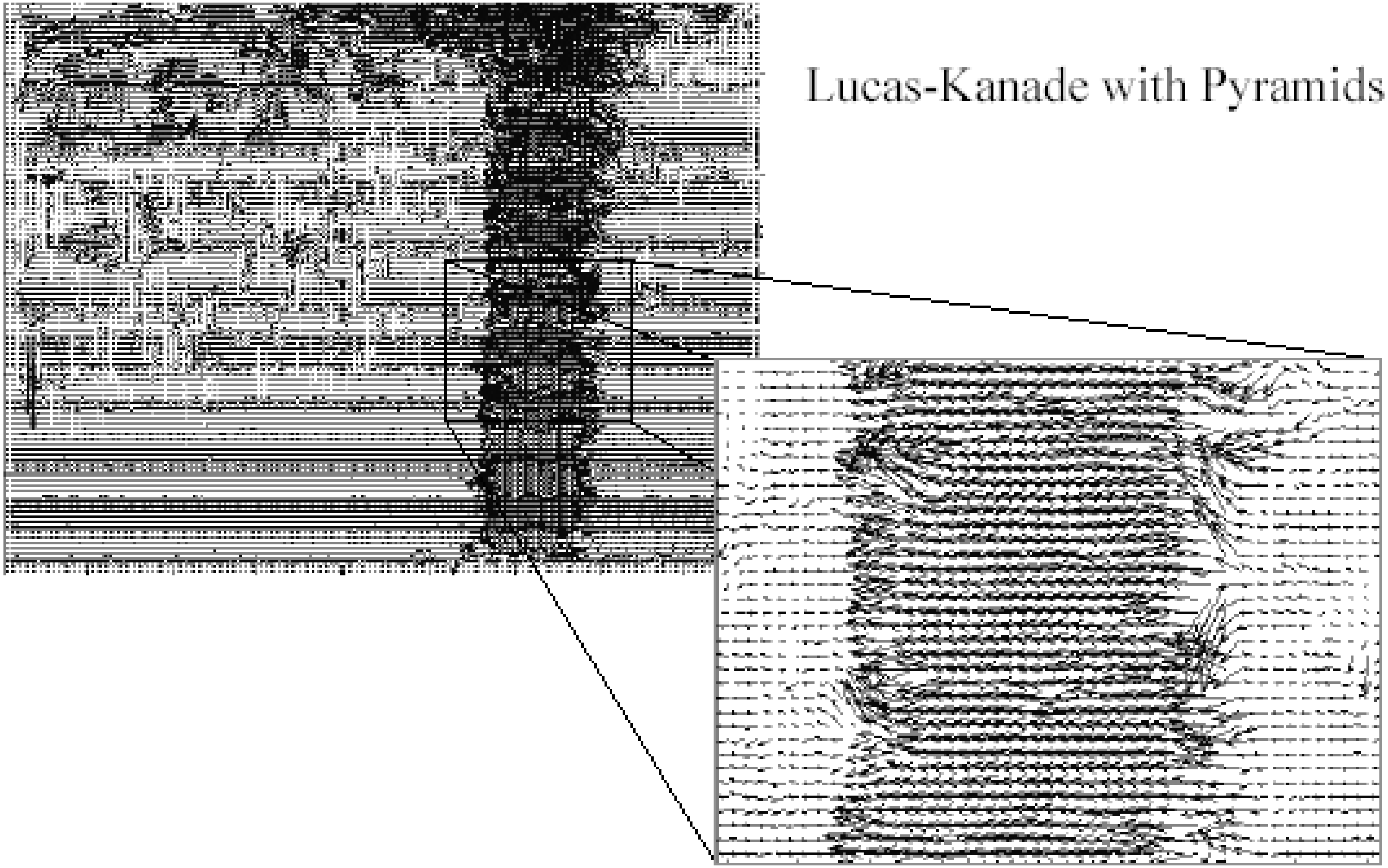


Lucas-Kanade
without pyramids

Fails in areas of large motion

* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

Optical Flow Results



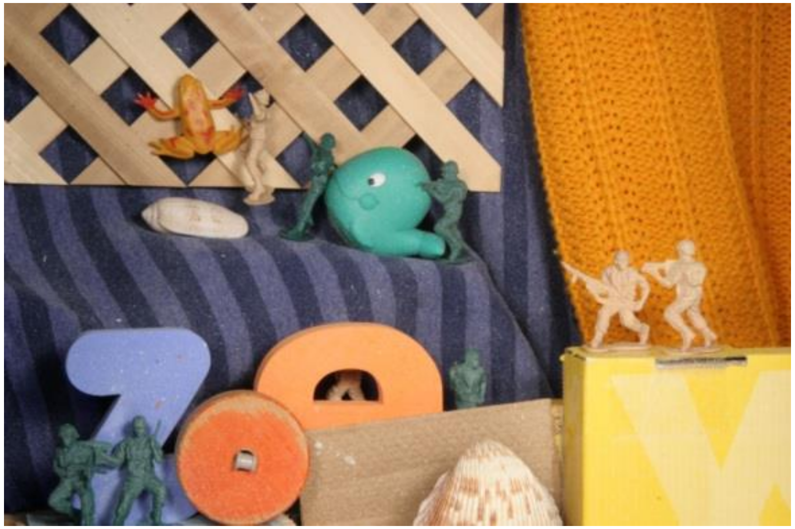
* From Khurram Hassan-Shafique CAP5415 Computer Vision 2003

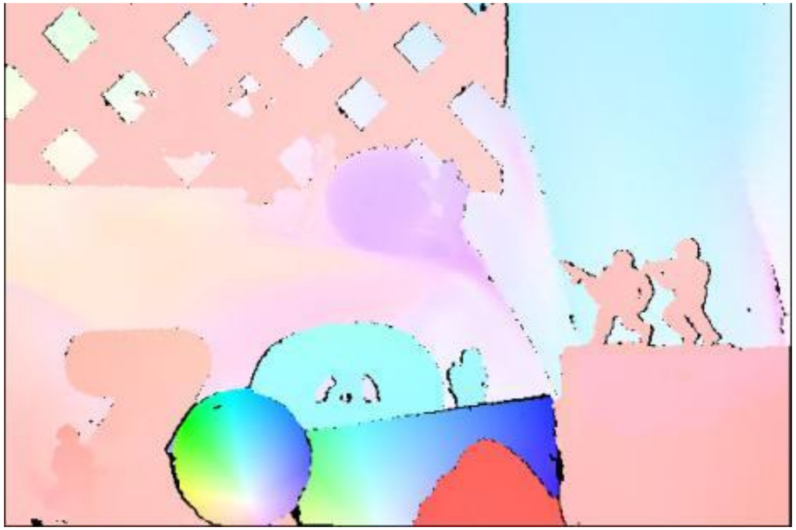
Flow quality evaluation

Flow quality evaluation

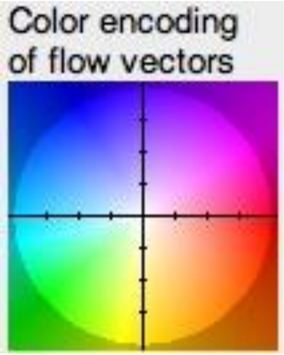
Flow quality evaluation

- Middlebury flow page
  - http://vision.middlebury.edu/flow/



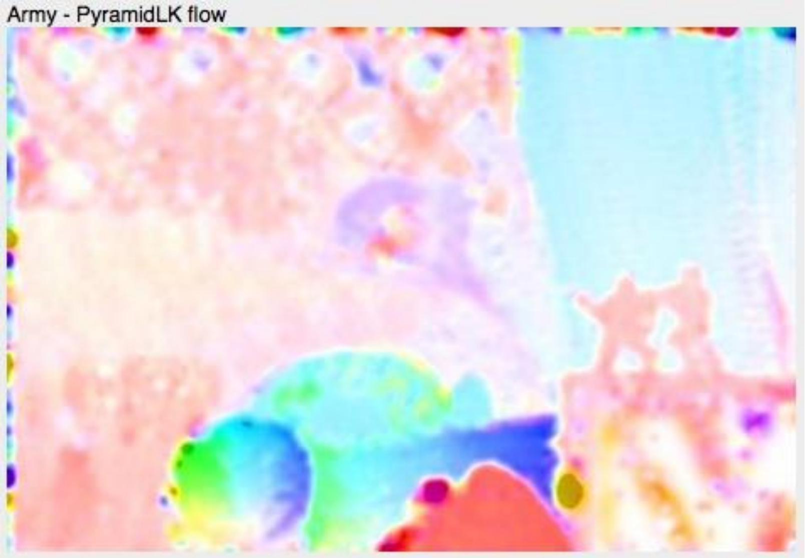


Ground Truth

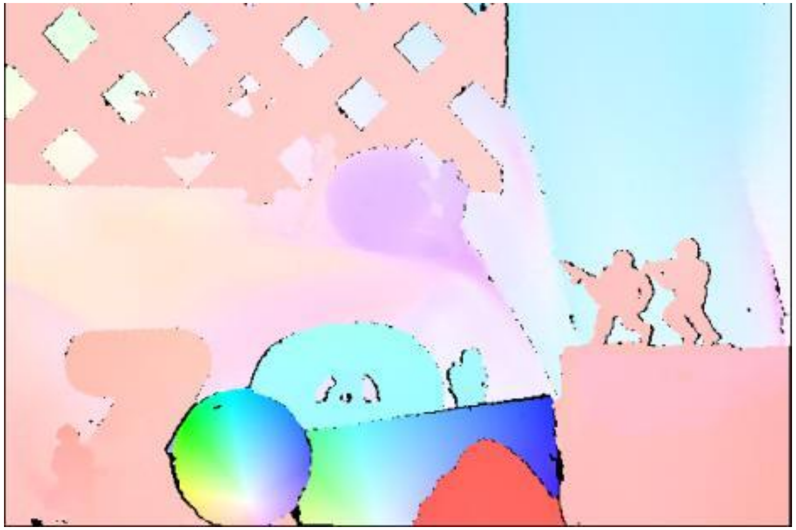


Flow quality evaluation

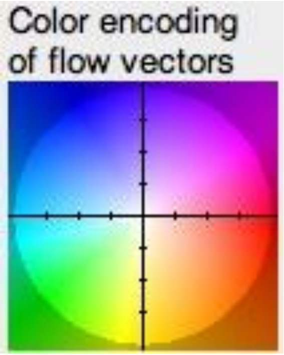
- Middlebury flow page
  - http://vision.middlebury.edu/flow/



