Motion Estimation (II)

Ce Liu
celiu@microsoft.com
Microsoft Research New England
Last time

• Motion perception
• Motion representation
• Parametric motion: *Lucas-Kanade*
• Dense optical flow: *Horn-Schunck*

\[
\begin{align*}
\begin{bmatrix}
\frac{du}{dv}
\end{bmatrix} &=
-\begin{bmatrix}
I_x^T I_x & I_x^T I_y \\
I_x^T I_y & I_y^T I_y
\end{bmatrix}^{-1}
\begin{bmatrix}
I_x^T I_t \\
I_y^T I_t
\end{bmatrix} \\
\begin{bmatrix}
I_x^2 + \alpha L & I_x I_y \\
I_x I_y & I_y^2 + \alpha L
\end{bmatrix}
\begin{bmatrix}
U \\
V
\end{bmatrix} &=
-\begin{bmatrix}
I_x I_t \\
I_y I_t
\end{bmatrix}
\end{align*}
\]
Who are they?

Berthold K. P. Horn

Takeo Kanade
Content

• Robust optical flow estimation
• Applications
• Feature matching
• Discrete optical flow
• Layer motion analysis
• Other representations
Content

• Robust optical flow estimation
• Applications
• Feature matching
• Discrete optical flow
• Layer motion analysis
• Other representations
Spatial regularity

- Horn-Schunck is a Gaussian Markov random field (GMRF)
  \[
  \iint (I_x u + I_y v + I_t)^2 + \alpha (|\nabla u|^2 + |\nabla v|^2)dx\,dy
  \]
- Spatial over-smoothness is caused by the quadratic smoothness term
- Nevertheless, real optical flow fields are sparse!
Data term

• Horn-Schunck is a Gaussian Markov random field (GMRF)
\[ \iint (I_xu + I_yv + I_t)^2 + \alpha(|\nabla u|^2 + |\nabla v|^2) \, dx \, dy \]

• Quadratic data term implies Gaussian white noise

• Nevertheless, the difference between two corresponded pixels is caused by
  – Noise (majority)
  – Occlusion
  – Compression error
  – Lighting change
  – ...

• The error function needs to account for these factors
Noise model

- Explicitly model the noise $n$

$$I_2(x + u, y + v) = I_1(x, y) + n$$

- It can be a mixture of two Gaussians, **inlier** and **outlier**

$$n \sim \lambda N(0, \sigma_i^2) + (1 - \lambda) N(0, \sigma_o^2)$$
More components in the mixture

• Consider a Gaussian mixture model

\[ n \sim \frac{1}{Z} \sum_{k=1}^{K} \xi^k N(0, (ks)^2) \]

• Varying the decaying rate \( \xi \), we obtain a variety of potential functions
Typical error functions

L2 norm
\[ \rho(z) = z^2 \]

L1 norm
\[ \rho(z) = |z| \]

Truncated L1 norm
\[ \rho(z) = \min(|z|, \eta) \]

Lorentzian
\[ \rho(z) = \log(1 + \gamma z^2) \]
Robust statistics

- Traditional L2 norm: only noise, no outlier
- Example: estimate the average of 0.95, 1.04, 0.91, 1.02, 1.10, 20.01
- Estimate with minimum error

\[ z^* = \arg \min_z \sum_i \rho(z - z_i) \]

- L2 norm: \( z^* = 4.172 \)
- L1 norm: \( z^* = 1.038 \)
- Truncated L1: \( z^* = 1.0296 \)
- Lorentzian: \( z^* = 1.0147 \)
The family of robust power functions

• Can we directly use L1 norm $\psi(z) = |z|$?
  – Derivative is not continuous

• Alternative forms
  – L1 norm: $\psi(z^2) = \sqrt{z^2 + \varepsilon^2}$
  – Sub L1: $\psi(z^2; \eta) = (z^2 + \varepsilon^2)^\eta$, $\eta < 0.5$
Modification to Horn-Schunck

