Super-Resolution Keyframe Fusion for 3D Modeling with High-Quality Textures

Robert Maier, Jörg Stückler, Daniel Cremers

International Conference on 3D Vision (3DV)
October 2015, Lyon, France
Motivation

Given:

Low-resolution RGB-D frames (640 x 480)

Accurate geometric reconstruction
Motivation

Given:

Low-resolution RGB-D frames (640 x 480)

State-of-the-art w.r.t. visual appearance:
• Vertex colors: limited resolution
• Texture mapping: good-quality results, but slow/impractical
Motivation

Given:

Low-resolution RGB-D frames (640 x 480)

State-of-the-art w.r.t. visual appearance:
• Vertex colors: limited resolution
• Texture mapping: good-quality results, but slow/impractical

Problem: Gap in research of fast and robust estimation of high-quality visual appearance from low-cost RGB-D sensors
Texture Mapping with Super-Resolution Keyframes

• Our Approach:
Texture Mapping with Super-Resolution Keyframes

- Our Approach:
  - Super-resolution (SR) Keyframe Fusion and Deblurring
Texture Mapping with Super-Resolution Keyframes

• Our Approach:
  – Super-resolution (SR) Keyframe Fusion and Deblurring
  – Texture Mapping using SR keyframes (weighted median)
Texture Mapping with Super-Resolution Keyframes

- Our Approach:
  - Super-resolution (SR) Keyframe Fusion and Deblurring
  - Texture Mapping using SR keyframes (weighted median)
Related Work
Related Work

- Vertex colors (weighted average)
  - Newcombe et al., KinectFusion: Real-time dense surface mapping and tracking. ISMAR 2011
  - Sturm et al., CopyMe3D: Scanning and printing persons in 3D. GCPR 2013
Related Work

• Vertex colors (weighted average)
  – Newcombe et al., KinectFusion: Real-time dense surface mapping and tracking. ISMAR 2011
  – Sturm et al., CopyMe3D: Scanning and printing persons in 3D. GCPR 2013

• Texture Mapping
  – Weighted Median [Coorg and Teller. Automatic extraction of textured vertical facades from pose imagery. 1998]
  – Single input view per face, minimize seams [Gal et al., Seamless montage for texturing models. 2010]
  – Variational super-resolution [Goldlücke et al., A super-resolution framework for high-accuracy multiview reconstruction, IJCV 2014]
Related Work

• Vertex colors (weighted average)
  – Newcombe et al., KinectFusion: Real-time dense surface mapping and tracking. ISMAR 2011
  – Sturm et al., CopyMe3D: Scanning and printing persons in 3D. GCPR 2013

• Texture Mapping
  – Weighted Median [Coorg and Teller. Automatic extraction of textured vertical facades from pose imagery. 1998]
  – Single input view per face, minimize seams [Gal et al., Seamless montage for texturing models. 2010]
  – Variational super-resolution [Goldlücke et al., A super-resolution framework for high-accuracy multiview reconstruction, IJCV 2014]

• RGB-D based 3D Reconstruction
  – Super-resolution RGB-D keyframes [Meilland and Comport. Super-resolution 3D tracking and mapping. ICRA 2013]
  – Optimization of camera poses and non-rigid correction [Zhou and Koltun. Color map optimization for 3D reconstruction with consumer depth cameras. TOG 2014]
Related Work

• **Vertex colors (weighted average)**
  – Newcombe et al., KinectFusion: Real-time dense surface mapping and tracking. ISMAR 2011
  – Sturm et al., CopyMe3D: Scanning and printing persons in 3D. GCPR 2013

• **Texture Mapping**
  – Weighted Median [Coorg and Teller. Automatic extraction of textured vertical facades from pose imagery. 1998]
  – Single input view per face, minimize seams [Gal et al., Seamless montage for texturing models. 2010]
  – Variational super-resolution [Goldlücke et al., A super-resolution framework for high-accuracy multiview reconstruction, IJCV 2014]

• **RGB-D based 3D Reconstruction**
  – Super-resolution RGB-D keyframes [Meilland and Comport. Super-resolution 3D tracking and mapping. ICRA 2013]
  – Optimization of camera poses and non-rigid correction [Zhou and Koltun. Color map optimization for 3D reconstruction with consumer depth cameras. TOG 2014]

• **Contribution: Texture Mapping using Super-resolution Keyframes**
System Overview
System Overview

Low-resolution RGB-D frames
System Overview

Low-resolution RGB-D frames → DVO-SLAM → TSDF Volume Integration → 3D Model
System Overview

1. Low-resolution RGB-D frames
2. DVO-SLAM
3. TSDF Volume Integration
4. 3D Model
5. Keyframe Fusion
6. Super-resolution Keyframes
System Overview

