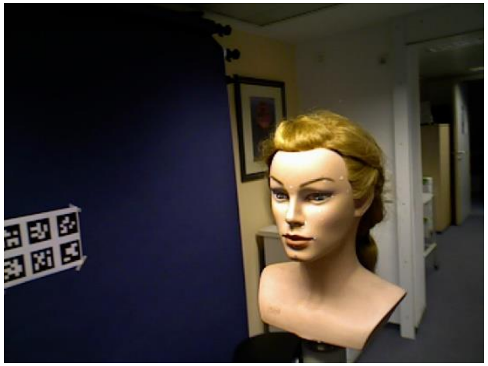
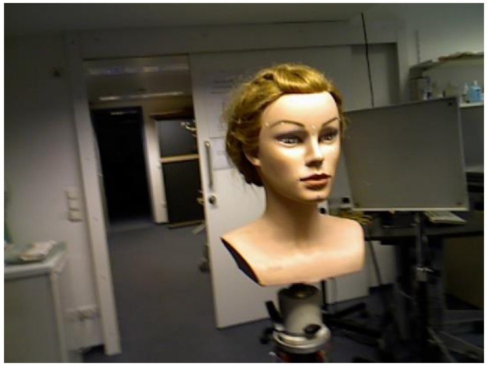
Super-Resolution Keyframe Fusion for 3D Modeling with High-Quality Textures

Robert Maier, Jörg Stückler, Daniel Cremers

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

Motivation

Given:

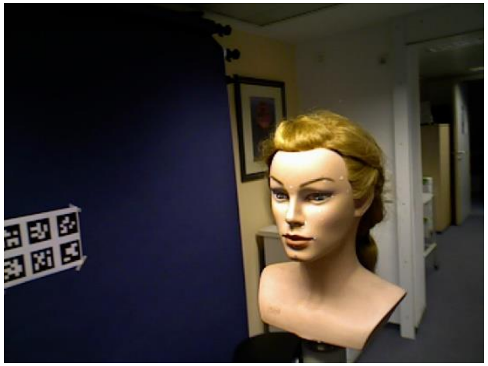



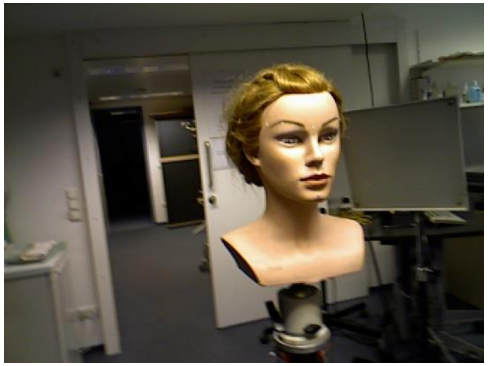
Low-resolution RGB-D frames (640 x 480)

Accurate geometric reconstruction

Motivation

Given:





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

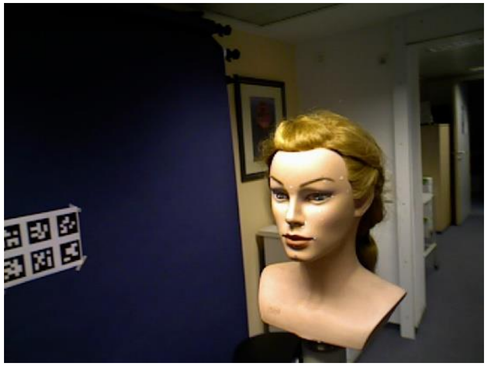
Accurate geometric reconstruction

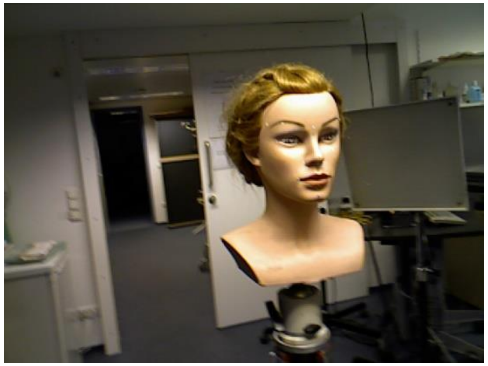
State-of-the-art w.r.t. visual appearance:

- Vertex colors: limited resolution
- Texture mapping: good-quality results, but slow/impractical

Motivation

Given:





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

Accurate geometric reconstruction

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)