- Let \( x = (x, y, t) \), and \( w(x) = (u(x), v(x), 1) \) be the flow vector
- Horn-Schunck (recall)
  \[
  \iint (I_xu + I_yv + I_t)^2 + \alpha (|\nabla u|^2 + |\nabla v|^2) \, dx \, dy
  \]
- Robust estimation
  \[
  \iint \psi(|I(x + w) - I(x)|^2) + \alpha \phi(|\nabla u|^2 + |\nabla v|^2) \, dx \, dy
  \]
- Robust estimation with Lucas-Kanade
  \[
  \iint g * \psi(|I(x + w) - I(x)|^2) + \alpha \phi(|\nabla u|^2 + |\nabla v|^2) \, dx \, dy
  \]
A unifying framework

- The robust object function

\[ \int\int g \ast \psi(|I(x+w) - I(x)|^2) + \alpha \phi(|\nabla u|^2 + |\nabla v|^2) dxdy \]

- Lucas-Kanade: \( \alpha = 0, \psi(z^2) = z^2 \)
- Robust Lucas-Kanade: \( \alpha = 0, \psi(z^2) = \sqrt{z^2 + \varepsilon^2} \)
- Horn-Schunck: \( g = 1, \psi(z^2) = z^2, \phi(z^2) = z^2 \)

- One can also learn the filters (other than gradients), and robust function \( \psi(\cdot), \phi(\cdot) \) [Roth & Black 2005]
Derivation strategies

• Euler-Lagrange
  – Derive in continuous domain, discretize in the end
  – Nonlinear PDE’s
  – Outer and inner fixed point iterations
  – Limited to derivative filters; cannot generalize to arbitrary filters

• Energy minimization
  – Discretize first and derive in matrix form
  – Easy to understand and derive
  – Iteratively reweighted least square (IRLS)

• Variational optimization

• Euler-Lagrange = Variational optimization = IRLS
Iteratively reweighted least square (IRLS)

- Let \( \phi(z) = (z^2 + \varepsilon^2)\eta \) be a robust function
- We want to minimize the objective function

\[
\Phi(Ax + b) = \sum_{i=1}^{n} \phi \left( (a_i^T x + b_i)^2 \right)
\]

where \( x \in \mathbb{R}^d, A = [a_1 \ a_2 \ \ldots \ a_n]^T \in \mathbb{R}^{n \times d}, b \in \mathbb{R}^n \)

- By setting \( \frac{\partial \Phi}{\partial x} = 0 \), we can derive

\[
\frac{\partial \Phi}{\partial x} \propto \sum_{i=1}^{n} \phi' \left( (a_i^T x + b_i)^2 \right) (a_i^T x + b_i) a_i \\
= \sum_{i=1}^{n} w_{ii} a_i^T x a_i + w_{ii} b_i a_i \\
= \sum_{i=1}^{n} a_i^T w_{ii} x a_i + b_i w_{ii} a_i \\
= A^T W A x + A^T W b \\
\]

\[
w_{ii} = \phi' \left( (a_i^T x + b_i)^2 \right)
\]

\[
W = \text{diag}(\Phi'(Ax + b))
\]
Iteratively reweighted least square (IRLS)

- Derivative: \( \frac{\partial \Phi}{\partial x} = A^T W A x + A^T W b = 0 \)
- Iterate between *reweighting* and *least square*

1. Initialize \( x = x_0 \)
2. Compute weight matrix \( W = \text{diag}(\Phi'(Ax + b)) \)
3. Solve the linear system \( A^T W A x = -A^T W b \)
4. If \( x \) converges, return; otherwise, go to 2

- Convergence is guaranteed (local minima)
IRLS for robust optical flow

• Objective function

\[
\iint g \ast \psi(|I(x + w) - I(x)|^2) + \alpha \phi(|\nabla u|^2 + |\nabla v|^2) \, dx \, dy
\]

• Discretize, linearize and increment

\[
\sum_{x,y} g \ast \psi \left(|I_t + I_x du + I_y dv|^2\right) + \alpha \phi(|\nabla (u + du)|^2 + |\nabla (v + dv)|^2)
\]

