Recovering Image Correspondence: New Methods and Applications

Sudipta N. Sinha
Microsoft Research
Redmond, USA

NII Shonan Meeting on Optimization Methods in Geometric Vision,
January 28, 2019
Overview

- Dense stereo matching
  - Optimization via Semi Global Matching (SGM)
  - Two extensions to SGM

- Learning to align images from scratch
  - Joint framework for local feature descriptor learning and image alignment
  - Application: RGB / NIR image registration
Semi Global Matching (SGM)

[Hirschmüller 2005]

- **Motivation:** Markov Random Field (MRF) inference via Graph Cuts, BP etc. is too slow and approximate. So why not approximate even more.
- SGM is parallelizable; runs on GPUs and FPGAs.
- Widely used: assisted driving, robotics, aerial mapping.
Semi Global Matching (SGM) [Hirschmüller 2005]

- Solve several independent 1D scanline optimization problems; one for each of 4 or 8 directions.

\[
L_r(p, d) = C_p(d) + \min_{d' \in D} (L_r(p - r, d') + V(d, d')).
\]
Semi Global Matching (SGM) [Hirschmüller 2005]

- Solve several independent 1D scanline optimization problems; one for each of 4 or 8 directions.

\[ L_r(p, d) = C_p(d) + \min_{d' \in D} (L_r(p - r, d') + V(d, d')) \]

- Sum the costs and select min cost disparity at each pixels.

\[ S(p, d) = \sum_r L_r(p, d) \]

\[ D_p = \arg\min_d S(p, d) \]
Two Limitations of SGM

- Fronto-parallel bias due to pairwise smoothness term; leads to errors on slanted textureless surfaces.

\[
V(d, d') = \begin{cases} 
0 & \text{if } d = d' \\
P_1 & \text{if } |d - d'| = 1 \\
P_2 & \text{if } |d - d'| \geq 2 
\end{cases}
\]
Two Limitations of SGM

- Fronto-parallel bias due to pairwise smoothness term; leads to errors on slanted textureless surfaces.

\[ V(d, d') = \begin{cases} 
0 & \text{if } d = d' \\
P_1 & \text{if } |d - d'| = 1 \\
P_2 & \text{if } |d - d'| \geq 2 
\end{cases} \]

- Summing up costs and picking the best disparity (last two steps lack proper justification)

\[ L_r(p, d) = C_p(d) + \min_{d' \in \mathcal{D}} (L_r(p - r, d') + V(d, d')) \]

\[ S(p, d) = \sum_r L_r(p, d) \]

\[ D_p = \arg \min_d S(p, d). \]
SGM with Surface Orientation Prior

[Scharstein, Taniai, Sinha, 3DV 2017]

- If we knew the surface slant, we can replace the fronto-parallel bias with bias parallel to surface.

- Approach:
  - *Fit surfaces (planes) to an initial depth map.*
  - *Alternatively, integrate a given surface normal map.*
  - *Discretize* disparity surface and record pixels where the disparity “changes” (+/- 1).
  - During optimization, bias pairwise terms at those pixels.
SGM with Surface Orientation Prior

[Scharstein, Taniai, Sinha, 3DV 2017]

SGM-EP
Low-resolution stereo matching
+ Plane fitting

SGM-GS
Ground truth oracle
SGM with Surface Orientation Prior

- Pairwise Terms.
  - SGM
    \[
    V(d, d') = \begin{cases} 
    0 & \text{if } d = d' \\
    P_1 & \text{if } |d - d'| = 1 \\
    P_2 & \text{if } |d - d'| \geq 2 
    \end{cases}
    \]
  - SGM-P (2D Prior)
    \[
    V_S(d_p, d'_p) = V(d_p + \hat{j}_p, d'_p)
    \]
    \(\hat{j}_p \in \{-1, 0, +1\}\)
  - SGM-P (3D Prior)
    \[
    V_S(d_p, d'_p) = V(d_p + \hat{j}_p(d_p), d'_p)
    \]
Results