Lucas-Kanade flow

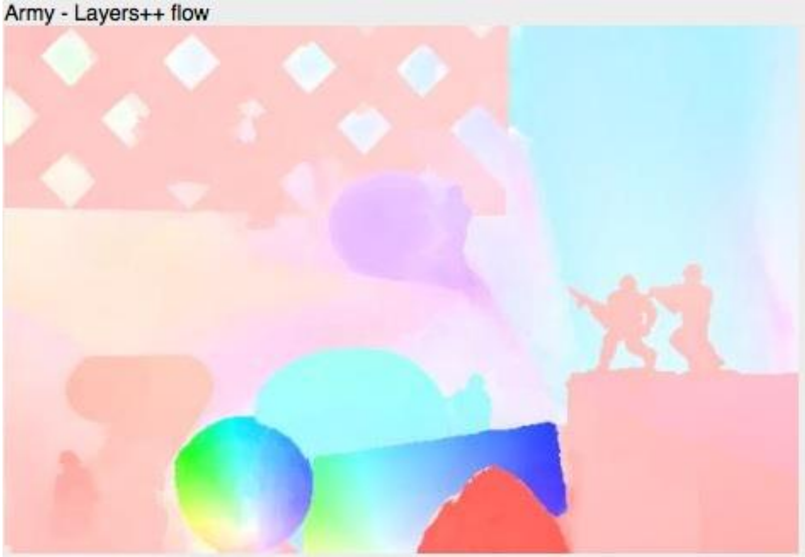


Ground Truth

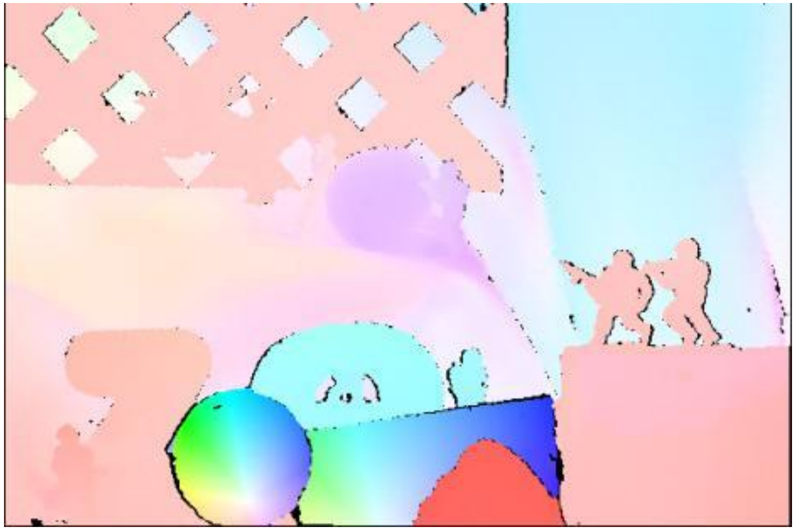


Flow quality evaluation

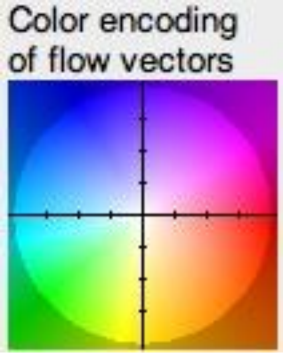
- Middlebury flow page
  - http://vision.middlebury.edu/flow/



Best-in-class alg

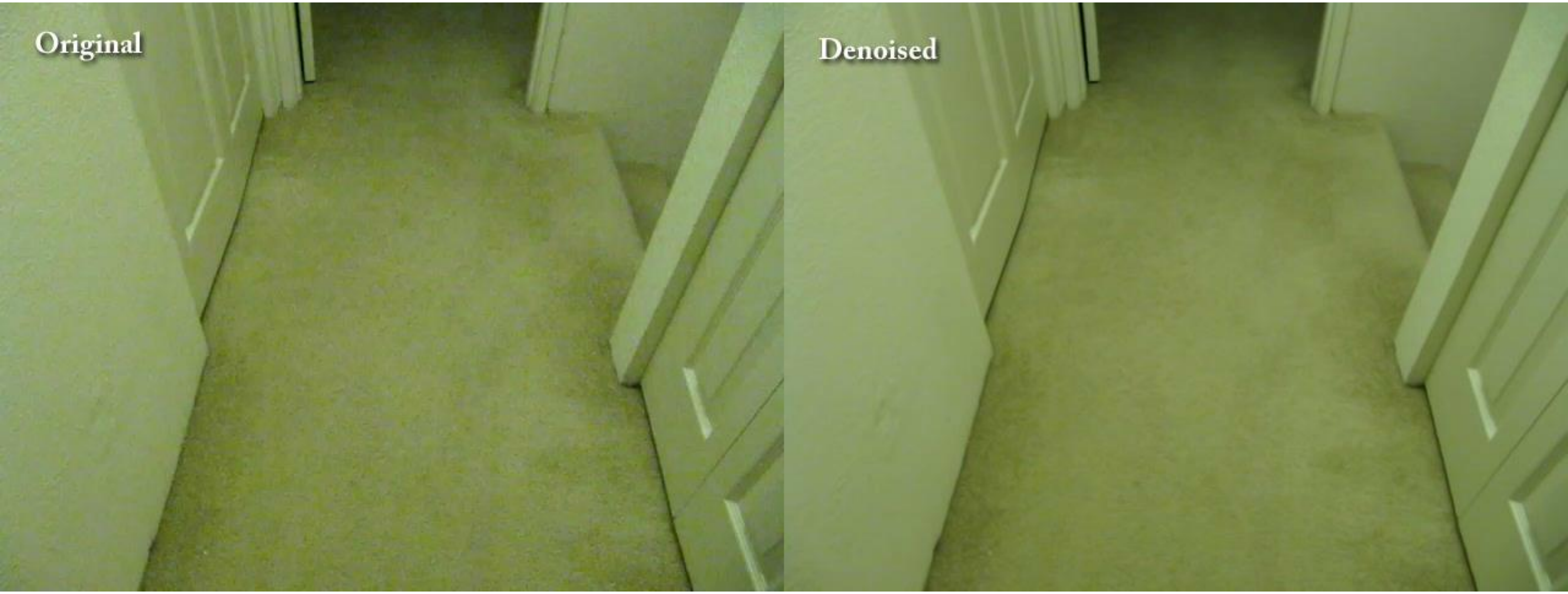


Ground Truth



Video stabilization

Video denoising



Video super resolution

Robust Visual Motion Analysis:

Piecewise-Smooth Optical Flow

Ming Ye

Electrical Engineering

University of Washington

Estimating Piecewise-Smooth Optical Flow with Global Matching and Graduated Optimization

Problem Statement:

Assuming only brightness conservation and piecewise-smooth motion, find the optical flow to best describe the intensity change in three frames.

Approach: Matching-Based Global Optimization

- **Step 1. Robust local gradient-based method for high-quality initial flow estimate.**

  Uses least median of squares instead of regular least squares.

- **Step 2. Global gradient-based method to improve the flow-field coherence.**

  Minimizes a global energy function E = Σ ($E_B(V_i) + E_S(V_i)$) where E_B is the brightness difference and E_S is the smoothness at flow vector V_i

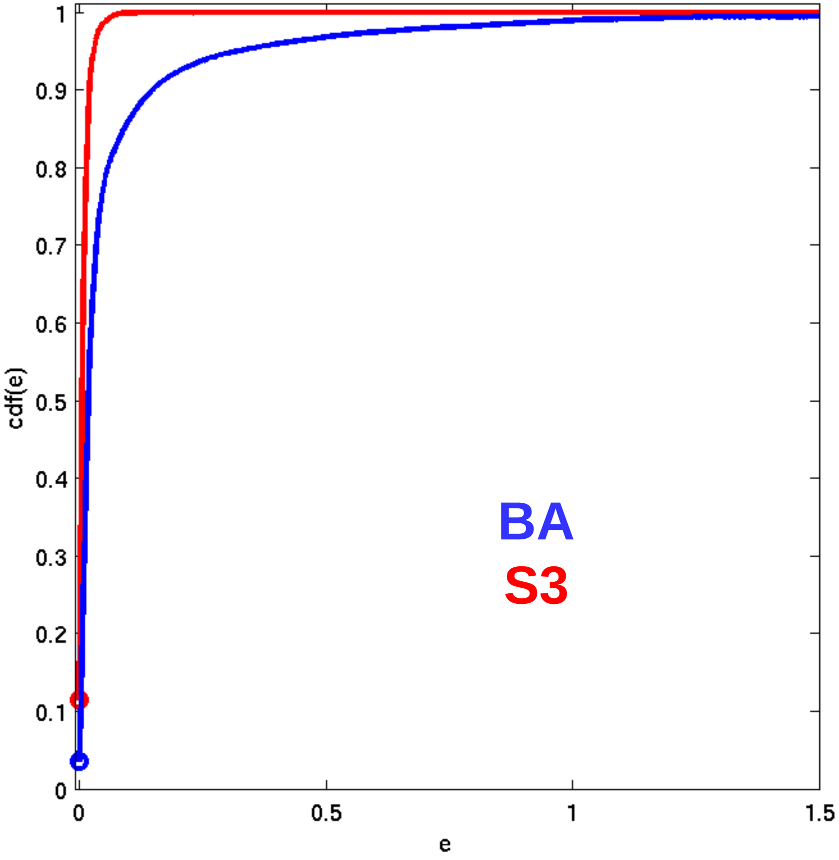
- **Step 3. Global matching that minimizes energy by a greedy approach.**

  Visits each pixel and updates it to be consistent with neighbors, iteratively.