• IRLS (initialize \(du = dv = 0\))
  - Reweight: \(\Psi'_{xx} = \text{diag}(g \ast \psi'I_x I_x), \Psi'_xy = \text{diag}(g \ast \psi'I_x I_y), \Psi'_yy = \text{diag}(g \ast \psi'I_y I_y), \Psi'_xt = \text{diag}(g \ast \psi'I_x I_t), \Psi'_yt = \text{diag}(g \ast \psi'I_y I_t), L = D_x^T \Phi' D_x + D_y^T \Phi' D_y\)
  - Least square:

\[
\begin{bmatrix}
\Psi'_{xx} + \alpha L & \Psi'_{xy} \\
\Psi'_{xy} & \Psi'_{yy} + \alpha L
\end{bmatrix}
\begin{bmatrix}
dU \\
dV
\end{bmatrix}
= - \begin{bmatrix}
\Psi'_{xt} + \alpha L U \\
\Psi'_{yt} + \alpha LV
\end{bmatrix}
\]
What’s changed?

• Optical flow with robust function

\[
\begin{align*}
\Psi'_{xx} &= \text{diag}(g \ast \psi'I_xI_x), \quad \Psi'_{xy} = \text{diag}(g \ast \psi'I_xI_y), \\
\Psi'_{yy} &= \text{diag}(g \ast \psi'I_yI_y), \quad \Psi'_{xt} = \text{diag}(g \ast \psi'I_xI_t), \\
\Psi'_{yt} &= \text{diag}(g \ast \psi'I_yI_t), \quad \mathbf{L} = \mathbf{D}_x^T\Phi'\mathbf{D}_x + \mathbf{D}_y^T\Phi'\mathbf{D}_y
\end{align*}
\]

\[
\begin{bmatrix}
\Psi'_{xx} + \alpha\mathbf{L} & \Psi'_{xy} \\
\Psi'_{xy} & \Psi'_{yy} + \alpha\mathbf{L}
\end{bmatrix}
\begin{bmatrix}
dU \\
dV
\end{bmatrix} = - \begin{bmatrix}
\Psi'_{xt} + \alpha\mathbf{L}U \\
\Psi'_{yt} + \alpha\mathbf{LV}
\end{bmatrix}
\]

• Horn-Schunck

\[
\begin{bmatrix}
I_x^2 + \alpha\mathbf{L} & I_xI_y \\
I_xI_y & I_y^2 + \alpha\mathbf{L}
\end{bmatrix}
\begin{bmatrix}
dU \\
dV
\end{bmatrix} = - \begin{bmatrix}
I_xI_t + \alpha\mathbf{L}U \\
I_yI_t + \alpha\mathbf{LV}
\end{bmatrix}
\]
Example

Input two frames

Robust optical flow

Horn-Schunck

Flow visualization

Coarse-to-fine LK with median filtering
Content

• Robust optical flow estimation

• Applications

• Feature matching

• Discrete optical flow

• Layer motion analysis

• Other representations
Video stabilization
Video denoising
Video super resolution

Low-Res
Content

• Robust optical flow estimation
• Applications
  • Feature matching
• Discrete optical flow
• Layer motion analysis
• Contour motion analysis
• Obtaining motion ground truth
Block matching

- Both Horn-Schunck and Lucas-Kanade are sub-pixel accuracy algorithms
- But in practice we may not need sub-pixel accuracy
- MPEG: $16 \times 16$ block matching using MMSE
- H264: variable block size and quarter-pixel precision
Tracking reliable features

• Idea: no need to work on ambiguous region pixels (flat regions & line structures)

• Instead, we can track features and then propagate the tracking to ambiguous pixels

• Good features to track [Shi & Tomashi 94]

\[
\begin{bmatrix} du \\ dv \end{bmatrix} = - \begin{bmatrix} I_x^T I_x & I_x^T I_y \\ I_x^T I_y & I_y^T I_y \end{bmatrix}^{-1} \begin{bmatrix} I_x^T I_t \\ I_y^T I_t \end{bmatrix}
\]