Percentage of pixels with Disparity Error > 2.0

% Error Reduction over SGM
Conclusions

- Huge accuracy boost in scenes with slanted untextured surfaces.
- Soft constraint; inaccurate normals don’t hurt accuracy.
- 2D prior version adds minimal computational overhead.
- Accurate estimation of surface orientation can be difficult.
Learning to Fuse Proposals in SGM

[Schoenberger, Sinha and Pollefeys, ECCV 2018]

SGM steps:

1. \( L_r(p, d) = C_p(d) + \min_{d' \in D} (L_r(p - r, d') + V(d, d')) \).

2. \( S(p, d) = \sum_r L_r(p, d) \)

3. \( D_p = \arg \min_d S(p, d) \).

Main Idea:

- Replace steps 2 and 3 with a learned predictor.
- The predictor takes disparity maps obtained via scanline optimization and directly estimates the final disparity map.
Learning to Fuse Proposals in SGM

[Schoenberger, Sinha and Pollefeys, ECCV 2018]

Motivation

“Best of k directions” oracle is significantly better than SGM.
Learning to Fuse Proposals in SGM
[Schoenberger, Sinha and Pollefeys, ECCV 2018]

- **Approach (SGM Forest):**
  1. Run SO to get $k$ disparity map proposals
  2. At each pixel
     - Construct *per-pixel* feature vector (*see next slide*)
     - Pick best disparity using a random forest classifier
     - Forest outputs probabilities
  3. Post-processing using probability map
Computing Per-pixel Features

\[
\text{Sparsely sample the cost volume: } \quad f = [d_1, d_2, d_3, c_{11}, c_{12}, c_{13}, c_{21}, c_{22}, c_{23}, c_{31}, c_{32}, c_{33}]
\]
**SGM Forest: Results**

<table>
<thead>
<tr>
<th>Datasize</th>
<th>Method</th>
<th>Middlebury 2014</th>
<th></th>
<th>KITTI 2015</th>
<th></th>
<th>ETH3D 2017</th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td></td>
<td>0.5px 1px 2px 4px</td>
<td>all</td>
<td>0.5px 1px 2px 4px</td>
<td>all</td>
<td>0.5px 1px 2px 4px</td>
<td>all</td>
</tr>
<tr>
<td>NCC</td>
<td>SGM</td>
<td>69.23 42.36 27.96 22.25</td>
<td></td>
<td>60.59 33.79 15.06 8.34</td>
<td></td>
<td>32.52 16.71 10.66 7.69</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SGM-F.</td>
<td>64.00 37.22 22.85 17.09</td>
<td></td>
<td>52.39 25.80 10.11 4.69</td>
<td></td>
<td>22.48 11.26 6.36 4.35</td>
<td></td>
</tr>
<tr>
<td>MC-CNN-fast</td>
<td>SGM</td>
<td>65.82 36.22 21.98 17.47</td>
<td></td>
<td>58.48 31.39 13.30 7.02</td>
<td></td>
<td>26.34 10.50 6.13 4.52</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SGM-F.</td>
<td>62.04 32.96 18.22 13.16</td>
<td></td>
<td>51.03 24.05 8.73 3.78</td>
<td></td>
<td>17.62 7.17 3.66 2.51</td>
<td></td>
</tr>
<tr>
<td></td>
<td>SGM-F.</td>
<td>59.20 30.58 16.57 11.62</td>
<td></td>
<td>46.88 19.77 6.51 2.97</td>
<td></td>
<td>27.40 11.89 7.30 5.52</td>
<td></td>
</tr>
</tbody>
</table>

MC-CNN [Zbontar and Lecun 2015]

- SGM-Forest consistently outperforms standard SGM and prior SGM variants.
SGM Forest: Ablation Study