TT: Translating Tree

150x150 (Barron 94)

	$e_{\angle}(^\circ)$	$e_{\lvert\bullet\rvert}(\text{pix})$	$\bar{e}(\text{pix})$
BA	2.60	0.128	0.0724
S3	0.248	0.0167	0.00984

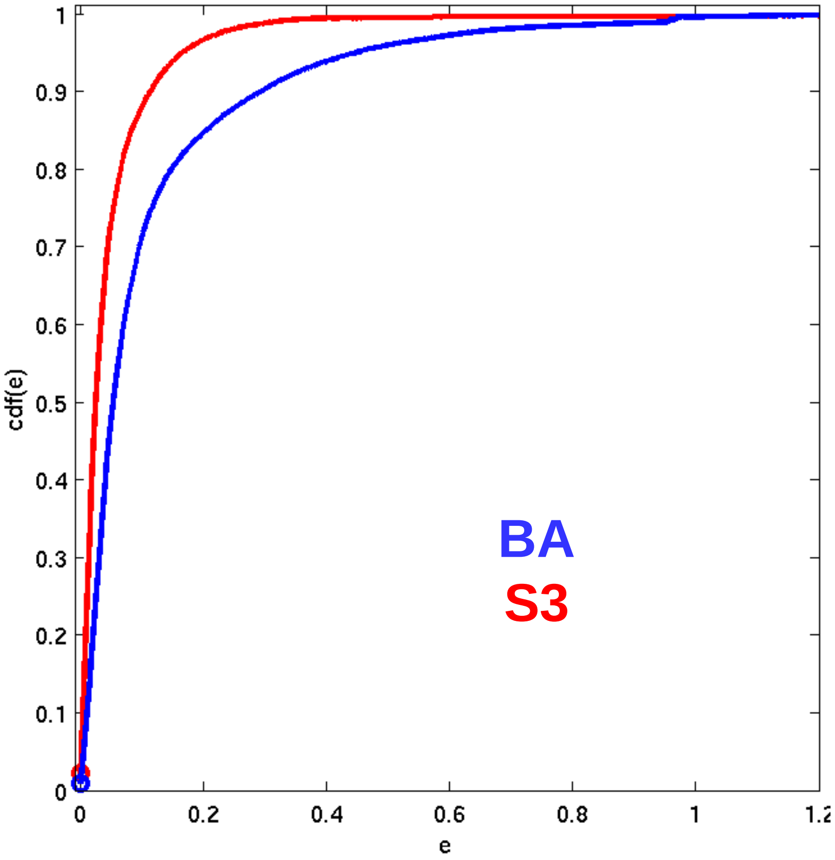


e: error in pixels, cdf: culmulative distribution function for all pixels

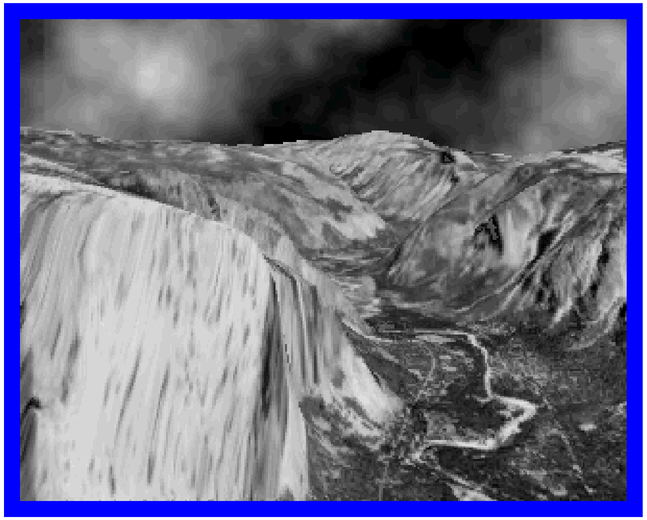
DT: Diverging Tree

150x150 (Barron 94)

	$e_{\angle}(^{\circ})$	$e_{\lvert\bullet\rvert}(\mathrm{pix})$	$\bar{e}(\mathrm{pix})$
BA	6.36	0.182	0.114
S3	2.60	0.0813	0.0507

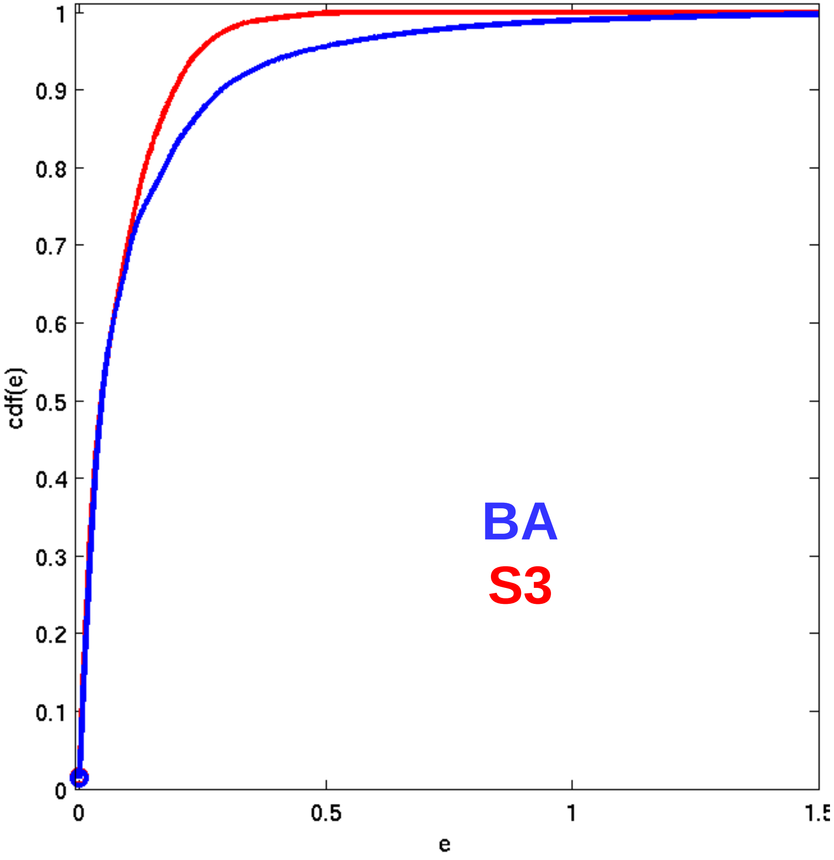


YOS: Yosemite Fly-Through

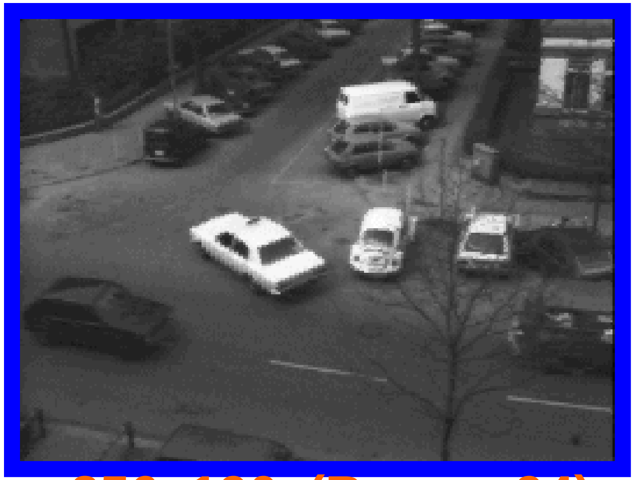


316x252 (Barron, cloud excluded)

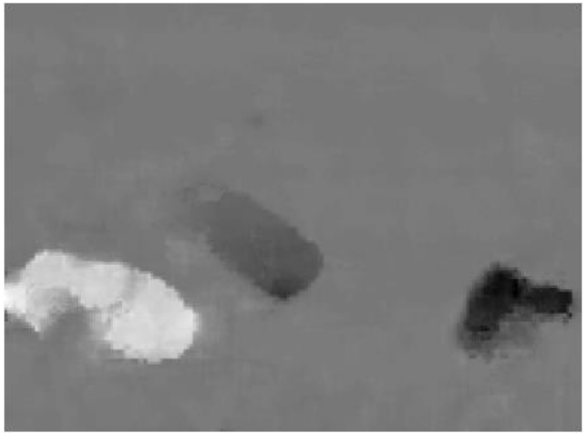
	$e_\angle(^\circ)$	$e_{\lvert\bullet\rvert}(\text{pix})$	$\bar{e}(\text{pix})$
BA	**2.71**	**0.185**	**0.118**
S3	**1.92**	**0.120**	**0.0776**