• Block matching + Lucas-Kanade refinement
Feature detection & tracking
From sparse to dense

• Interpolation: given values \( \{d_i\} \) at \( \{(x_i, y_i)\} \), reconstruct a smooth plane \( f(x, y) \)

• Membrane model (first order smoothness)

\[
\iint \sum_i \left( w_i (f(x_i, y_i) - d_i)^2 + \alpha (f_x^2 + f_y^2) \right) \, dx \, dy
\]

• Thin plate model (second order smoothness)

\[
\iint \sum_i \left( w_i (f(x_i, y_i) - d_i)^2 + \alpha (f_{xx}^2 + f_{xy}^2 + f_{yy}^2) \right) \, dx \, dy
\]
Membrane vs. thin plate

Fig. 1. Sample data points and interpolated solutions: (a) sample data points, (b) membrane interpolant, (c) thin plate interpolant, (d) controlled continuity spline (thin plate with discontinuities and creases).
Dense flow field from sparse tracking
Pros and Cons of Feature Matching

• Pros
  – Efficient (a few feature points vs. all pixels)
  – Reliable (with advanced feature descriptors)

• Cons
  – Independent tracking (tracking can be unreliable)
  – Not all information is used (may not capture weak features)

• How to improve
  – Track every pixel with uncertainty
  – Integrate spatial regularity (neighboring pixels go together)
Content

- Robust optical flow estimation
- Applications
- Feature matching
  - Discrete optical flow
- Layer motion analysis
- Other representations
Discrete optical flow

• The objective function is similar to that of continuous flow
• \( x = (x, y) \) is pixel coordinate, \( w = (u, v) \) is flow vector

\[
E(w) = \sum_x \min(|I_1(x) - I_2(x + w(x))|, t) + \\
\sum_x \eta(|u(x)| + |v(x)|) + \\
\sum_{(x_1, x_2) \in \varepsilon} \min(\alpha |u(x_1) - u(x_2)|, d) + \min(\alpha |v(x_1) - v(x_2)|, d)
\]

• Truncated L1 norms:
  – Account for outliers in the data term
  – Encourage piecewise smoothness in the smoothness term

Data term
Small displacement
Spatial regularity
Decoupled smoothness

Coupled smoothness
\[ \sqrt{u_x^2 + v_x^2} \]

Decoupled smoothness
\[ |u_x| + |v_x| \]
Combinatorial optimization on graph

\[
E(w) = \sum_x \min(|I_1(x) - I_2(x + w(x))|, t) + \\
\sum_x \eta(|u(x)| + |v(x)|) + \\
\sum_x \min(\alpha |u(x_1) - u(x_2)|, d) + \min(\alpha |v(x_1) - v(x_2)|, d)
\]

- Optimization strategies
  - Belief propagation
  - Graph cuts
  - MCMC (simulated annealing)
Dual-layer belief propagation

\[ w = (u, v) \]

- **Data term**
  \[ \min(|I_1(x) - I_2(x + w)|, t) \]

- **Smoothness term on \( u \)**
  \[ \min(\alpha|u(x_1) - u(x_2)|, d) \]

- **Smoothness term on \( v \)**
  \[ \min(\alpha|v(x_1) - v(x_2)|, d) \]

- **Regularization term on \( u \)**
  \[ \eta|u(x)| \]

- **Regularization term on \( v \)**
  \[ \eta|v(x)| \]

[Shekhovtsov et al. CVPR 07]
Dual-layer belief propagation

Message $M_j^k$: given all the information at node $k$, predict the distribution at node $j$.