- Excellent cross-dataset generalization.
- Model trained on 2005-06 data shows large accuracy gain on the significantly harder Middlebury 2014 scenes.
- Forest learns abstract patterns in the DSI; not in the input images.
SGM Forest: Benchmark Results

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>non-ocl.</td>
<td>all</td>
</tr>
<tr>
<td>LocalExp</td>
<td>5.43%</td>
<td>#1</td>
</tr>
<tr>
<td>3DMST</td>
<td>5.92%</td>
<td>#2</td>
</tr>
<tr>
<td>MC-CNN+TDSR</td>
<td>6.35%</td>
<td>#2</td>
</tr>
<tr>
<td>PMSC</td>
<td>6.71%</td>
<td>#4</td>
</tr>
<tr>
<td>LW-CNN</td>
<td>7.04%</td>
<td>#5</td>
</tr>
<tr>
<td>MeshStereoExt</td>
<td>7.08%</td>
<td>#6</td>
</tr>
<tr>
<td>FEN-D2DRR</td>
<td>7.23%</td>
<td>#7</td>
</tr>
<tr>
<td>APAP-Stereo</td>
<td>7.26%</td>
<td>#8</td>
</tr>
<tr>
<td>SGM-Forest</td>
<td>7.37%</td>
<td>#9</td>
</tr>
<tr>
<td>NTDE</td>
<td>7.44%</td>
<td>#10</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th></th>
<th></th>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>non-ocl.</td>
<td>all</td>
</tr>
<tr>
<td>LocalExp</td>
<td>6.52%</td>
<td>#1</td>
</tr>
<tr>
<td>3DMST</td>
<td>7.08%</td>
<td>#2</td>
</tr>
<tr>
<td>APAP-Stereo</td>
<td>7.53%</td>
<td>#3</td>
</tr>
<tr>
<td>FEN-D2DRR</td>
<td>7.89%</td>
<td>#4</td>
</tr>
<tr>
<td>MC-CNN-acrt</td>
<td>10.1%</td>
<td>#12</td>
</tr>
<tr>
<td>SGM-Forest</td>
<td>11.1%</td>
<td>#19</td>
</tr>
<tr>
<td>NTDE</td>
<td>11.7%</td>
<td>#21</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>KITTI 2015</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Error</td>
</tr>
<tr>
<td>CNNF+SGM</td>
<td>3.60%</td>
</tr>
<tr>
<td>SGM-Net</td>
<td>3.66%</td>
</tr>
<tr>
<td>MC-CNN-acrt</td>
<td>3.89%</td>
</tr>
<tr>
<td>SGM-Forest</td>
<td>4.38%</td>
</tr>
<tr>
<td>MC-CNN-WS</td>
<td>4.97%</td>
</tr>
<tr>
<td>SGM_ROB [17]</td>
<td>6.38%</td>
</tr>
<tr>
<td>SGM+C+NL</td>
<td>6.84%</td>
</tr>
<tr>
<td>SGM+LDOF</td>
<td>6.84%</td>
</tr>
<tr>
<td>SGM+SF</td>
<td>6.84%</td>
</tr>
<tr>
<td>CSCT+SGM+MF</td>
<td>8.24%</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Method</th>
<th>ETH3D 2017</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>non-ocl.</td>
</tr>
<tr>
<td>SGM-Forest</td>
<td>5.40%</td>
</tr>
<tr>
<td>SGM_ROB [17]</td>
<td>10.08%</td>
</tr>
<tr>
<td>MeshStereo</td>
<td>11.94%</td>
</tr>
<tr>
<td>SPS-Stereo</td>
<td>15.83%</td>
</tr>
<tr>
<td>ELAS</td>
<td>17.99%</td>
</tr>
</tbody>
</table>

- #1 (ETH3D), #9 (Middlebury 2014), #14 (KITTI).
- Retains computational efficiency of SGM.
Learning to Align Images using Weak Geometric Supervision

Jing Dong\textsuperscript{1,2} Byron Boots\textsuperscript{1} Frank Dellaert\textsuperscript{1}
Ranveer Chandra\textsuperscript{2} Sudipta N. Sinha\textsuperscript{2}