Vertex colors


Our approach

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

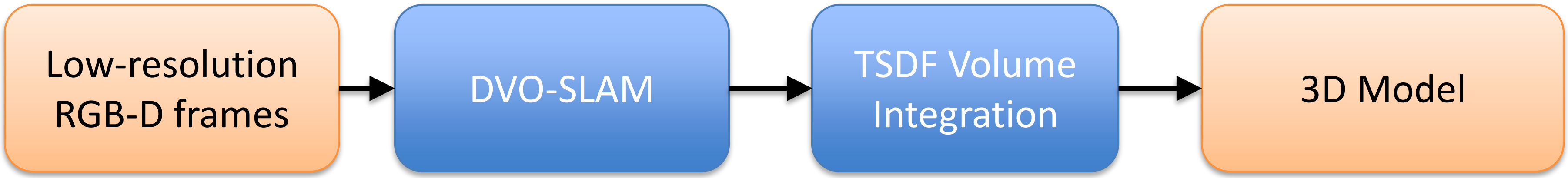
- Vertex colors (weighted average)
  - Newcombe et al., KinectFusion: Real-time dense surface mapping and tracking. ISMAR 2011
  - Sturm et al., CopyMe3D: Scanning and printing persons in 3D. GCPR 2013
- Texture Mapping
  - Weighted Median [Coorg and Teller. Automatic extraction of textured vertical 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

