Optical flow
Focus of expansion

J. J. Gibson
Optical flow due to camera motion

- Consider camera translating and rotating

\[ \mathbf{P} = (X, Y, Z)^T \]

\[ x = \frac{X}{Z} \quad y = \frac{Y}{Z} \]

\[ \dot{\mathbf{P}} = -\mathbf{t} - \omega \times \mathbf{P} \]
Optical flow due to camera motion

\[
\begin{bmatrix}
  u \\
  v
\end{bmatrix} = \begin{bmatrix}
  \dot{x} \\
  \dot{y}
\end{bmatrix} = \frac{1}{Z} \begin{bmatrix}
  -1 & 0 & x \\
  0 & -1 & y
\end{bmatrix} \begin{bmatrix}
  t_x \\
  t_y \\
  t_z
\end{bmatrix} + \begin{bmatrix}
  xy & -(1 + x^2) & y \\
  1 + y^2 & -xy & -x
\end{bmatrix} \begin{bmatrix}
  \omega_x \\
  \omega_y \\
  \omega_z
\end{bmatrix}
\]
Optical flow for moving scenes
Optical flow for moving scenes
Optical flow for moving scenes

- Optical flow helps *grouping*
- *Gestalt principle of common fate*
  - *Things that move together belong together*
Motion segmentation in humans

Motion segmentation in humans

Motion segmentation in humans

1. Collect videos
2. Segment using motion
3. Train ConvNet

Optical flow for moving scenes

- Motion is cue for recognition
  - Gestures, actions, ...

Optical flow for moving scenes

• Motion is cue for recognition
  • Gestures, actions, ...

<table>
<thead>
<tr>
<th>Model</th>
<th>Accuracy</th>
</tr>
</thead>
<tbody>
<tr>
<td>Without optical flow</td>
<td>73.0%</td>
</tr>
<tr>
<td>With optical flow</td>
<td>88.0%</td>
</tr>
</tbody>
</table>

Estimating optical flow

• Yet another correspondence problem!
• But:
  • Bad: scene can move
  • Good: changes are usually very small (often sub-pixel)
Optical flow constraint equation

- Image intensity *continuous* function of x, y, t
- In time dt, pixel (x,y,t) moves to (x + u dt, y + v dt, t + dt)

\[
\min_{u,v} (I(x + u \Delta t, y + v \Delta t, t + \Delta t) - I(x, y, t))^2
\]

\[
\equiv \min_{u,v} (I(x, y, t) + I_x u \Delta t + I_y v \Delta t + I_t \Delta t - I(x, y, t))^2
\]

\[
\equiv \min_{u,v} (I_x u \Delta t + I_y v \Delta t + I_t \Delta t)^2
\]

\[
I_x u + I_y v + I_t = 0
\]

- Optical flow constraint equation: One equation, two variables
Lucas-Kanade

- Assume all pixels in patch have the same flow
- When will this produce a unique solution?

\[
\begin{pmatrix}
\nabla I(x_1, y_1)^T \\
\vdots \\
\nabla I(x_n, y_n)^T \\
\end{pmatrix}
\begin{pmatrix}
u \\
v \\
\end{pmatrix}
= -
\begin{pmatrix}
I_t(x_1, y_1) \\
\vdots \\
I_t(x_n, y_n) \\
\end{pmatrix}
\]
Aperture problem
Aperture problem
Lucas-Kanade

\[
\begin{pmatrix}
\nabla I(x_1, y_1)^T \\
\vdots \\
\nabla I(x_n, y_n)^T
\end{pmatrix}
\begin{pmatrix}
u \\
v
\end{pmatrix}
= -
\begin{pmatrix}
I_t(x_1, y_1) \\
\vdots \\
I_t(x_n, y_n)
\end{pmatrix}
\]

- Equation of the form \( Ax = b \)
- Solve using Normal equations: \( x = (A^T A)^{-1} A^T b \)
- Need \( A^T A \) to be invertible - corners!
Lucas-Kanade

• What if we consider the whole image as one patch?
  • Constant optical flow for the entire image?

