Computational Photography

Prof. Feng Liu

Spring 2020

http://www.cs.pdx.edu/~fliu/courses/cs510/

05/14/2020
Last Time

- Compositing and Matting
Today

- Video Stabilization
  - Video stabilization pipeline
A Tracking Shot

Orson Welles, *Touch of Evil*, 1958
Images courtesy Peter Sand and Flickr user Charles W. Brown
Traditional 2D Video Stabilization Result
3D Video Stabilization Result [Liu et al. 09]
Stabilization: An Old Problem

- iMovie from Apple
- De-shaker, a free tool
- Most modern camcorders
Video Stabilization Pipeline

Input → Trajectory Estimation → Motion Model Estimation → Motion Plan → Video Transform → Output
Video Stabilization Pipeline

Input → Trajectory Estimation → Motion Model Estimation → Motion Plan → Video Transform → Output

Feature trajectories
Video Stabilization Pipeline

Input → Trajectory Estimation → Motion Model Estimation → Motion Plan → Video Transform → Output

Feature trajectories
Trajectory Estimation

- Kanade-Lucas-Tomasi feature tracker (KLT)

- Implementations
  - OpenCV
  - http://www.ces.clemson.edu/~stb/klt/
  - ...

Feature Tracking

Brightness Constancy Equation:

\[ I(x, y, t - 1) = I(x + u(x, y), y + v(x, y), t) \]

Linearizing the right side using Taylor expansion:

\[ I(x, y, t - 1) \approx I(x, y, t) + I_x \cdot u(x, y) + I_y \cdot v(x, y) \]

Hence,

\[ I_x \cdot u + I_y \cdot v + I_t \approx 0 \]

Spatial Coherence Constraint

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

- How many equations and unknowns per pixel?
  - One equation, two unknowns

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

\[
0 = I_t(p_i) + \nabla I(p_i) \cdot [u \ v]
\]

\[
\begin{bmatrix}
I_x(p_1) & I_y(p_1) \\
I_x(p_2) & I_y(p_2) \\
\vdots & \vdots \\
I_x(p_{25}) & I_y(p_{25})
\end{bmatrix}
\begin{bmatrix}
u \\
v
\end{bmatrix}
= -
\begin{bmatrix}
I_t(p_1) \\
I_t(p_2) \\
\vdots \\
I_t(p_{25})
\end{bmatrix}
\]
Solving the Tracking Problem

- Least squares problem:

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

- When is this system solvable?
  - What if the window contains just a single straight edge?

Conditions for Solvability

- “Bad” case: single straight edge
Lucas-Kanade Flow

- Least squares problem:

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

Solution given by

\[
(A^T A) \quad d = A^T b
\]

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

The summations are over all pixels in the window.

Lucas-Kanade Flow

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

\[ A^T A \quad A^T b \]

- Recall the Harris corner detector: $M = A^T A$ is the *second moment matrix*.

- We can figure out whether the system is solvable by looking at the eigenvalues of the second moment matrix:
  - The eigenvectors and eigenvalues of $M$ relate to edge direction and magnitude.
  - The eigenvector associated with the larger eigenvalue points in the direction of fastest intensity change, and the other eigenvector is orthogonal to it.
Interpreting the eigenvalues

Classification of image points using eigenvalues of the second moment matrix:

- **“Corner”**
  - $\lambda_1$ and $\lambda_2$ are large,
  - $\lambda_1 \sim \lambda_2$

- **“Edge”**
  - $\lambda_1 >> \lambda_2$
  - $\lambda_2 >> \lambda_1$

- **“Flat”** region
  - $\lambda_1$ and $\lambda_2$ are small
Uniform Region

- gradients have small magnitude
- small $\lambda_1$, small $\lambda_2$
- system is ill-conditioned
Edge

- gradients have one dominant direction
- large $\lambda_1$, small $\lambda_2$
- system is ill-conditioned
High-texture or Corner Region

- gradients have different directions, large magnitudes
- large $\lambda_1$, large $\lambda_2$
- system is well-conditioned
Feature tracking