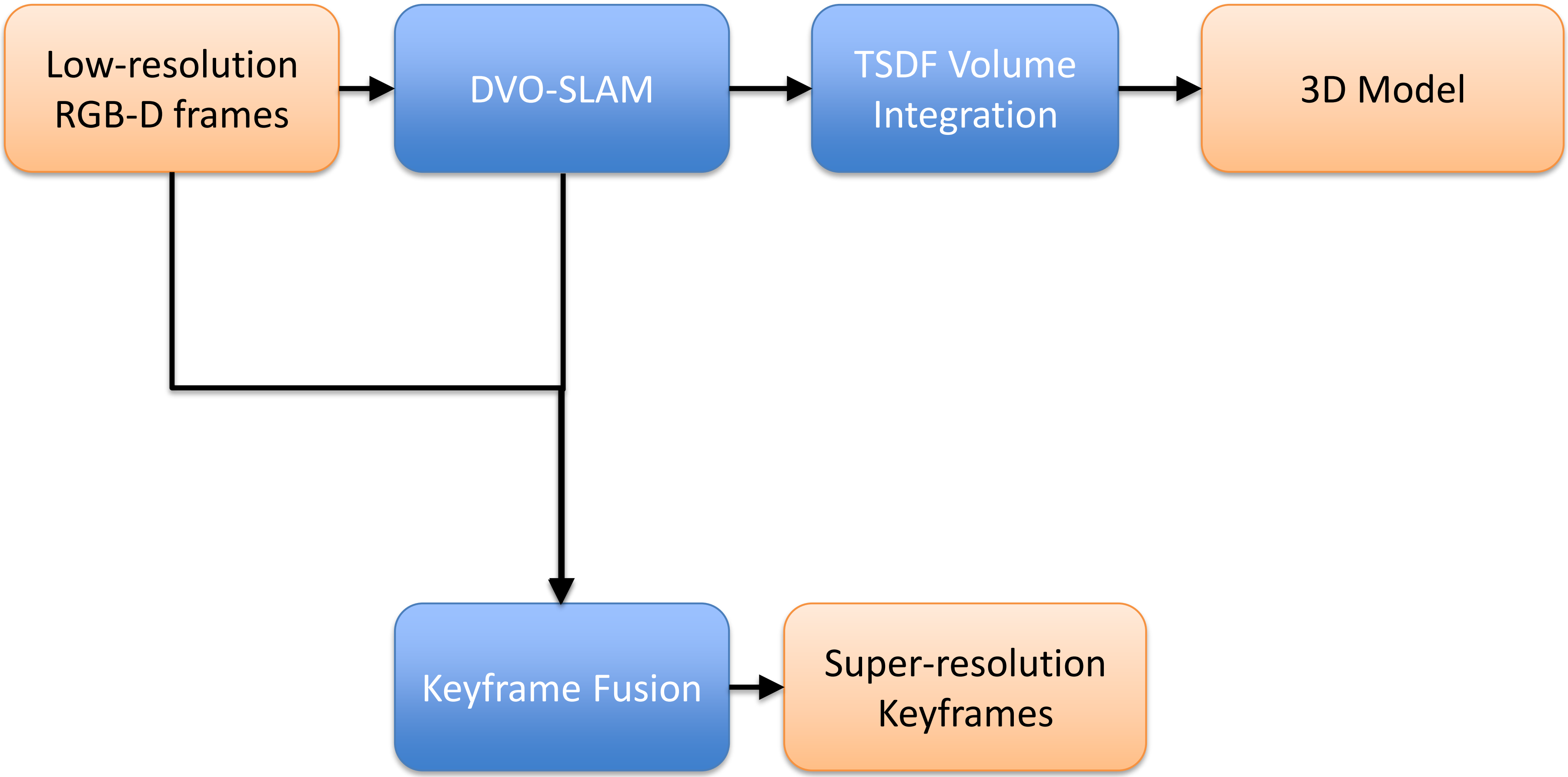
System Overview

Low-resolution
RGB-D frames

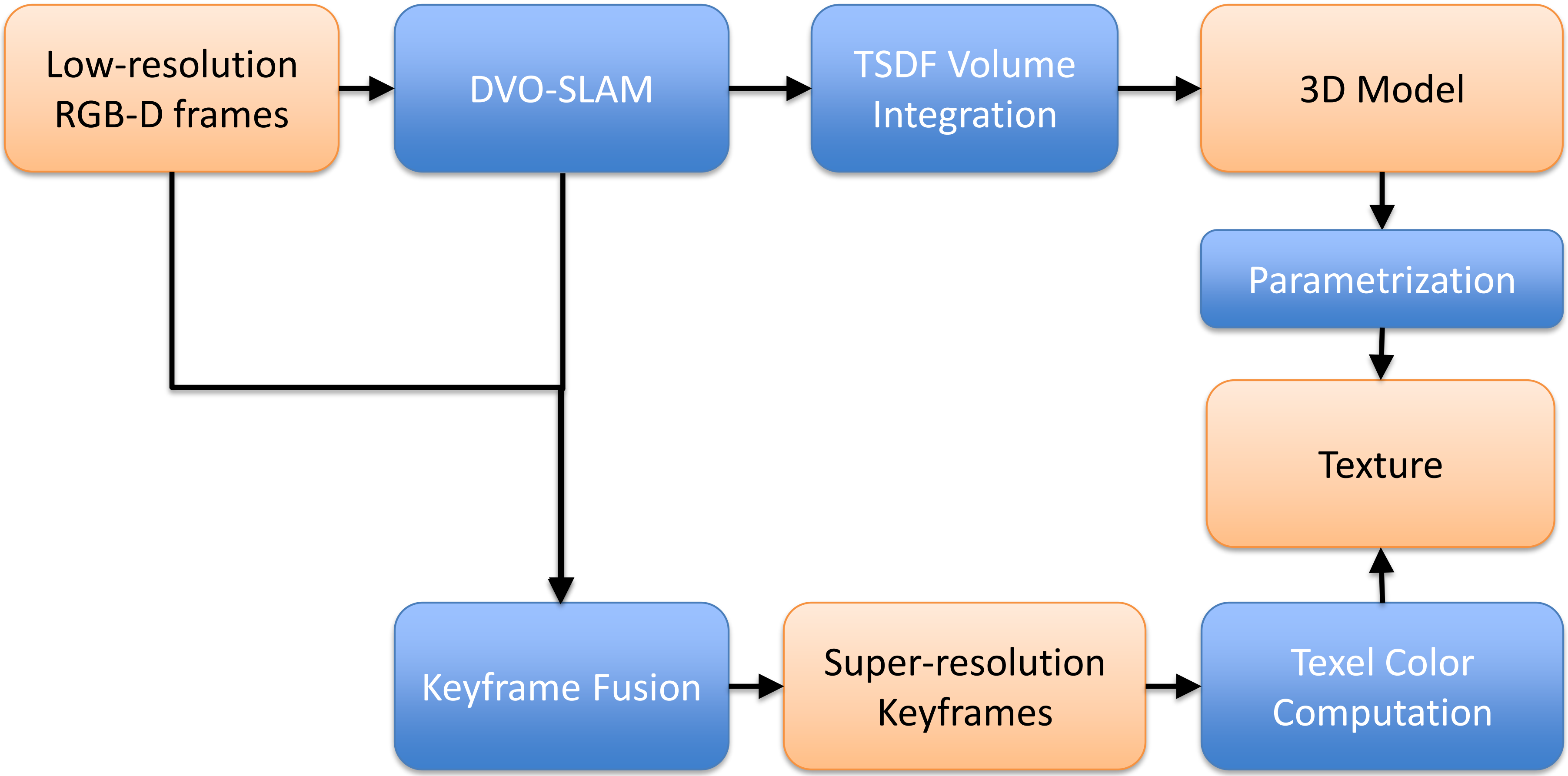
System Overview



System Overview



System Overview



Geometric 3D Reconstruction

Geometric 3D Reconstruction

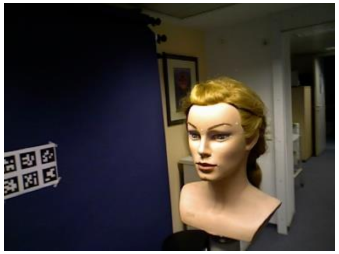
- DVO-SLAM: camera trajectory
  – [Kerl et al., Dense visual slam for RGB-D cameras. IROS 2013]
  – Real-time 3D reconstruction on a CPU
  – Robust Dense Visual Odometry
  – Loop closure detection + pose graph optimization

