Visual Information Processing

Menandro Roxas
Computer Vision Laboratory
November 07, 2018
Today’s Topic

• Spatio-Temporal Image Processing / Video Processing
  • 3D reconstruction
  • Video Modification/Restoration
3D Reconstruction

• Model Based Methods
  • Pose Estimation
  • PTAM
  • DTAM
  • LSD-SLAM
  • SOF3D

• Learning Based Methods
  • CNN-SLAM
  • DeMoN
Camera Pose Estimation

• Two cases

2D-2D

3D-2D
2D-2D Pose Estimation

• Find correspondences \( x, x' \)
• Solve for the fundamental matrix \( F(3 \times 3) \)
  \[ x'^T F x = 0 \]
• Extract the essential matrix \( E \)
  \[ E = K'^TFK \]
• Decompose \( E \) to find relative camera rotation \( R \) and translation \( t \)
  \[ E = [t]_\times R \]

• Scale is relative \( |t| = 1 \)

\( K = \) camera intrinsic parameters
3D-2D Pose Estimation

- (PnP or Perspective n-point problem)
- Camera projection model

\[
\begin{bmatrix}
u \\
v \\
1
\end{bmatrix} = K[R|t]\begin{bmatrix}
X \\
Y \\
Z \\
1
\end{bmatrix}
\]

\[
\begin{bmatrix}
u \\
v \\
1
\end{bmatrix} = \begin{bmatrix}
f_x & \gamma & u_0 \\
0 & f_y & v_0 \\
0 & 0 & 1
\end{bmatrix}\begin{bmatrix}
r_{11} & r_{12} & r_{13} & t_1 \\
r_{21} & r_{22} & r_{23} & t_2 \\
r_{31} & r_{32} & r_{33} & t_3
\end{bmatrix}\begin{bmatrix}
X \\
Y \\
Z \\
1
\end{bmatrix}
\]

- Calibrated (K = known) and Uncalibrated (K = unknown)
- Minimal case: solution for P3P or n=3 (n correspondences)
PTAM
(Parallel Tracking and Mapping)
PTAM

A comparison between PTAM and frame-by-frame algorithms (2)
PTAM
Parallel Tracking and Mapping for Small AR Workspaces

ISMAR 2007 video results

Georg Klein and David Murray
Active Vision Laboratory
University of Oxford
DTAM: Dense Tracking and Mapping in Real-Time
DTAM
(Dense Tracking and Mapping)

Live Dense Reconstruction with a Single Moving Camera
LSD-SLAM
(Large-Scale Direct Monocular SLAM)
LSD-SLAM

**Feature-Based**
- Input Images
- Extract & Match Features (SIFT / SURF / ...)
- Abstract image to feature observations
- Track: min. reprojection error (point distances)
- Map: est. feature-parameters (3D points / normals)

**Direct**
- Input Images
- Keep full images (no abstraction)
- Track: min. photometric error (intensity differences)
- Map: est. per-pixel depth (semi-dense depth map)
Feature-Based Alignment vs. Direct Alignment

Uses point correspondence and 3D point differences

Uses intensity difference

Failure: Non-textured and repeating patterns

Failure: Large homography
LSD-SLAM

We present LSD-SLAM: a Large-Scale Direct monocular SLAM system, which allows to build large-scale semi-dense maps in real-time on a CPU.
Scale Ambiguity

Image 1 Image 2

3D

Image 1 Image 2

camera center

3D

camera center
CNN-SLAM
(Dense SLAM with learned depth prediction)

Figure 2. CNN-SLAM Overview.
CNN-SLAM
DeMoN
(Learning Monocular Stereo)
**DeMoN**

Figure 2. Overview of the architecture. DeMoN takes an image pair as input and predicts the depth map of the first image and the relative pose of the second camera. The network consists of a chain of encoder-decoder networks that iterate over optical flow, depth, and egomotion estimation; see Fig. 3 for details. The refinement network increases the resolution of the final depth map.