\textsuperscript{1} Georgia Institute of Technology \hspace{1cm} \textsuperscript{2} Microsoft Research

3DV 2018
Learning Local Feature Descriptors

- Descriptor Learning typically needs supervised learning.
- Training them requires good image correspondences.
- For RGB images, easy to obtain such training data.
- However, not so easy for different imaging modalities (e.g. RGB/NIR).
Goal

- Given two coarsely aligned images of scenes related by an unknown 2D homography, we compute the homography parameters.
- We do not assume any prior knowledge about features or image representations.

Main Idea: We learn the feature descriptor representation from scratch on the image pair and jointly estimate the 2D homography parameters.
Siamese Networks

- Used for local descriptor learning
- Training set
  - $P$: true correspondence pairs
  - $N$: false correspondence pairs

![Diagram of Siamese Networks](image)
Siamese Networks

- Used for local descriptor learning
- Training set
  - $P$: true correspondence pairs
  - $N$: false correspondence pairs
- Contrastive Loss

$$L_0(x, x'; \theta) = \| f(x; \theta) - f(x'; \theta) \|_2$$
$$L_1(x, x'; \theta) = \max(0, \mu - \| f(x; \theta) - f(x'; \theta) \|_2)$$
$$\arg\min_\theta \left( \sum_{i=1}^{\|P\|} L_0(x_i, x'_i; \theta) + \sum_{j=1}^{\|N\|} L_1(x_j, x'_j; \theta) \right)$$
Insight

- Visualization of the training loss when several networks are trained on misaligned image patches (shifted by 2D translational offsets).
Insight

- Visualization of the training loss when several networks are trained on misaligned image patches (shifted by 2D translational offsets).

- Siamese network can be trained and homography parameters can be updated while minimizing the standard loss.
- Updates to the homography can also computed using backpropagation and SGD.
Our Formulation

- Positive set (true correspondences) re-estimated from current homography estimate

\[
L_0(x; \psi, \theta) = \|f(x; \theta) - f(w(x; \psi); \theta)\|_2
\]

\[
L_1(x, x'; \theta) = \max(0, \mu - \|f(x; \theta) - f(x'; \theta)\|_2)
\]

Homography-based image warping
Homography parameters
Our Formulation

- Positive set (true correspondences) re-estimated from current homography estimate

\[ L_0(x; \psi, \theta) = \| f(x; \theta) - f(w(x; \psi); \theta) \|_2 \]
\[ L_1(x, x'; \theta) = \max(0, \mu - \| f(x; \theta) - f(x'; \theta) \|_2) \]

Joint Optimization

\[ \theta^*, \psi^* = \arg\min_{\theta, \psi} \left( \sum_{i=1}^{|\mathcal{P}|} L_0(x_i; \psi, \theta) + \sum_{j=1}^{|\mathcal{N}|} L_1(x_j, x'_j; \theta) \right) \]
RGB-NIR image alignment

RGB

overlay

NIR

Result after alignment
Summary of Results

- Learned RGB—NIR descriptor(s) perform better than existing descriptors.
- Competitive with supervised RGB descriptors.
  - Evaluated on HPatches [Balntas+ 2017].
- Robust to medium degree of initial misalignment
  - outperforms Mutual-Information (MI) methods.
- Bootstrapping:
  - Used our method to automatically obtain precise correspondences from multiple pairs.
  - Then, trained a supervised descriptor with improved generalization to new scenes.
Conclusions

❑ New extensions of Semi Global Matching (SGM)
  ▪ Adding soft precomputed surface orientation priors.
  ▪ Using learned strategy to fuse multiple proposals.

❑ Towards aligning images from scratch
  ▪ Jointly trained a Siamese network and estimated a homography to align an image pair.
  ▪ Weakly supervised local descriptor learning.
  ▪ Extend to general scenes in the future.