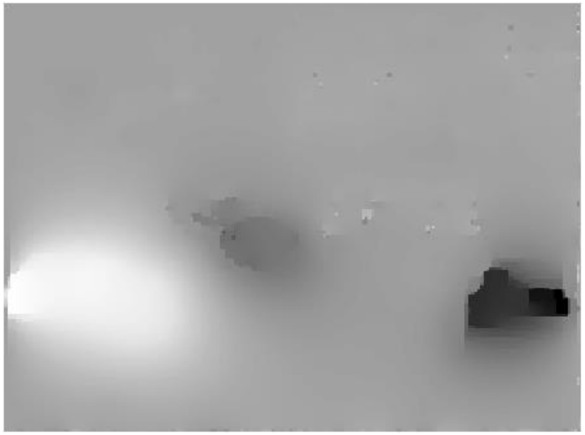
TAXI: Hamburg Taxi



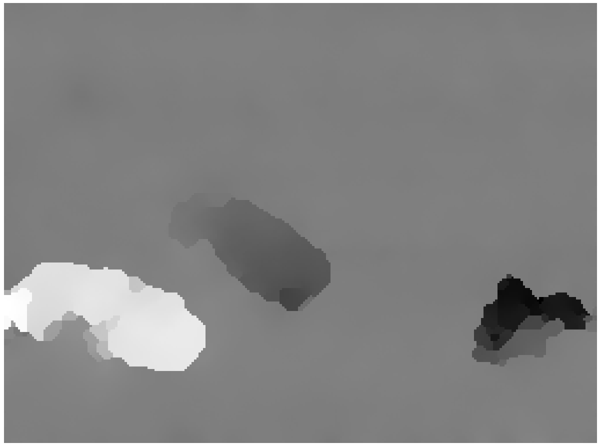
256x190, (Barron 94)
max speed 3.0 pix/frame



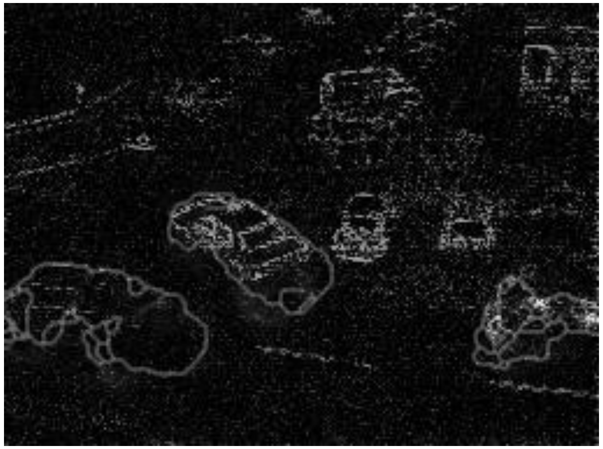
LMS



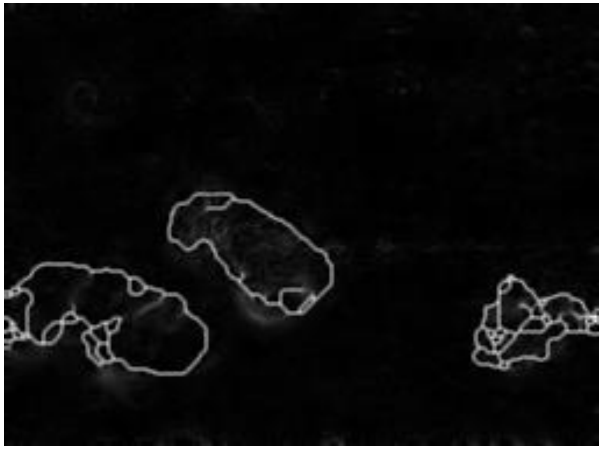
BA



Ours

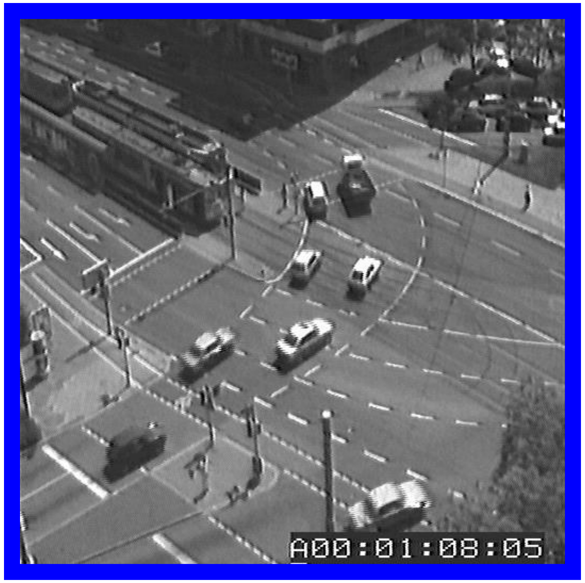


Error map

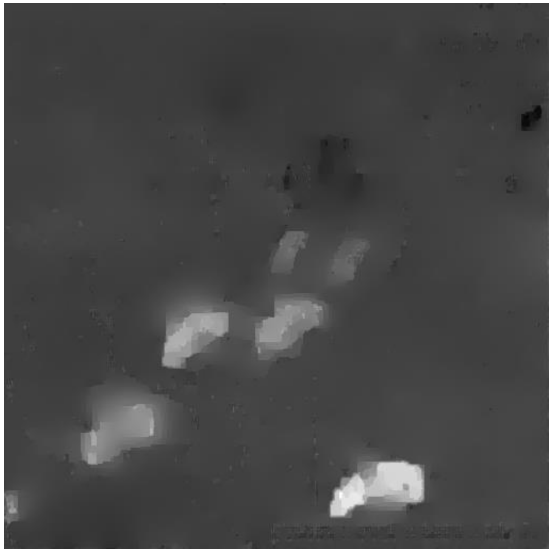


Smoothness error

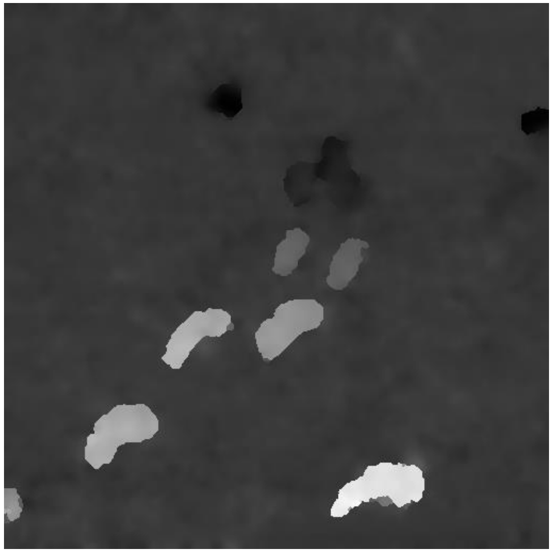
Traffic



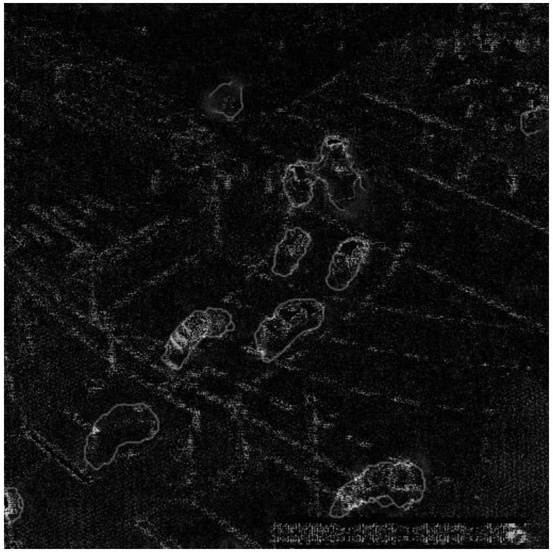
512x512
(Nagel)
max speed:
6.0 pix/frame