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

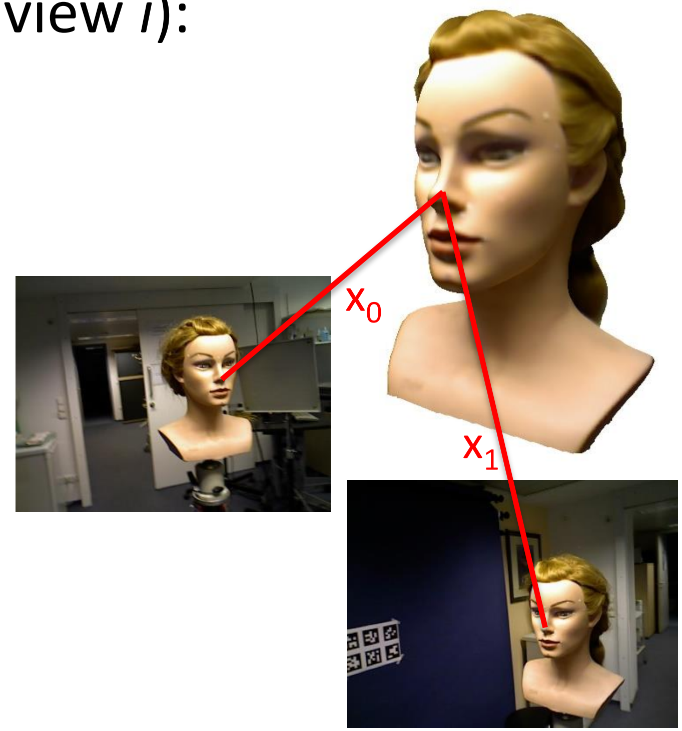
- Mesh vertex v, input views C_i (blur measure b_i), camera poses ξ_i



Vertex Color Computation

- Mesh vertex v, input views C_i (blur measure b_i), camera poses ξ_i
- Compute observations (x_i is obs. of v in view i):

$$x_i = C_i(\pi(v, \xi_i))$$

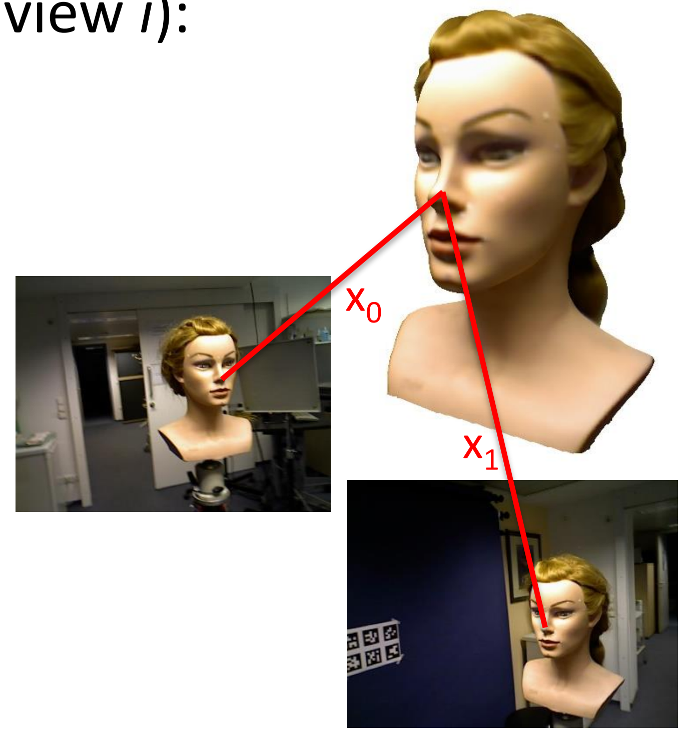


Vertex Color Computation

- Mesh vertex v, input views C_i (blur measure b_i), camera poses ξ_i
- Compute observations (x_i is obs. of v in view i):