Update within $u$ plane.
Dual-layer belief propagation

Update within $v$ plane
Dual-layer belief propagation

Update from $u$ plane to $v$ plane
Dual-layer belief propagation

Update from $\nu$ plane to $u$ plane
Example

Input two frames

Discrete optical flow

Robust optical flow

Flow visualization

Coarse-to-fine LK with median filtering
Content

- Robust optical flow estimation
- Applications
- Feature matching
- Discrete optical flow
- Layer motion analysis
- Other representations
Layer representation

- Optical flow field is able to model complicated motion

- Different angle: a video sequence can be a composite of several moving layers

- Layers have been widely used
  - Adobe Photoshop
  - Adobe After Effect

- Compositing is straightforward, but inference is hard

Wang & Adelson, 1994
Wang & Adelson, 1994

• **Strategy**
  - Obtaining dense optical flow field
  - Divide a frame into non-overlapping regions and fit affine motion for each region
  - Cluster affine motions by k-means clustering
  - Region assignment by hypothesis testing
  - Region splitter: disconnected regions are separated
Results

- Optical flow field
- Clustering to affine regions
- Clustering with error metric

Flower garden

Three layers with affine motion superimposed

Reconstructed background layer
Weiss & Adelson, 1996

• Chicken & egg problem
  – Good motion → good segmentation
  – Good segmentation → good motion

• We don’t have either of them, so iterate!

• Perceptually organized expectation & maximization (POEM)
  – E-step: estimate the motion parameter of each layer
  – M-step: estimate the likelihood that a pixel belongs to each of the layers (segmentation)
Liu et. al. 2005

- Reliable layer segmentation for motion magnification
- Layer segmentation pipeline

Feature point tracking → Trajectory clustering → Dense optical flow interpolation → Layer segmentation
Normalized Complex Correlation

- The similarity metric should be independent of phase and magnitude.
- Normalized complex correlation

\[ S(C_1, C_2) = \frac{\left| \sum_t C_1(t) \overline{C_2(t)} \right|^2}{\sqrt{\sum_t C_1(t) \overline{C_1(t)}} \sqrt{\sum_t C_2(t) \overline{C_2(t)}}} \]
Spectral Clustering

Feature point tracking → Trajectory clustering → Dense optical flow interpolation → Layer segmentation

Affinity matrix → Clustering → Reordering of affinity matrix
Clustering Results

Feature point tracking → **Trajectory clustering** → Dense optical flow interpolation → Layer segmentation
From Sparse Feature Points to Dense Optical Flow Field

- Interpolate dense optical flow field using locally weighted linear regression

Cluster 1: leaves
Cluster 2: swing
Motion Layer Assignment

- Assign each pixel to a motion cluster layer, using four cues:
  - **Motion likelihood**—consistency of pixel’s intensity if it moves with the motion of a given layer (dense optical flow field)
  - **Color likelihood**—consistency of the color in a layer
  - **Spatial connectivity**—adjacent pixels favored to belong the same group
  - **Temporal coherence**—label assignment stays constant over time

- Energy minimization using graph cuts
Motion Magnification

Ce Liu
Antonio Torralba
William T. Freeman
Fredo Durand
Edward H. Adelson

Massachusetts Institute of Technology
Computer Science and Artificial Intelligence Laboratory

SIGGRAPH 2005
The 32nd International Conference on Computer Graphics and Interactive Techniques
How good is optical flow?

• The AAE (average angular error) race on the *Yosemite* sequence for over 15 years