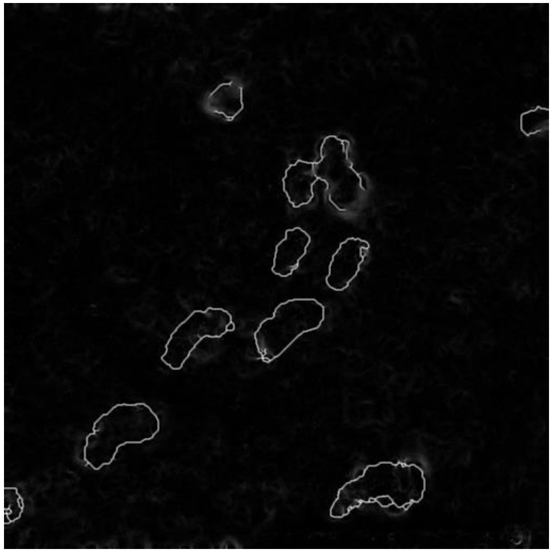
BA



Ours



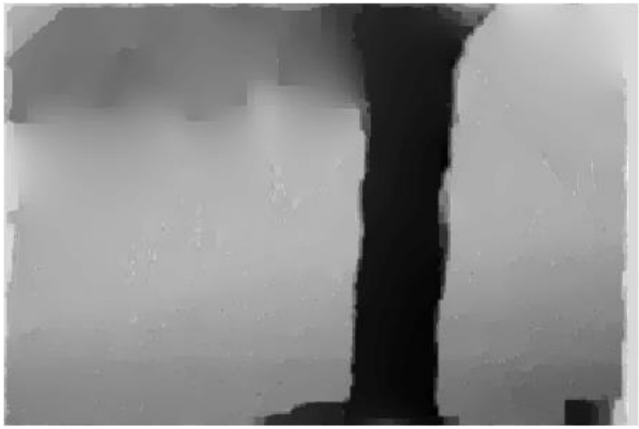
Error map



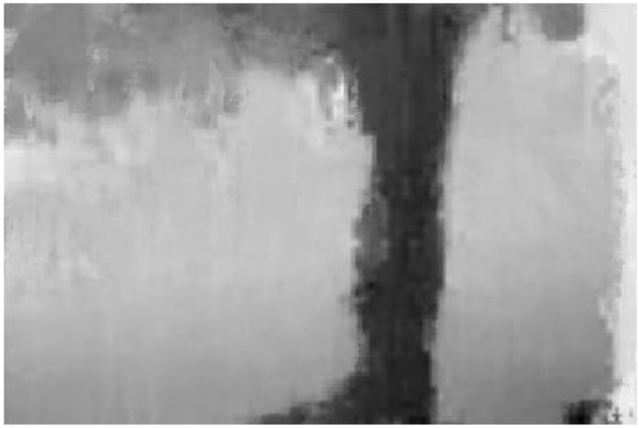
Smoothness error

FG: Flower Garden

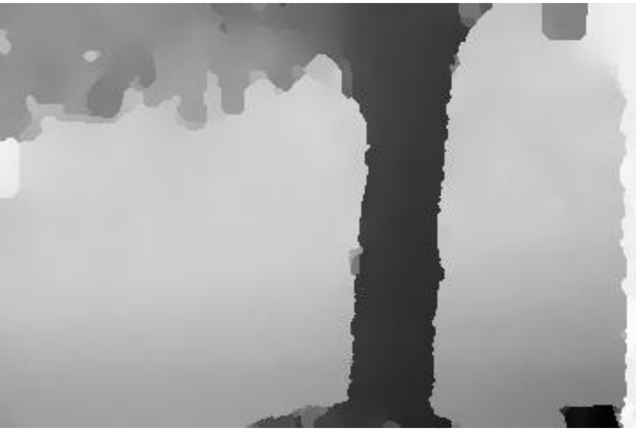
360x240 (Black)
Max speed: 7pix/frame



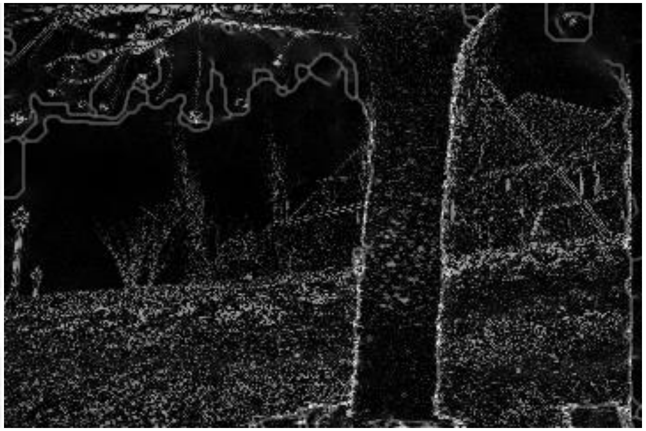
BA



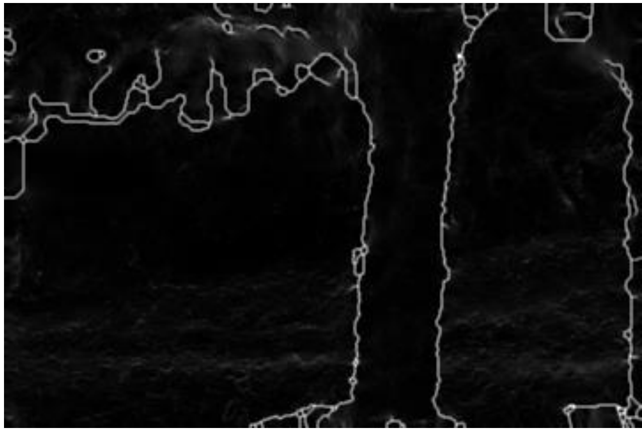
LMS



Ours



Error map



Smoothness error

Representing Moving Images with Layers

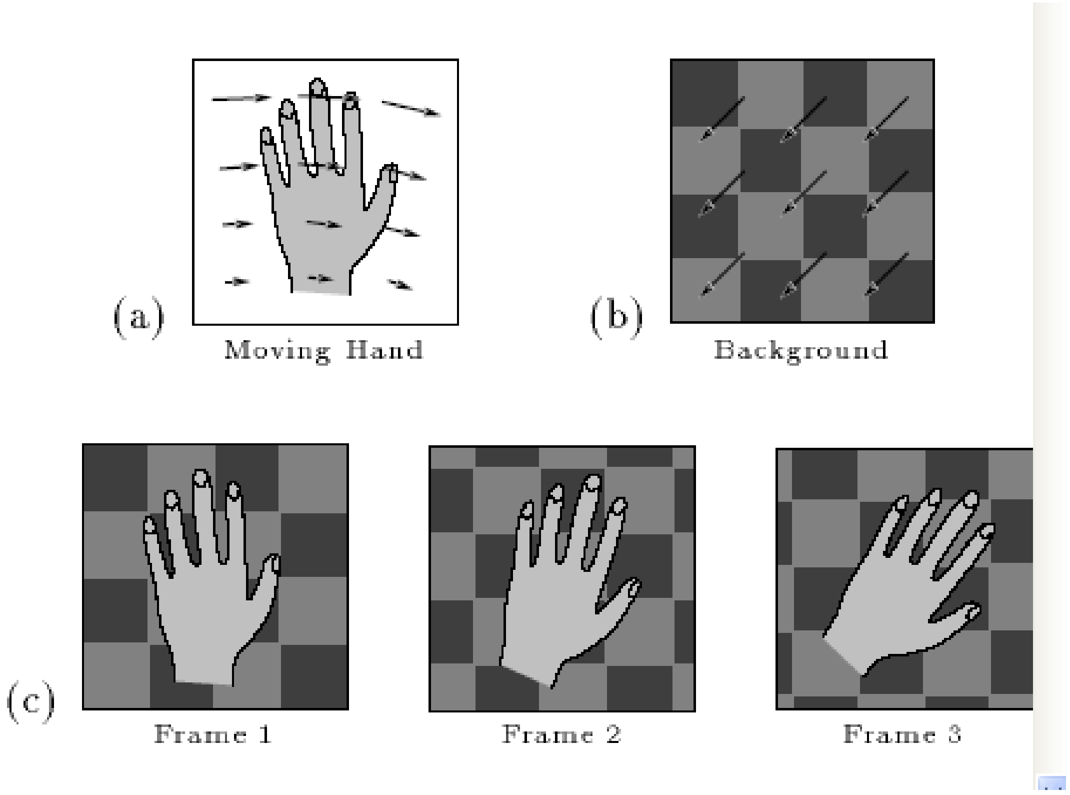
J. Y. Wang and E. H. Adelson

MIT Media Lab

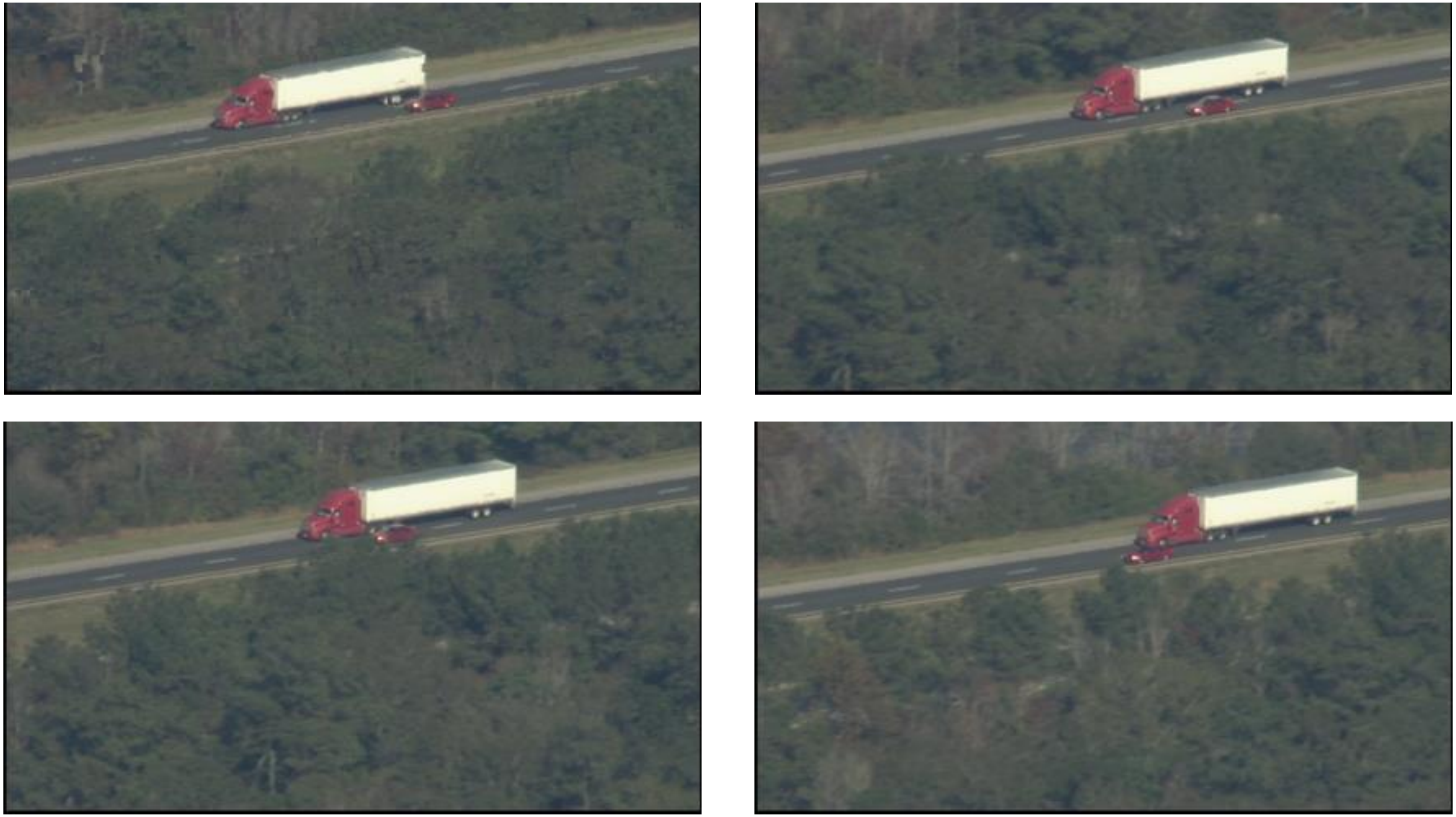
Goal

- Represent moving images with sets of overlapping layers
- Layers are ordered in depth and occlude each other
- Velocity maps indicate how the layers are to be warped over time