Low-resolution RGB-D frames → DVO-SLAM → TSDF Volume Integration → 3D Model

Keyframe Fusion → Super-resolution Keyframes → Texel Color Computation

Parametrization

Texture
Geometric 3D Reconstruction
Geometric 3D Reconstruction

• DVO-SLAM: camera trajectory
  – [Kerl et al., Dense visual slam for RGB-D cameras. IROS 2013]
  – Real-time 3D reconstruction on a CPU
  – Robust Dense Visual Odometry
  – Loop closure detection + pose graph optimization
Geometric 3D Reconstruction

- **DVO-SLAM: camera trajectory**
  - [Kerl et al., Dense visual slam for RGB-D cameras. IROS 2013]
  - Real-time 3D reconstruction on a CPU
  - Robust Dense Visual Odometry
  - Loop closure detection + pose graph optimization

- **Model Fusion (TSDF Volume): 3D mesh**
Vertex Color Computation

- Mesh vertex $v$, input views $C_i$ (blur measure $b_i$), camera poses $\xi_i$
Vertex Color Computation

- Mesh vertex $v$, input views $C_i$ (blur measure $b_i$), camera poses $\xi_i$

- Compute observations ($x_i$ is obs. of $v$ in view $i$):
  \[ x_i = C_i(\pi(v, \xi_i)) \]
Vertex Color Computation

• Mesh vertex $v$, input views $C_i$ (blur measure $b_i$), camera poses $\xi_i$

• Compute observations ($x_i$ is obs. of $v$ in view $i$):
  \[ x_i = C_i(\pi(v, \xi_i)) \]

• Discard $x_i$ close to depth discontinuities
Vertex Color Computation

- Mesh vertex $v$, input views $C_i$ (blur measure $b_i$), camera poses $\xi_i$

- Compute observations ($x_i$ is obs. of $v$ in view $i)$:
  \[ x_i = C_i(\pi(v, \xi_i)) \]
  - Discard $x_i$ close to depth discontinuities
  - Weights of obs. $x_i$:
  \[ w_i = \frac{\cos(\theta) \cdot b_i}{d^2} \]
Vertex Color Computation

- Mesh vertex $v$, input views $C_i$ (blur measure $b_i$), camera poses $\xi_i$

- Compute observations ($x_i$ is obs. of $v$ in view $i$):
  \[ x_i = C_i(\pi(v, \xi_i)) \]

- Discard $x_i$ close to depth discontinuities

- Weights of obs. $x_i$:
  \[ w_i = \frac{\cos(\theta) \cdot b_i}{d^2} \]

- Compute vertex color $x$:
  - Weighted mean: $\arg\min_x \sum_{x_i} w_i \| x - x_i \|^2$
  - Weighted median: $\arg\min_x \sum_{x_i} w_i \| x - x_i \|$
Vertex Color Computation

Unweighted Mean

Weighted Mean

Weighted Median
Texture Mapping using SR Keyframes

• Per-vertex colors: limited resolution
• Increase resolution: Texture

• Approach:
  – SR Keyframe Fusion
  – Texture Mapping from SR keyframes
Keyframe Fusion

• Idea: fuse low-resolution (LR) input RGB-D frames into high resolution RGB-D keyframes
Keyframe Fusion

• Idea: fuse low-resolution (LR) input RGB-D frames into high resolution RGB-D keyframes
  – Depth fusion
    • Warp LR depth maps into keyframe (using relative poses)
    • Upsample and fuse depth using weighted averaging
Keyframe Fusion

• Idea: fuse low-resolution (LR) input RGB-D frames into high resolution RGB-D keyframes
  – Depth fusion
    • Warp LR depth maps into keyframe (using relative poses)
    • Upsample and fuse depth using weighted averaging
  – Color fusion
    • Deconvolution: Wiener Filter on LR input images
    • Warp fused keyframe depth to input images for color lookup
    • Fuse colors using weighted median
Keyframe Fusion

LR input image

Fused SR keyframe

R. Maier, J. Stückler, D. Cremers: Super-Resolution Keyframe Fusion for 3D Modeling with High-Quality Textures
Texture Mapping using SR Keyframes
Texture Mapping using SR Keyframes

• Texture Parametrization: One-to-one mapping between 3D mesh and 2D texture
Texture Mapping using SR Keyframes

• Texture Parametrization: One-to-one mapping between 3D mesh and 2D texture

• Texel color computation:
  – Compute 3D vertex for 2D texel (based on enclosing triangle using barycentric coordinates)
  – Compute color from SR keyframes analogous to per-vertex recoloring scheme (weighted median)
Qualitative Evaluation: Deconvolution