• So far, we have only considered feature tracking in a pair of images
• If we have more than two images, we can track feature from each frame to the next
• Given a point in the first image, we can in principle reconstruct its path by simply “following the arrows”
Tracking over Many Frames

- Select features in first frame
- For each frame:
  - Update positions of tracked features
    - Discrete search or Lucas-Kanade (or a combination of the two)
  - Terminate inconsistent tracks
    - Compute similarity with corresponding feature in the previous frame or in the first frame where it’s visible
  - Find more features to track
Shi-Tomasi Feature Tracker

- Find good features using eigenvalues of second-moment matrix
  - Key idea: “good” features to track are the ones whose motion can be estimated reliably

- From frame to frame, track with Lucas-Kanade
  - This amounts to assuming a translation model for frame-to-frame feature movement

- Check consistency of tracks by affine registration to the first observed instance of the feature
  - Affine model is more accurate for larger displacements
  - Comparing to the first frame helps to minimize drift

Traditional 2D Video Stabilization

Input → Trajectory Estimation → Motion Model Estimation → Motion Plan → Video Transform → Output

2D motion model (homography)
Homography

\[ \lambda x'_i = T x_i \]

expand

\[ \begin{bmatrix} x'_i \\ y'_i \\ 1 \end{bmatrix} = \begin{bmatrix} h_{11} & h_{12} & h_{13} \\ h_{21} & h_{22} & h_{23} \\ h_{31} & h_{32} & h_{33} \end{bmatrix} \begin{bmatrix} x_i \\ y_i \\ 1 \end{bmatrix} \]
Fitting a homography

• Equation for homography:

\[
\lambda \begin{bmatrix}
    x'_i \\
y'_i \\
1
\end{bmatrix} = \begin{bmatrix}
h_{11} & h_{12} & h_{13} \\
h_{21} & h_{22} & h_{23} \\
h_{31} & h_{32} & h_{33}
\end{bmatrix} \begin{bmatrix}
x_i \\
y_i \\
1
\end{bmatrix}
\]

\[
\lambda x'_i = Tx_i
\]

\[
x'_i \times Tx_i = 0
\]

\[
\begin{bmatrix}
x'_i \\
y'_i \\
1
\end{bmatrix} \times \begin{bmatrix}
h_1^T x_i \\
h_2^T x_i \\
h_3^T x_i
\end{bmatrix} = \begin{bmatrix}
y_i' h_3^T x_i - h_2^T x_i \\
h_1^T x_i - x_i' h_3^T x_i \\
x_i' h_2^T x_i - y_i' h_1^T x_i
\end{bmatrix}
\]

\[
\begin{bmatrix}
0^T & -x_i^T & y_i' x_i^T \\
x_i^T & 0^T & -x_i' x_i^T \\
-y_i' x_i^T & x_i' x_i^T & 0^T
\end{bmatrix} \begin{bmatrix}
h_1 \\
h_2 \\
h_3
\end{bmatrix} = 0
\]

3 equations, only 2 linearly independent
Direct linear transform

\[
\begin{bmatrix}
0^T & x_1^T & -y_1' x_1^T \\
0^T & x_1^T & -y_1' x_1^T \\
\vdots & \vdots & \vdots \\
0^T & x_n^T & -y_n' x_n^T \\
x_n^T & 0^T & -x_n' x_n^T
\end{bmatrix}
\begin{pmatrix}
h_1 \\
h_2 \\
h_3
\end{pmatrix} = 0 \\
A h = 0
\]

- H has 8 degrees of freedom (9 parameters, but scale is arbitrary)
- One match gives us two linearly independent equations
- Four matches needed for a minimal solution (null space of 8x9 matrix)
- More than four: homogeneous least squares
Traditional 2D Video Stabilization

1. Input
2. Trajectory Estimation
3. Motion Model Estimation
4. Motion Plan
5. Video Transform
6. Output
Motion Plan

\[ S_t = \sum_{i \in N_t} T_t^i \ast G, \text{ where } T_t^i = \prod_{j=i}^{t} T_j \]
Traditional 2D Video Stabilization Result
Limitations

- No knowledge of actual 3D camera path, so cannot control desired motion directly

- Homography cannot model 3D camera motion and scene structure
3D Video Stabilization