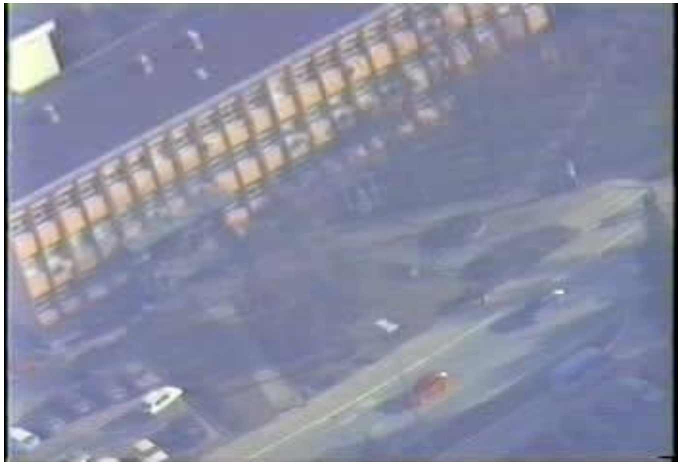
Simple Domain: Gesture Recognition

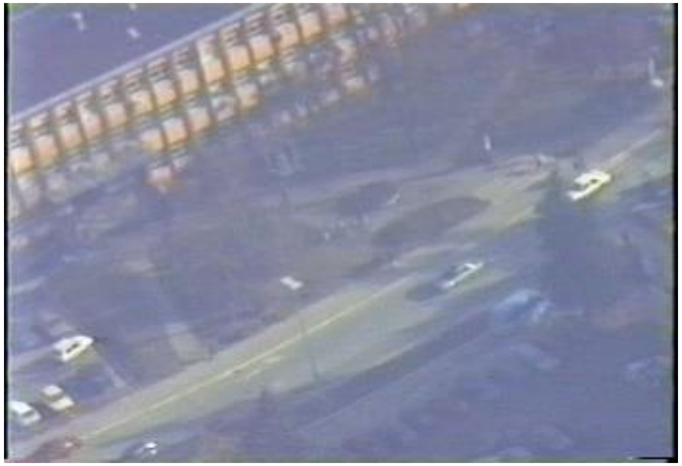


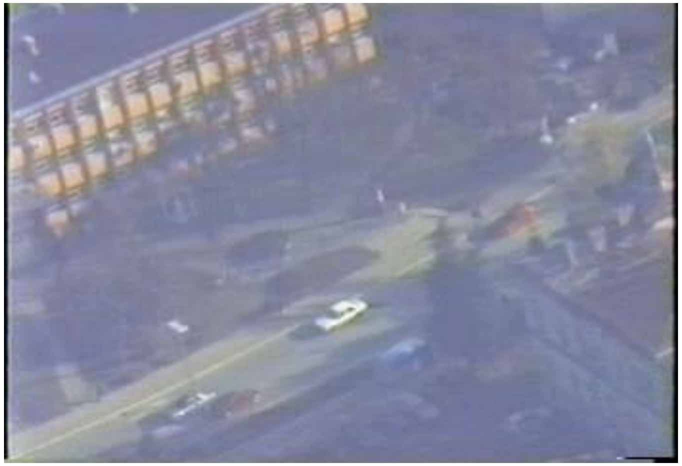
More Complex:
What are the layers?

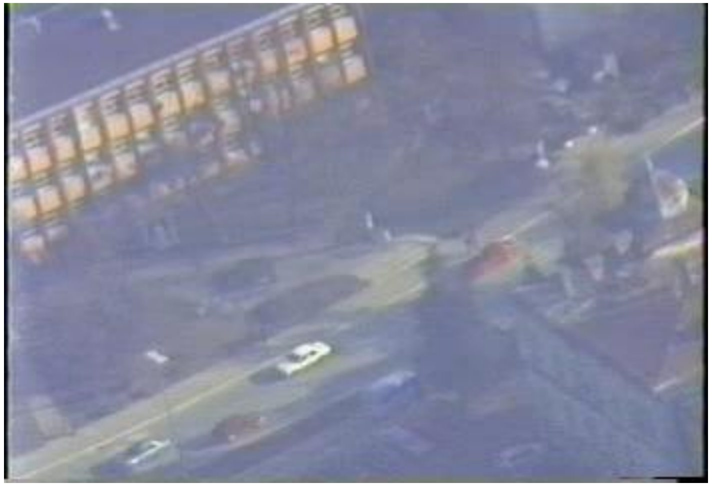


Even More Complex:
How many layers are there?









Definition of Layer Maps

- Each layer contains three maps

  1. intensity map (or texture map)

  2. alpha map (opacity at each point)

  3. velocity map (warping over time)

- Layers are ordered by depth

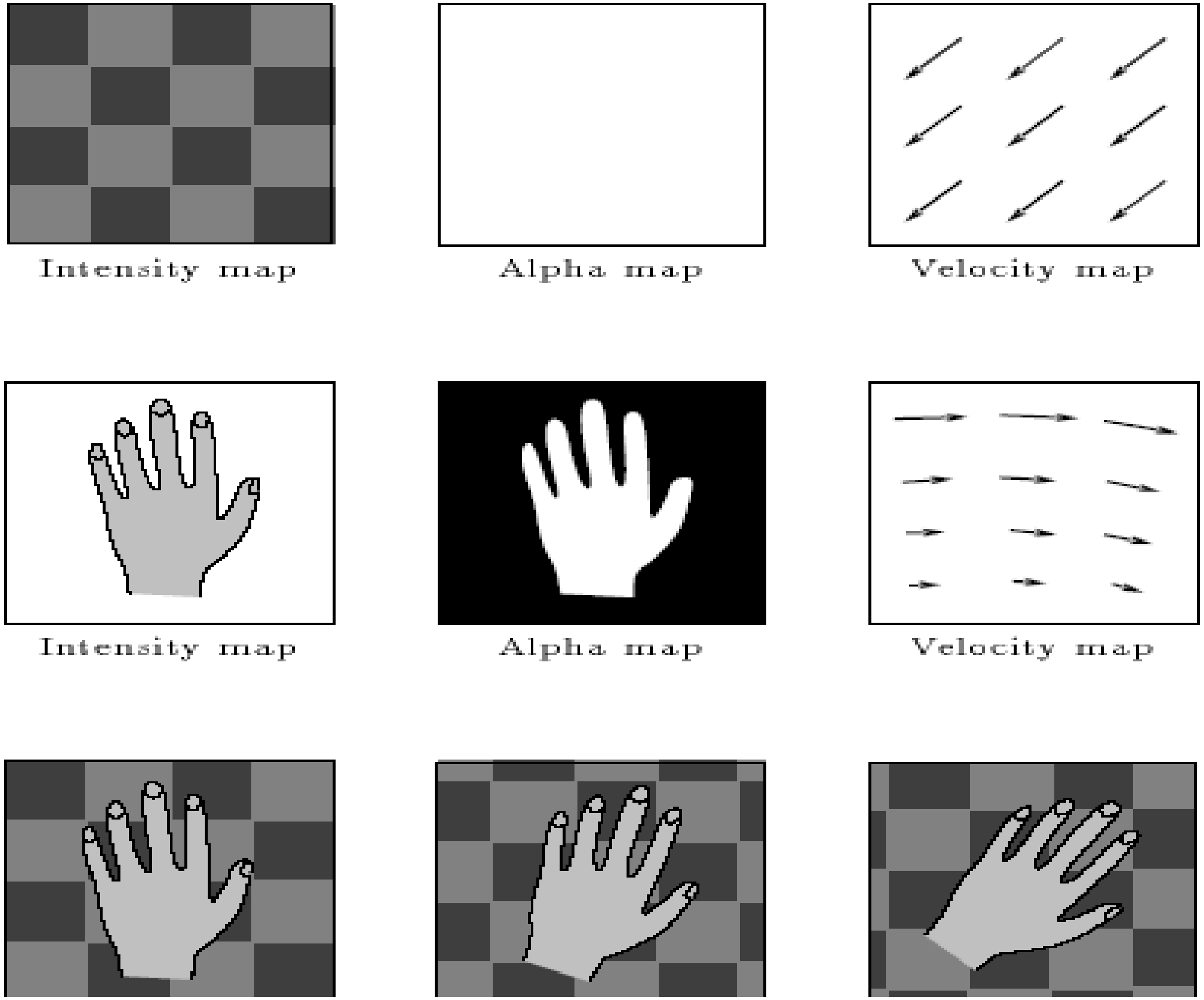
- This can be for vision or graphics or both

Layers for the Hand Gestures

Background Layer

Hand Layer

Re-synthesized Sequence



Optical Flow Doesn’t Work

- Optical flow techniques typically model the world as a 2-D rubber sheet that is distorted over time.
- When one object moves in front of another, the rubber sheet model fails.
- Image information appears and disappears; optical flow can’t represent this.
- Motion estimates near boundaries are bad.

Block Matching Can’t Do It

- Block motion only handles translation well.

- Like optical flow, block matching doesn’t deal with occlusion or objects that suddenly appear or disappear off the edge of the frame.