• Better: what if we consider flow as a \textit{parametric function} of pixel location?
  • e.g. affine

\[ \begin{bmatrix} u \\ v \end{bmatrix} = Ax + b \]

• More generally: \( \mathbf{x}' = W(\mathbf{x}; \mathbf{p}) \)
  • \( W \) is some 2D \( \rightarrow \) 2D parametric warp function
  • \( \mathbf{p} \) is a parameter vector

• “Motion models”
Lucas-Kanade

\[
\min_{\mathbf{p}} \sum_{\mathbf{x}} (I(W(x; \mathbf{p})) - T(x))^2
\]

- T is the previous frame, also called template
- I is the current frame
- Goal is to find \( \mathbf{p} \)

Lucas-Kanade

• Iterative process
• Assume that we have a current iterate \( p \) and we want to find the next iterate \( p + \Delta p \)
• Find \( \Delta p \) by optimizing \( \min_{\Delta p} \sum_{x} (I(W(x; p + \Delta p)) - T(x))^2 \)
• Hard because \( I \) and \( W \) are both non-linear
• Assume \( \Delta p \) is small and linearize:
  • Linearize \( W \): \( W(x; p + \Delta p) \approx W(x; p) + \frac{\partial W}{\partial p} \Delta p \)
  • Linearize \( I \): \( I(W(x; p + \Delta p)) \approx I(W(x; p)) + \nabla I \frac{\partial W}{\partial p} \Delta p \)

Lucas Kanade

• Iterative process
• At each step, find $\Delta p$ that optimizes

$$
\min_{\Delta p} \left( I(W(x; p)) + \nabla I \frac{\partial W}{\partial p} \Delta p - T(x) \right)^2
$$

• Warped image
• Gradient of warped image
• Jacobian of warp function
• Template

• Quadratic in $\Delta p$, solve exactly
Lucas-Kanade

- Solve by iterating on parameters
- Equivalent to Newton iteration + linearization
- Can we remove the parametric assumption

Horn-Schunk

\[ E(u, v) = E_{data}(u, v) + E_{smoothness}(u, v) \]

\[ E(u, v) = \int \int (I(x + u(x, y)\Delta t, y + v(x, y)\Delta t, t + \Delta t) - I(x, y, t))^2 \]
\[ + \alpha(\|\nabla u\|^2 + \|\nabla v\|^2) \, dx \, dy \]
Horn-Schunk

\[ E(u, v) = E_{data}(u, v) + E_{smoothness}(u, v) \]

\[ E(u, v) = \int \int (I_x u + I_y v + I_t)^2 + \alpha (\|\nabla u\|^2 + \|\nabla v\|^2) \, dx \, dy \]
Variational minimization

• $u$ and $v$ are functions
• Euler-lagrange equations
  • Similar to “gradient=0”

$$\min_q \int L(t, q(t), \dot{q}(dt)) dt$$

$$\frac{\partial L}{\partial q} - \frac{d}{dt} \frac{\partial L}{\partial \dot{q}} = 0$$
Variational minimization

\[
\min_q \int L(t, q(t), \dot{q}(dt))dt
\]

\[
\frac{\partial L}{\partial q} - \frac{d}{dt} \frac{\partial L}{\partial \dot{q}} = 0
\]

\[
\min_{u,v} \int \int f(x, y, u, v, u_x, u_y, v_x, v_y)dx dy
\]

\[
\frac{\partial f}{\partial u} - \frac{d}{dx} \frac{\partial f}{\partial u_x} - \frac{d}{dy} \frac{\partial f}{\partial u_y} = 0
\]

\[
\frac{\partial f}{\partial v} - \frac{d}{dx} \frac{\partial f}{\partial v_x} - \frac{d}{dy} \frac{\partial f}{\partial v_y} = 0
\]
MPI-Sintel

- Open-source animated movie “Sintel”
- “Naturalistic” video
- Ground truth optical flow
- Large motions
- Complex scenes

MPI-Sintel results
Optical flow with large displacements

• Optical flow constraint equation assumes differential optical flow
• “Large displacement”? 
• Key idea: reducing resolution reduces displacement
• Reduce resolution, then upsample?
  • will lose fine details
Optical flow with large displacements

- Key idea 2: Use upsampled flow as *initialization*
- *Changes to initialization will be infinitesimal*

Optical flow for large displacements

• Possible issue: large appearance change => incorrect matching based on color alone
• Use descriptor matching (e.g. SIFT) on sparse points
• Use smoothness to propagate to all pixels:
  • Flow is weighted average of nearest neighbors: \( F(p) = \frac{\sum_q k(p,q)F(q)}{\sum_q k(p,q)} \)
  • *Kernel* \( k \) dependent on relative position + edges
• Use this as initialization for optimization

LDOF and EpicFlow

Video

Basic Horn-Schunk (Error = 2.069)

LDOF (Brox et al, 2009) (Error = 1.606)

EpicFlow (Revaud et al, 2015) (Error = 1.295)
Coarse-to-fine processing

• A specific instance of a general idea
  • Also coarse-to-fine versions of Lucas-Kanade

• Coarse scales:
  • Global / large structures
  • Long-range relationships
  • But: imprecise localization

• Fine scales:
  • Precise localization
  • But: aperture problem

• Idea: start from coarse scales, add fine scale detail
Coarse-to-fine processing