$$x_i = \mathcal{C}_i(\pi(v, \xi_i))$$

- Discard x_i close to depth discontinuities



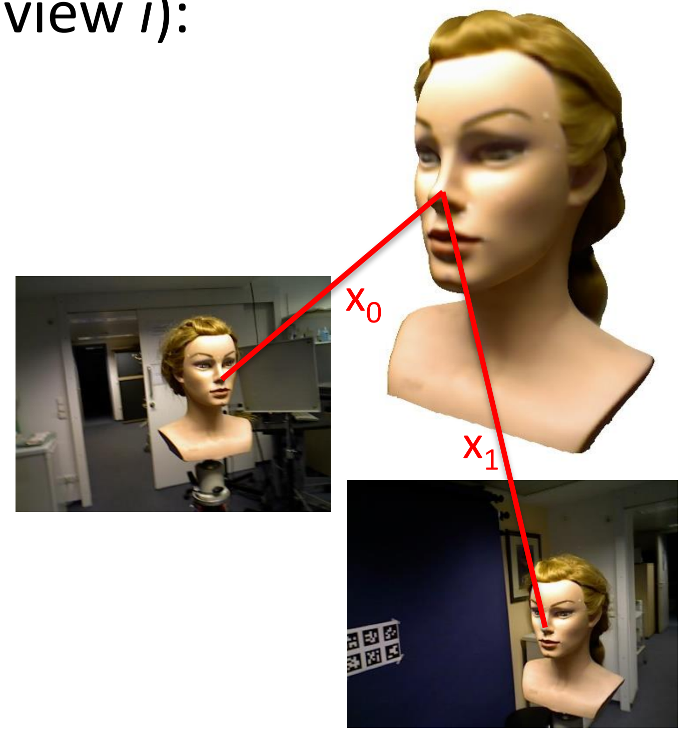
Vertex Color Computation

- Mesh vertex v, input views C_i (blur measure b_i), camera poses ξ_i
- Compute observations (x_i is obs. of v in view i):

$$x_i = \mathcal{C}_i(\pi(v, \xi_i))$$

- Discard x_i close to depth discontinuities
- Weights of obs. x_i:

$$w_i = \frac{\cos(\theta) * b_i}{d^2}$$



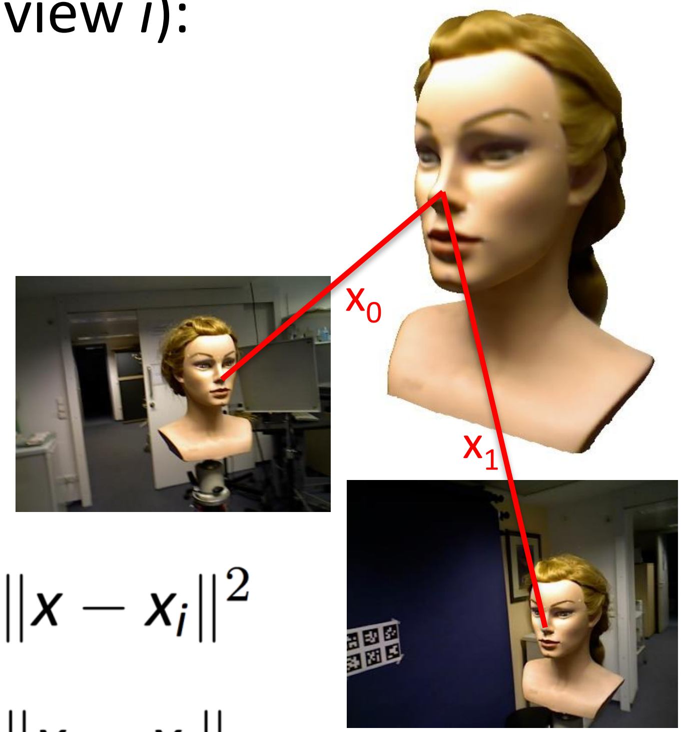
Vertex Color Computation

- Mesh vertex v, input views C_i (blur measure b_i), camera poses ξ_i
- Compute observations (x_i is obs. of v in view i):