- Non-metric image-based rendering for video stabilization [Buehler et al. 01]
- Image-based rendering using image-based priors [Fitzgibbon et al. 05]
- Using photographs to enhance videos of a static scene [Bhat et al. 07]
3D Video Stabilization

Input → Trajectory Estimation → Motion Model Estimation → Motion Plan → Video Transform → Output

3D reconstruction via structure from motion

Voodoo Camera Tracker (http://www.digilab.uni-hannover.de)
Structure from Motion

Voodoo Camera Tracker (http://www.digilab.uni-hannover.de)
3D Video Stabilization

Input \rightarrow Trajectory Estimation \rightarrow Motion Model Estimation \rightarrow Motion Plan \rightarrow Video Transform \rightarrow Output

- Line
- Parabola
- Low-pass filter
Novel View Synthesis by Image based Rendering

Unstructured lumigraph rendering [Buehler et al. 01]
Content-preserving warps based 3D video stabilization

F Liu, M Gleicher, H Jin, A Agarwala. Content-preserving warps for 3D video stabilization, SIGGRAPH’ 09
3D Video Stabilization

Input → Trajectory Estimation → Motion Mode Estimation → Motion Plan → Video Transform → Output

Novel view synthesis

- image based rendering
Our method for novel view synthesis

Temporal Constraint

Input

Trajectory Estimation

Motion Mode Estimation

Motion Plan

Video Transform

Output

One input frame

One output frame
Novel View from One Frame

- A Series of Vision Challenges!
  - Segment out layers
  - Determine depth
  - Shift and re-composite layers
  - Fill holes

- Cannot achieve accurate dis-occlusions, non-Lambertian reflection, etc.
Human Perception

- Viewpoint shifts will be small
- Aim for perceptual plausibility rather than accurate novel view synthesis
  - Move salient content along stabilized paths
  - No noticeable artifacts
Problem Setup

input frame and points
Problem Setup

input frame and points                         output points
Problem Setup

input frame and points

output frame
Option 1: Scattered Data Interpolation
Option 2: Full-frame Warping with Homography
A Less Successful Result
Our Approach: Content-preserving Warping

Warp each input frame to create the output frame by least-squares minimization

- Data term: **Soft, sparse displacement constraint**
- Smoothness term: **Local similarity transformation constraint**
Smoothness Term: Minimize Visual Distortion

Local similarity transformation constraint
Smoothness Term: Minimize Visual Distortion

Local similarity transformation constraint

[Igarashi et al. 05]
Visual saliency: “the distinct subjective perceptual quality which makes some items in the world stand out from their neighbors and immediately grab our attention” from [Itti 07]
Content-Preserving Warping

Input

Output
Content-Preserving Warping

 Input                                           Output

 texture mapping [Shirley et al. 2005]
Content-Preserving Warping

Grid mesh & points

Output
Student Paper Presentations

- Presenter: Kwong, Marcus
  - Intelligent scissors for image composition
    E. Mortensen and W. Barrett
    SIGGRAPH 1995

- Presenter: Lajara, Armando
  - Learning to See in the Dark
    C. Chen, Q. Chen, J. Xu and V. Koltun
    IEEE CVPR 2018
Next Time

☐ More on Video stabilization

☐ Student paper presentations

■ 05/19: Mauck, Milan
  ☐ Video SnapCut: Robust Video Object Cutout Using Localized Classifiers
    X. Bai, J. Wang, D. Simons, G. Sapiro
    SIGGRAPH 2009

■ 05/19: Mudgal, Priyanka
  ☐ A global sampling method for alpha matting
    K. He, C. Rhemann, C. Rother, X. Tang, and J. Sun
    CVPR 2011

■ 05/21: Niedermeyer, Jon
  ☐ A Closed Form Solution to Natural Image Matting
    A. Levin, D. Lischinski, and Y. Weiss
    CVPR 2006

■ 05/21: Parker-Durost, Madison
  ☐ Full-Frame Video Stabilization with Motion Inpainting
    Y. Matsushita, E. Ofek, W. Ge, Xi. Tang, and H. Shum. IEEE PAMI 2006