Layered Representation: Compositing

$$I_1(x, y) = E_0(x, y)(1 - \alpha_1(x, y)) + E_1(x, y)\alpha_1(x, y). \quad (1)$$

- E_0 is the background layer.
- E_1 is the next layer (and there can be more).
- α_1 is the alpha channel of E_1, with values between 0 and 1 (for graphics).
- The velocity map tells us how to warp the frames over time.
- The intensity map and alpha map are warped together, so they stay registered.

Analysis: Flower Garden Sequence

Frame 1 Frame 15 Frame 30

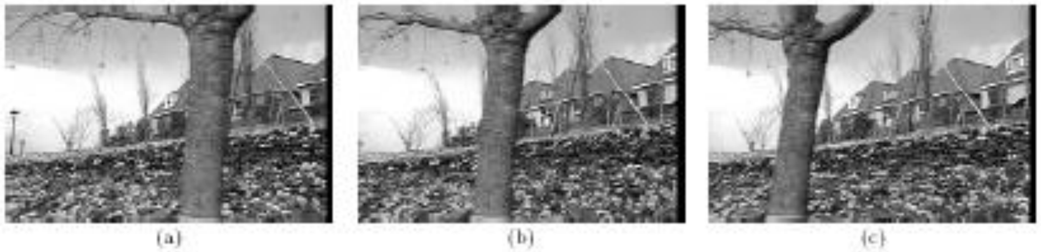
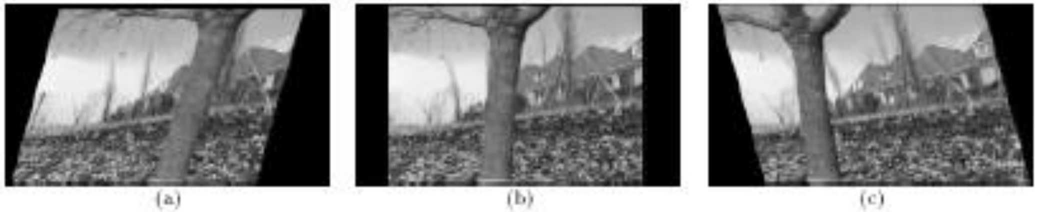


Figure 6: Frames 0, 15 and 30, of MPEG flower garden sequence are shown in figures (a-c), respectively.

Camera is panning to the right.



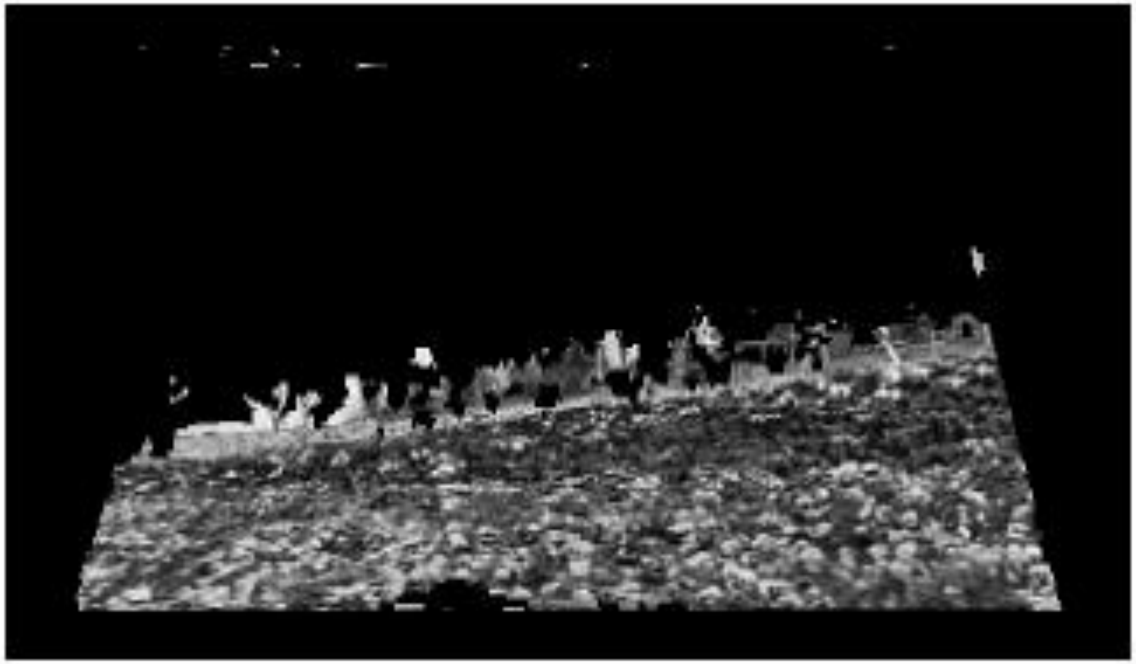
Frame 1 warped

Frame 15

Frame 30 warped

What’s going on here?

Accumulation of the Flowerbed Layer



Motion Analysis

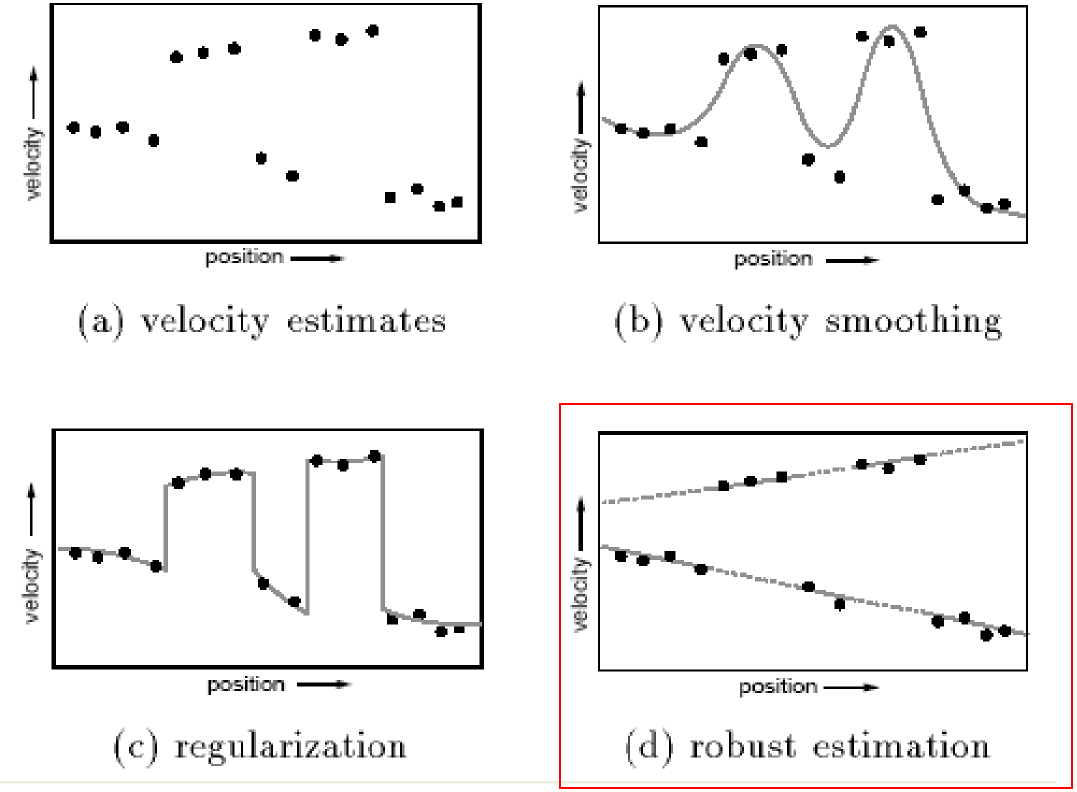
1. Robust motion segmentation using a parametric (affine) model.

$$V_x(x,y) = a_{x0} + a_{xx}x + a_{xy}y$$

$$V_y(x,y) = a_{y0} + a_{yx}x + a_{yy}y$$

2. Synthesis of the layered representation.

Motion Analysis Example



(a) velocity estimates

(b) velocity smoothing

(c) regularization

(d) robust estimation

2 separate layers shown as 2 affine models (lines);

The gaps show the occlusion.

Motion Estimation Steps

1. Conventional optical flow algorithm and representation (uses multi-scale, coarse-to-fine Lucas-Kanade approach).

2. From the optical flow representation, determine a set of affine motions. Segment into regions with an affine motion within each region.

Motion Segmentation

1. Use an array of non-overlapping square regions to derive an initial set of motion models.

2. Estimate the affine parameters within these regions by linear regression, applied separately on each velocity component (dx, dy).

3. Compute the reliability of each hypothesis according to its residual error.

4. Use an adaptive k-means clustering that merges two clusters when the distance between their centers is smaller than a threshold to produce a set of likely affine models.

Region Assignment by Hypothesis Testing

- Use the motion models derived from the motion segmentation step to identify the coherent regions.
- Do this by minimizing an error (distortion) function:

$$G(i(x,y)) = \sum_{x,y} (V(x,y) - V_{ai}(x,y))^2$$

where $i(x,y)$ is the model assigned to pixel (x,y)

and $V_{ai}(x,y)$ is the affine motion for that model.

- The error is minimized at each pixel to give the best model for that pixel position.
- Pixels with too high error are not assigned to models.

Iterative Algorithm

- The initial segmentation step uses an array of square regions.
- At each iteration, the segmentation becomes more accurate, because the parameter estimation is within a single coherent motion region.
- A region splitter separates disjoint regions.
- A filter eliminates small regions.
- At the end, intensity is used to match unassigned pixels to the best adjacent region.

Layer Synthesis

- The information from a longer sequence must be combined over time, to accumulate each layer.
- The transforms for each layer are used to warp its pixels to align a set of frames.
- The median value of a pixel over the set is used for the layer.
- Occlusion relationships are determined.