#I. Austvoll. Lecture Notes in Computer Science, 2005

Middlebury flow database

Middlebury flow database

<table>
<thead>
<tr>
<th>Average endpoint error</th>
<th>Army (Hidden texture)</th>
<th>Mequon (Hidden texture)</th>
<th>Schefflera (Hidden texture)</th>
<th>Wooden (Hidden texture)</th>
<th>Grove (Synthetic)</th>
<th>Urban (Synthetic)</th>
<th>Yosemite (Synthetic)</th>
<th>Teddy (Stereo)</th>
</tr>
</thead>
<tbody>
<tr>
<td>avg. rank</td>
<td>avg. rank</td>
<td>avg. rank</td>
<td>avg. rank</td>
<td>avg. rank</td>
<td>avg. rank</td>
<td>avg. rank</td>
<td>avg. rank</td>
<td>avg. rank</td>
</tr>
<tr>
<td>4.4</td>
<td>0.03</td>
<td>0.26</td>
<td>0.06</td>
<td>0.23</td>
<td>0.78</td>
<td>0.54</td>
<td>0.47</td>
<td>0.31</td>
</tr>
<tr>
<td>5.7</td>
<td>0.11</td>
<td>0.28</td>
<td>0.10</td>
<td>0.18</td>
<td>0.63</td>
<td>0.31</td>
<td>0.31</td>
<td>0.31</td>
</tr>
<tr>
<td>5.8</td>
<td>0.10</td>
<td>0.23</td>
<td>0.08</td>
<td>0.12</td>
<td>0.75</td>
<td>0.18</td>
<td>0.39</td>
<td>0.31</td>
</tr>
<tr>
<td>6.1</td>
<td>0.13</td>
<td>0.32</td>
<td>0.12</td>
<td>0.19</td>
<td>0.62</td>
<td>0.19</td>
<td>0.74</td>
<td>0.10</td>
</tr>
<tr>
<td>7.2</td>
<td>0.09</td>
<td>0.26</td>
<td>0.07</td>
<td>0.23</td>
<td>0.71</td>
<td>0.71</td>
<td>0.19</td>
<td>0.19</td>
</tr>
<tr>
<td>7.8</td>
<td>0.10</td>
<td>0.32</td>
<td>0.12</td>
<td>0.15</td>
<td>0.61</td>
<td>0.15</td>
<td>0.74</td>
<td>0.10</td>
</tr>
<tr>
<td>8.4</td>
<td>0.11</td>
<td>0.32</td>
<td>0.11</td>
<td>0.21</td>
<td>0.74</td>
<td>0.21</td>
<td>0.74</td>
<td>0.10</td>
</tr>
<tr>
<td>8.5</td>
<td>0.11</td>
<td>0.31</td>
<td>0.10</td>
<td>0.25</td>
<td>0.84</td>
<td>0.21</td>
<td>0.74</td>
<td>0.10</td>
</tr>
<tr>
<td>8.8</td>
<td>0.13</td>
<td>0.35</td>
<td>0.14</td>
<td>0.25</td>
<td>0.86</td>
<td>0.25</td>
<td>0.74</td>
<td>0.10</td>
</tr>
<tr>
<td>9.0</td>
<td>0.11</td>
<td>0.31</td>
<td>0.09</td>
<td>0.26</td>
<td>0.83</td>
<td>0.26</td>
<td>0.74</td>
<td>0.10</td>
</tr>
<tr>
<td>9.4</td>
<td>0.11</td>
<td>0.30</td>
<td>0.10</td>
<td>0.25</td>
<td>0.88</td>
<td>0.25</td>
<td>0.74</td>
<td>0.10</td>
</tr>
<tr>
<td>11.1</td>
<td>0.12</td>
<td>0.34</td>
<td>0.11</td>
<td>0.22</td>
<td>0.89</td>
<td>0.22</td>
<td>0.74</td>
<td>0.10</td>
</tr>
<tr>
<td>11.7</td>
<td>0.15</td>
<td>0.36</td>
<td>0.10</td>
<td>0.25</td>
<td>0.90</td>
<td>0.25</td>
<td>0.74</td>
<td>0.10</td>
</tr>
<tr>
<td>13.3</td>
<td>0.15</td>
<td>0.32</td>
<td>0.12</td>
<td>0.29</td>
<td>0.79</td>
<td>0.29</td>
<td>0.74</td>
<td>0.10</td>
</tr>
<tr>
<td>14.3</td>
<td>0.16</td>
<td>0.36</td>
<td>0.13</td>
<td>0.43</td>
<td>0.80</td>
<td>0.43</td>
<td>0.74</td>
<td>0.10</td>
</tr>
<tr>
<td>14.5</td>
<td>0.16</td>
<td>0.38</td>
<td>0.14</td>
<td>0.43</td>
<td>0.80</td>
<td>0.43</td>
<td>0.74</td>
<td>0.10</td>
</tr>
<tr>
<td>15.0</td>
<td>0.18</td>
<td>0.42</td>
<td>0.19</td>
<td>0.58</td>
<td>0.78</td>
<td>0.58</td>
<td>0.78</td>
<td>0.10</td>
</tr>
<tr>
<td>15.7</td>
<td>0.18</td>
<td>0.45</td>
<td>0.17</td>
<td>0.58</td>
<td>0.78</td>
<td>0.58</td>
<td>0.78</td>
<td>0.10</td>
</tr>
<tr>
<td>19.9</td>
<td>0.21</td>
<td>0.51</td>
<td>0.21</td>
<td>0.79</td>
<td>0.79</td>
<td>0.79</td>
<td>0.79</td>
<td>0.10</td>
</tr>
<tr>
<td>17.4</td>
<td>0.28</td>
<td>0.62</td>
<td>0.22</td>
<td>0.67</td>
<td>0.70</td>
<td>0.67</td>
<td>0.70</td>
<td>0.10</td>
</tr>
<tr>
<td>18.8</td>
<td>0.22</td>
<td>0.55</td>
<td>0.22</td>
<td>0.67</td>
<td>0.70</td>
<td>0.67</td>
<td>0.70</td>
<td>0.10</td>
</tr>
<tr>
<td>19.8</td>
<td>0.38</td>
<td>0.64</td>
<td>0.47</td>
<td>1.12</td>
<td>0.80</td>
<td>1.12</td>
<td>0.80</td>
<td>0.10</td>
</tr>
<tr>
<td>22.6</td>
<td>0.29</td>
<td>0.73</td>
<td>0.33</td>
<td>1.52</td>
<td>0.83</td>
<td>1.52</td>
<td>0.83</td>
<td>0.10</td>
</tr>
<tr>
<td>23.7</td>
<td>0.39</td>
<td>0.61</td>
<td>0.21</td>
<td>1.67</td>
<td>0.83</td>
<td>1.67</td>
<td>0.83</td>
<td>0.10</td>
</tr>
</tbody>
</table>