Without deconvolution

With deconvolution
Qualitative Evaluation: SR Keyframe resolution

- Keyframe dimensions 1280 x 960 (scale $s = 2$)
- Keyframe dimensions 2560 x 1920 (scale $s = 4$)
Qualitative Evaluation: LR input frames vs. SR keyframes

With LR input frames

With SR keyframes
## Runtime Evaluation

### Datasets:

<table>
<thead>
<tr>
<th></th>
<th>face</th>
<th>phone</th>
<th>keyboard</th>
</tr>
</thead>
<tbody>
<tr>
<td># RGB-D frames</td>
<td>512</td>
<td>1359</td>
<td>642</td>
</tr>
<tr>
<td># vertices (original)</td>
<td>159583</td>
<td>82942</td>
<td>155842</td>
</tr>
<tr>
<td># triangles (original)</td>
<td>319176</td>
<td>165888</td>
<td>311686</td>
</tr>
<tr>
<td># triangles (decimated)</td>
<td>40000</td>
<td>40000</td>
<td>40000</td>
</tr>
</tbody>
</table>
# Runtime Evaluation

## Datasets:

<table>
<thead>
<tr>
<th></th>
<th>face</th>
<th>phone</th>
<th>keyboard</th>
</tr>
</thead>
<tbody>
<tr>
<td># RGB-D frames</td>
<td>512</td>
<td>1359</td>
<td>642</td>
</tr>
<tr>
<td># vertices (original)</td>
<td>159583</td>
<td>82942</td>
<td>155842</td>
</tr>
<tr>
<td># triangles (original)</td>
<td>319176</td>
<td>165888</td>
<td>311686</td>
</tr>
<tr>
<td># triangles (decimated)</td>
<td>40000</td>
<td>40000</td>
<td>40000</td>
</tr>
</tbody>
</table>

## Runtimes:

<table>
<thead>
<tr>
<th>Method</th>
<th>$s$</th>
<th>face $t$ [s]</th>
<th>face fps</th>
<th>phone $t$ [s]</th>
<th>phone fps</th>
<th>keyboard $t$ [s]</th>
<th>keyboard fps</th>
</tr>
</thead>
<tbody>
<tr>
<td>Texture Mapping</td>
<td>2</td>
<td>57.5</td>
<td>8.9</td>
<td>222.0</td>
<td>6.1</td>
<td>72.1</td>
<td>8.9</td>
</tr>
<tr>
<td>Keyframe Fusion</td>
<td>4</td>
<td>100.9</td>
<td>5.1</td>
<td>362.8</td>
<td>2.2</td>
<td>214.9</td>
<td>3.0</td>
</tr>
<tr>
<td>SR Texture Mapping</td>
<td>4</td>
<td>26.4</td>
<td>2.0</td>
<td>58.2</td>
<td>1.4</td>
<td>42.6</td>
<td>1.5</td>
</tr>
</tbody>
</table>

(Standard desktop PC with Intel Core i7-2600 CPU with 3.40GHz and 8GB RAM)
Phone dataset

RGB input images  Vertex colors  Our approach

R. Maier, J. Stückler, D. Cremers: Super-Resolution Keyframe Fusion for 3D Modeling with High-Quality Textures
Keyboard dataset

Vertex colors

Our approach

RGB input images
Conclusion

• **Robust** and **efficient** method for high-quality texture mapping in RGB-D-based 3D reconstruction
Conclusion

- Robust and efficient method for high-quality texture mapping in RGB-D-based 3D reconstruction
  - Fuse low-quality color images into **SR keyframes**
Conclusion

• **Robust** and **efficient** method for high-quality texture mapping in RGB-D-based 3D reconstruction
  – Fuse low-quality color images into **SR keyframes**
  – **Map** high-quality keyframes onto 3D model **texture** using **weighted median** scheme
Conclusion

- **Robust and efficient** method for high-quality texture mapping in RGB-D-based 3D reconstruction
  - Fuse low-quality color images into **SR keyframes**
  - **Map** high-quality keyframes onto 3D model **texture** using **weighted median** scheme

- Experimental results:
  - **Increased photo-realism** of reconstructed 3D models
  - Very **efficient** and **practical** post-processing step (runtimes within a few minutes)
Conclusion

• Robust and efficient method for high-quality texture mapping in RGB-D-based 3D reconstruction
  – Fuse low-quality color images into SR keyframes
  – Map high-quality keyframes onto 3D model texture using weighted median scheme

• Experimental results:
  – Increased photo-realism of reconstructed 3D models
  – Very efficient and practical post-processing step (runtimes within a few minutes)

Thank you!