Results

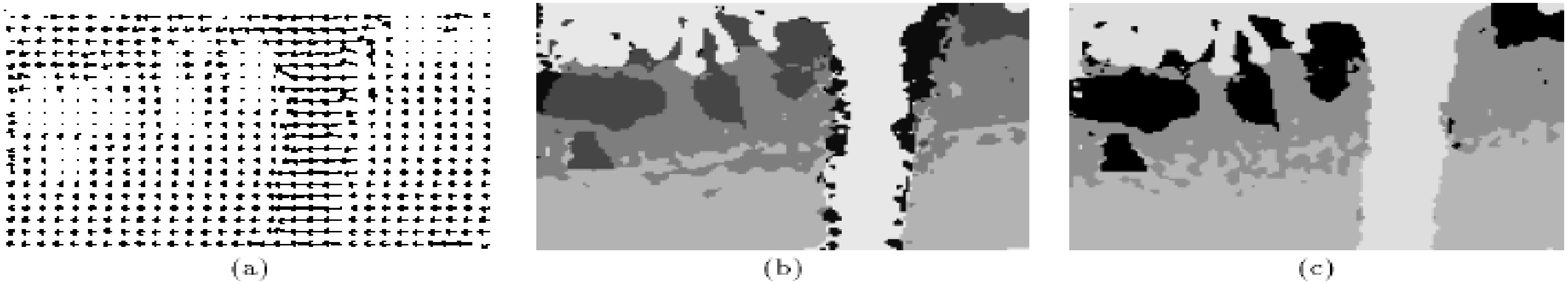

(a) (b) (c)

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.

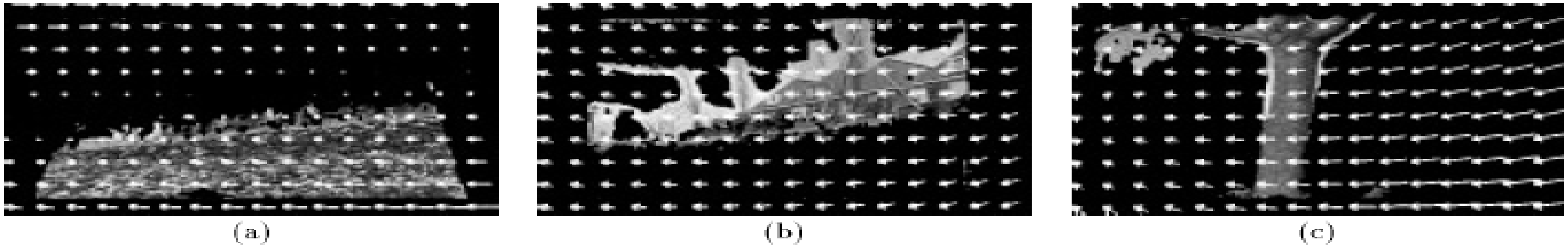
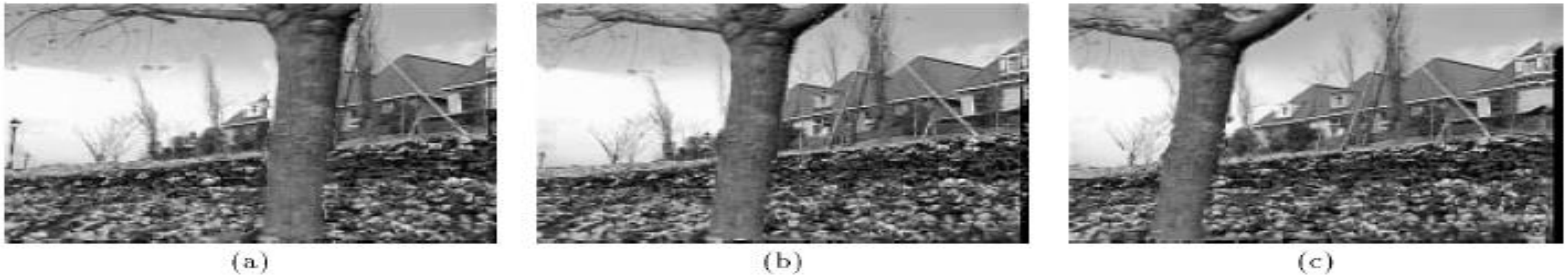

(a) (b) (c)

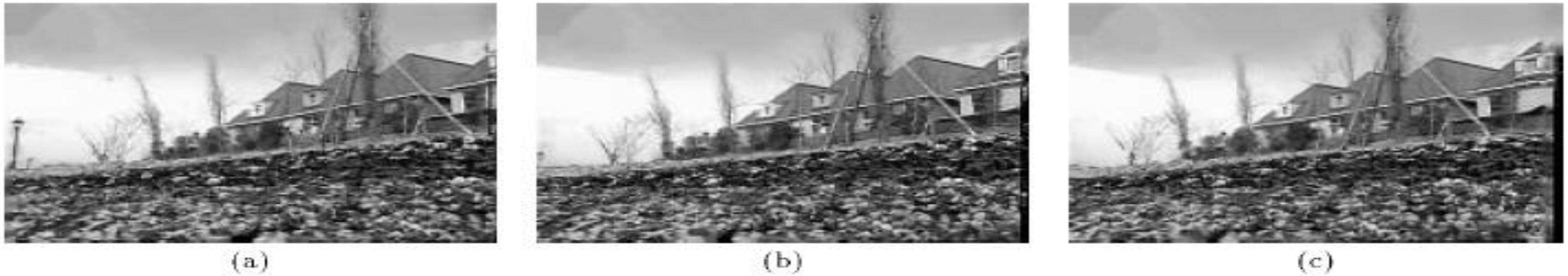
Figure 12: The layers corresponding to the tree, the flower bed, and the house shown in figures (a-c), respectively. The affine flow field for each layer is superimposed.

Results



(a) (b) (c)

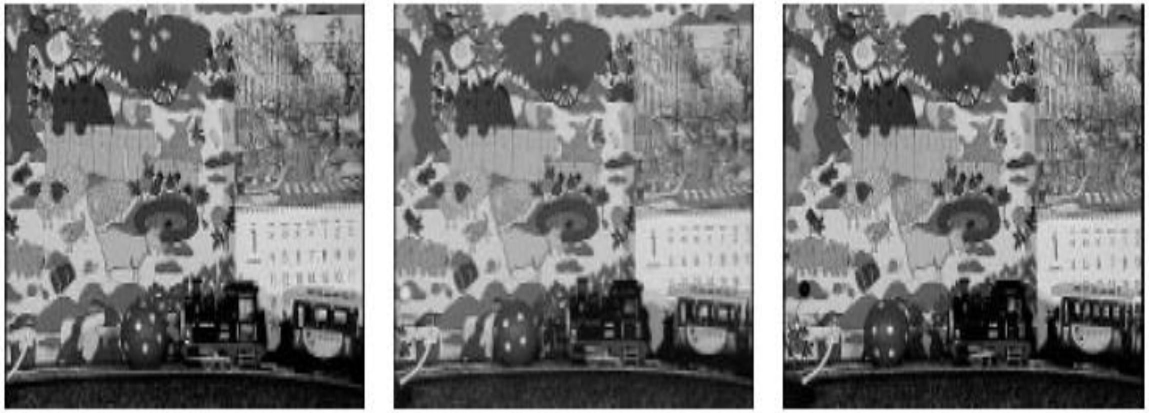
Figure 13: Frames 0, 15, and 30 as reconstructed from the layered representation shown in figures (a-c), respectively.



(a) (b) (c)

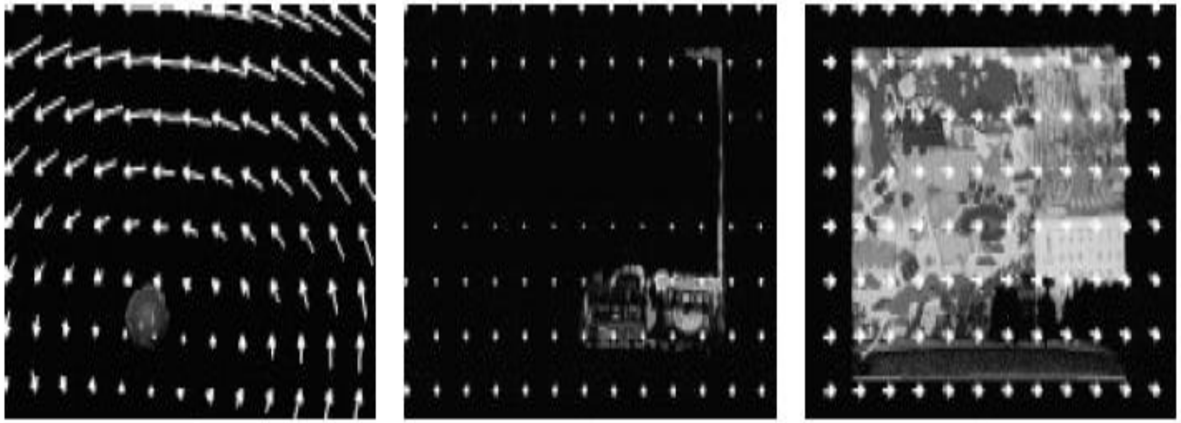
Figure 14: The sequence reconstructed without the tree layer shown in figures (a-c), respectively.

Results



(a) (b) (c)

Figure 15: Frames 0, 15 and 30, of MPEG Calendar sequence shown in figures (a-c), respectively.



(a) (b) (c)

Figure 16: The layers corresponding to the ball, the train, and the background shown in figures (a-c), respectively.

Summary

- Major contributions from Lucas, Tomasi, Kanade
  - Tracking feature points
  - Optical flow
  - Stereo
  - Structure from motion
- Key ideas
  - By assuming brightness constancy, truncated Taylor expansion leads to simple and fast patch matching across frames
  - Coarse-to-fine registration
  - Global approach by former EE student Ming Ye
  - Motion layers methodology by Wang and Adelson