Move the mouse over the numbers in the table to see the corresponding images. Click to compare with the ground truth.
Human-assisted motion annotation

- Ground truth is essential to progress the field
- A first step in computer vision to obtain ground-truth motion for arbitrary scenes (a different approach from Baker et al. 2007)
- An interactive system to combine human perception and the state-of-the-art computer vision algorithms to annotate motion

Liu et al. Human-assisted motion annotation. CVPR 2008
Demo: interactive layer segmentation
Demo: interactive motion labeling
Motion database of natural scenes

Content

• Robust optical flow estimation
• Applications
• Feature matching
• Discrete optical flow
• Layer motion analysis
• Other representations
Particle video

P. Sand and S. Teller. Particle Video: Long-Range Motion Estimation using Point Trajectories. CVPR 2006
Seemingly Simple Examples

Kanizsa square

From real video
Output from the State-of-the-Art Optical Flow Algorithm

Kanizsa square

Optical flow field

T. Brox et al. High accuracy optical flow estimation based on a theory for warping. ECCV 2004
Output from the State-of-the-Art Optical Flow Algorithm

T. Brox et al. High accuracy optical flow estimation based on a theory for warping. ECCV 2004
Optical flow representation: aperture problem
Optical flow representation: aperture problem

We need motion representation beyond pixel level!
Challenge: Textureless Objects under Occlusion

• Corners are not always trustworthy (junctions)

• Flat regions do not always move smoothly (discontinuous at illusory boundaries)

• How about boundaries?
  – Easy to detect and track for textureless objects
  – Able to handle junctions with illusory boundaries
Extracted boundary fragments
Optical flow from Lucas-Kanade algorithm
Estimated motion by our system, after grouping
Boundary grouping and illusory boundaries (frame 1)
Boundary grouping and illusory boundaries (frame 2)
Rotating Chair
Estimated flow field from Brox et al.
Estimated motion by our system, after grouping
Boundary grouping and illusory boundaries (frame 1)
Boundary grouping and illusory boundaries (frame 2)