Figure 3. Schematic representation of the encoder-decoder pair used in the bootstrapping and iterative network. Inputs with gray font are only available for the iterative network. The first encoder-decoder predicts optical flow and its confidence from an image pair and previous estimates. The second encoder-decoder predicts the depth map and surface normals. A fully connected network appended to the encoder estimates the camera motion \( \mathbf{r}, \mathbf{t} \) and a depth scale factor \( s \). The scale factor \( s \) relates the scale of the depth values to the camera motion.
DeMoN: Depth and Motion Network for Learning Monocular Stereo

Benjamin Ummenhofer\textsuperscript{*1}  Huizhong Zhou\textsuperscript{*1}  Jonas Uhrig\textsuperscript{1,2}
Nikolaus Mayer\textsuperscript{1}  Eddy Ilg\textsuperscript{1}  Alexey Dosovitskiy\textsuperscript{1}  Thomas Brox\textsuperscript{1}

\textsuperscript{1}University of Freiburg  \textsuperscript{2}Daimler AG R&D

\textsuperscript{*equal contribution}
SOF3D – Simultaneous Optical Flow Estimation and 3D Reconstruction
Monocular Stereo

• Find the depth map using two images (assuming known camera poses)

• Relative camera pose defines a 2D search space
Issue

• **Disparity error caused by pose error**

\[
\begin{align*}
\text{actual pose} & \quad \text{est. pose} \\
\theta = 0^\circ & \quad \theta = 5^\circ \\
\end{align*}
\]

Input Frame 1

Ground Truth 2D Disparity

Error = 5.56%

Error = 32.43%

[Graber, CVPR 2015]
Problem

- Pose error causes wrong search space
  - Pose cannot be refined with just two frames (no bundle adjustment)

- 2D-3D hard constraint in Variational Stereo methods
  - \( \arg \min_x F(X) \)
  - \( X = f(u) \) -> hard constraint
Solution

• Relax disparity estimation by decoupling depth \((X)\) and correspondence \((u)\)

• Proposed decoupled method
  \[
  \arg \min_{u,X} G(u) + F(u,X)
  \]
  \(G\): optical flow constraints
  \(F\): 3D constraints
Optical flow and 3D constraints

• Optical Flow Constraints
  \[ G(u) = G_I(I' (x + u) + I(x)) + G_{tv}(u) + \alpha \|u - u_{fn}\|^2 \]
  TV-L1 [Wedel et al. 2008]

• 3D Constraints
  \[ F(u, X) = F_{data}(u, X) + F_{ms}(X) \]

  Reprojection error
  Minimal Surface Regularizer
  FlowNet2 (Large Displacement) [Ilg et al. 2017]
  [Grabber et al. 2015]
Optimization

• Minimize the energy
  \[ \arg \min_{u,X} G(u) + F(u, X) \]

• Decouple \( u \) from optical flow and 3D constraints
  \[ \arg \min_{u,u_{pj},X} G(u) + F(u_{pj}, X) + \lambda \| u - u_{pj} \|^2 \]
Optimization

• Solve equation using single loop, double split \((u_{tv}, u_{pj})\)
  ADMM (alternating direction multiplier method)

\[
\arg \min_{u, u_{pj}, X} G(u) + F(u_{pj}, X) + \lambda \|u - u_{pj}\|^2
\]

Iteration of following steps
1. Solving for \(u\) while holding \(X, u_{pj}\) constant
2. Solving for \(u_{pj}\) while holding \(X, u\) constant
3. Solving for \(X\) while holding \(u_{pj}\) constant
Real-Time implementation

• GTX 1080 GPUs x 2
  • 1 running FlowNet2-CSS
  • 1 running our method

• Image size (1024x512)
  • FlowNet2-CSS = 51ms
  • Our method = 41ms
  • Total = 92ms (10.8 fps)
Depth estimation (KITTI2012)
Depth estimation (ETH3D)
Depth estimation (Middlebury)

Frame 1

GT Depth

Graber

Ours
Comparison (Depth estimation)