$$x_i = \mathcal{C}_i(\pi(v, \xi_i))$$

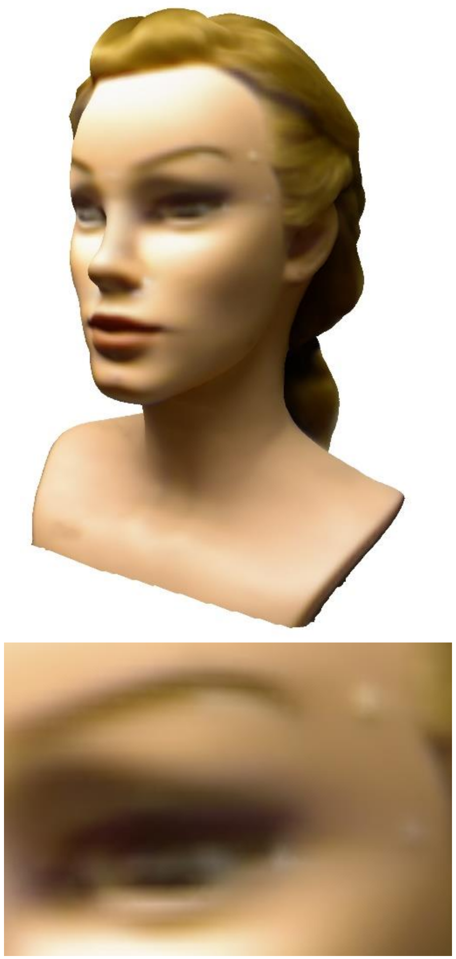
- Discard x_i close to depth discontinuities
- Weights of obs. x_i:

$$w_i = \frac{\cos(\theta) * b_i}{d^2}$$

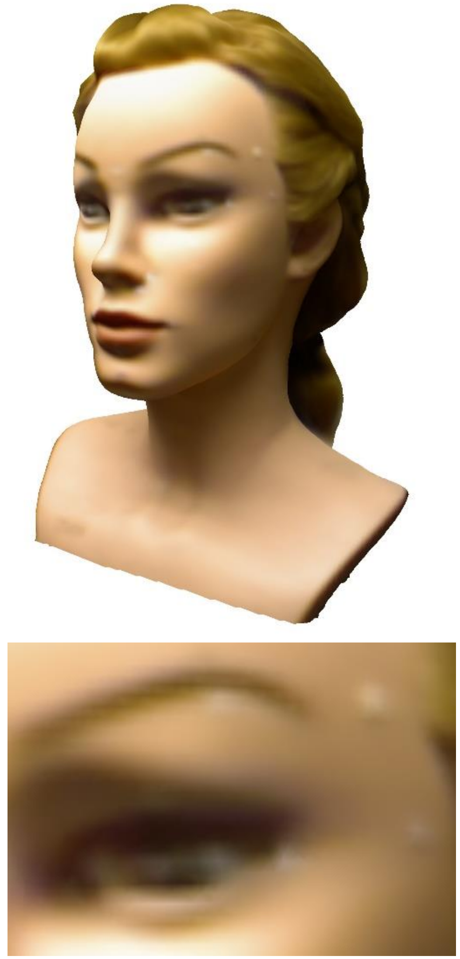


- Compute vertex color x:
  – Weighted mean: $\arg\min_x \sum_{x_i} w_i \|x - x_i\|^2$
  – Weighted median: $\arg\min_x \sum_{x_i} w_i \|x - x_i\|$

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

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:

	face	*phone*	*keyboard*
# RGB-D frames	512	1359	642
# vertices (original)	159583	82942	155842
# triangles (original)	319176	165888	311686
# triangles (decimated)	40000	40000	40000

Runtime Evaluation

Datasets:

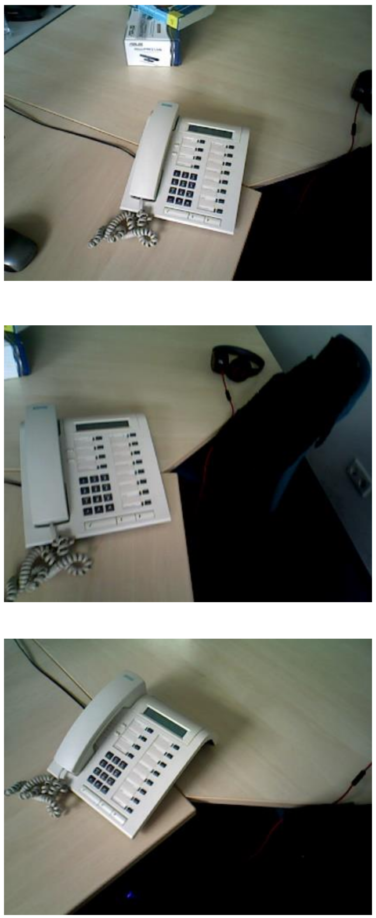
	face	*phone*	*keyboard*
# RGB-D frames	512	1359	642
# vertices (original)	159583	82942	155842
# triangles (original)	319176	165888	311686
# triangles (decimated)	40000	40000	40000

Runtimes:

		face		*phone*		*keyboard*	
	s	t [s]	fps	t [s]	fps	t [s]	fps
Texture Mapping		91.5	5.6	330.8	4.1	128.8	5.0
Keyframe Fusion	2	57.5	8.9	222.0	6.1	72.1	8.9
SR Texture Mapping	2	18.7	2.8	50.7	2.7	18.8	3.5
Keyframe Fusion	4	100.9	5.1	362.8	2.2	214.9	3.0
SR Texture Mapping	4	26.4	2.0	58.2	1.4	42.6	1.5

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

Phone dataset



RGB input images

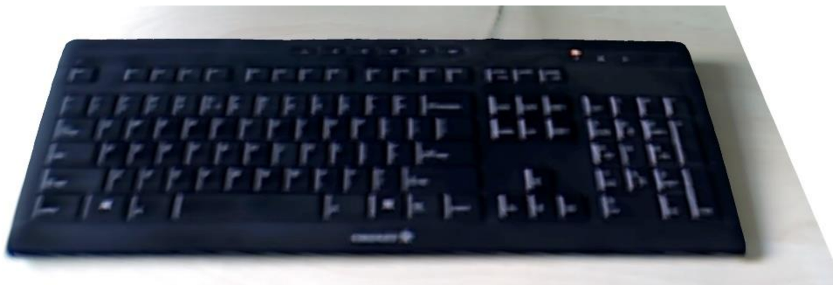
Vertex colors

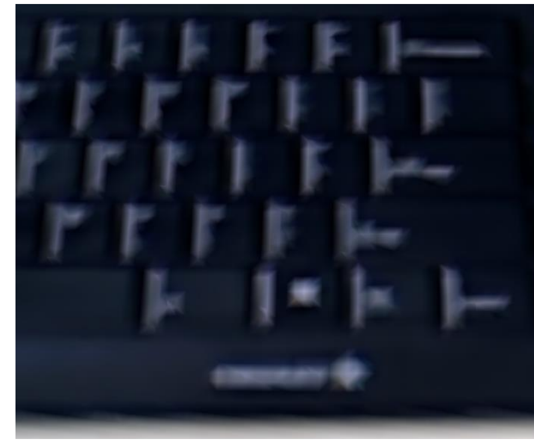
Our approach

Keyboard dataset

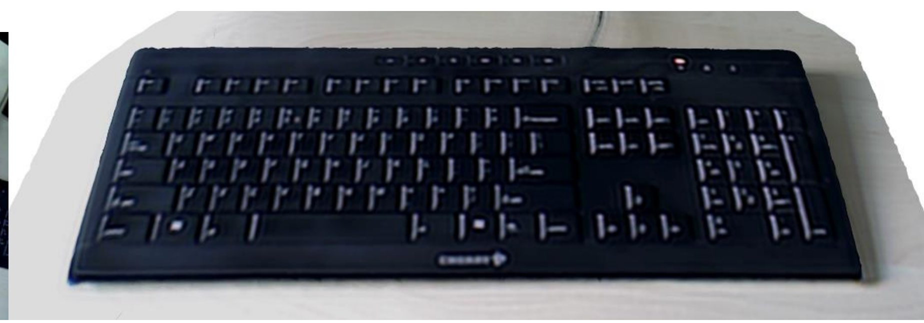
RGB input images

Vertex colors





Our approach



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!