- Percentage of erroneous pixels, $\tau$

<table>
<thead>
<tr>
<th></th>
<th>Graber $\tau &gt; 1$</th>
<th>Graber $\tau &gt; 3$</th>
<th>Ours $\tau &gt; 1$</th>
<th>Ours $\tau &gt; 3$</th>
</tr>
</thead>
<tbody>
<tr>
<td>KITTI2012</td>
<td>-</td>
<td>46.04</td>
<td>-</td>
<td>14.24</td>
</tr>
<tr>
<td>ETH3D</td>
<td>57.84</td>
<td>45.40</td>
<td><strong>12.81</strong></td>
<td><strong>3.49</strong></td>
</tr>
<tr>
<td>Middlebury</td>
<td>57.87</td>
<td>44.41</td>
<td><strong>42.06</strong></td>
<td><strong>20.19</strong></td>
</tr>
</tbody>
</table>
Pose error/decoupling weight (lambda)
Video Result
Video Modification/Restoration

- Video Completion
  - Wexler, Irani
  - Shiratori
  - Roxas

- Video Stabilization
  - Matsushita
Video Completion, Wexler
Video Completion, Wexler
Video Completion, Wexler
Video Completion, Wexler
Video Completion, Wexler
Video Completion, Wexler
Video Completion, Shiratori
Video Completion, Shiratori
Video Completion, Shiratori
Remaining Issue

• Temporal aliasing or “ghost” exists

• Why do ‘ghosts’ exist?
  • Discontinuity in motion of objects
  • Inconsistency between the hole and its boundary
Video Completion, Roxas

iterative step

input

trajectory prior estimation

image pyramid

motion estimation and inpainting

color propagation

update mask function

output
Three-frame Optical Flow

$$\min_{u_f, u_b} E_{data} + E_{spatial} + E_{trajectory}$$

$$E_{data} = \mu(x)f(I, u_f, u_b)$$  \( f = \text{brightness constancy} \)

$$E_{spatial} = \lambda_s TV(u_f, u_b)$$  \( \mu = \text{mask} \)

$$E_{trajectory} = \lambda_t \varphi(u_f, u_b)$$  \( TV = \text{total variation} \)

\( \varphi = \text{trajectory} \)

backward flow, \( u_b \)

forward flow, \( u_f \)
Color Propagation

\[ \mu(x_1) = 1 \]
\[ \mu(x_2) = 2 \]

distance of the source to frame 0
Color Propagation

• $\mu(x)$:

source frame distance from one optical flow direction
Color Propagation

input frame

using one direction

INCONSISTENCY!
Color Propagation

- Issue: two optical flow direction (forward and backward)
- How to choose between the two? -> Blend both directions using $\mu(x)$
Color Propagation

- Input frame using one direction
- Using one direction
- Using both directions without blending
- Blending both directions

INCONSISTENCY!
Results

Input Sequence

Ground Truth Optical Flow

Inpainted Video

Solved and Inpainted Optical Flow
Results
Results
Omnidirectional Camera
Another View
Video Stabilization, Matsushita

Stabilized sequence

Trimming

Mosaicing

Our method
Video Stabilization, Matsushita

1. Input Video
   - Global Motion Estimation
     - Motion Smoothing
       - Local Motion Estimation
         - Motion Inpainting
           - Image Deblurring
             - Completion
               - Output Video

   - Image with missing image area
     - Local motion computation
       - Mosaicing with local warping
         - Motion Inpainting
Video Stabilization, Matsushita
Video Stabilization, Matsushita
Removing Raindrops from Videos

Fig. 1. An example of the results of our proposed detection and removal method. (a) Scene with raindrop. (b) Dense long trajectories. (c) Matching of trajectories occluded by raindrop. (d) Trajectory based video completion.

Fig. 2. The pipeline of our method.
Removing Raindrops from Videos
Papers

• PTAM
  • Georg Klein and David Murray, "Improving the Agility of Keyframe-based SLAM", Proc. ECCV 2008

• DTAM

• LSD-SLAM

• CNN-SLAM

• DeMoN

• Sof3D
Papers

• Video Completion, Wexler
  • Y. Wexler, et al. “Space-time video completion”, CVPR 2004

• Video Completion, Shiratori

• Video Completion, Roxas

• Video Stabilization, Matsushita

• Removing Raindrops, You