High-Quality 3D Reconstruction from RGB-D Sensors

Computer Vision II: Multiple View Geometry

Current Research

Robert Maier

robert.maier@in.tum.de
https://vision.in.tum.de/members/maierr

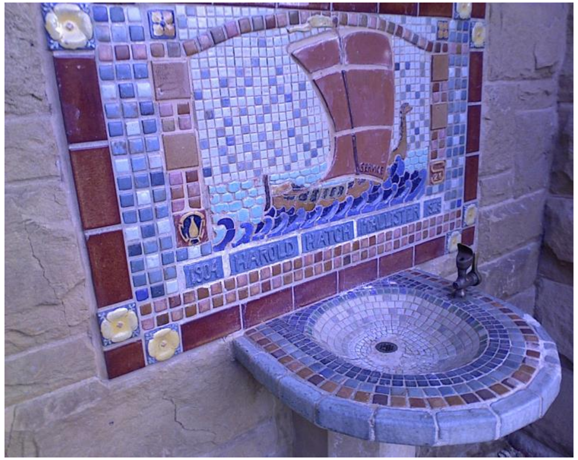
July 19, 2017

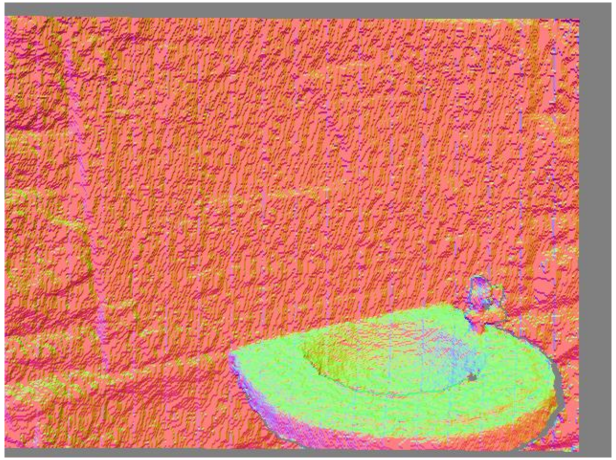
Overview

- RGB-D Sensors
- Real-Time Camera Tracking and 3D Reconstruction Using Signed Distance Functions (Bylow et al, RSS 2013)
- De-noising, Stabilizing and Completing 3D Reconstructions On-the-go using Plane Priors (Dzitsiuk et al, ICRA 2017)
- Intrinsic3D: High-Quality 3D Reconstruction by Joint Appearance and Geometry Optimization with Spatially-Varying Lighting (Maier et al, ICCV 2017)

RGB-D Sensors

- RGB-D: color (RGB) + depth (metric!)
- Depth 640x480px @ 30 fps
- Color (up to 1280x1024px @ 10 fps)





RGB-D Sensors

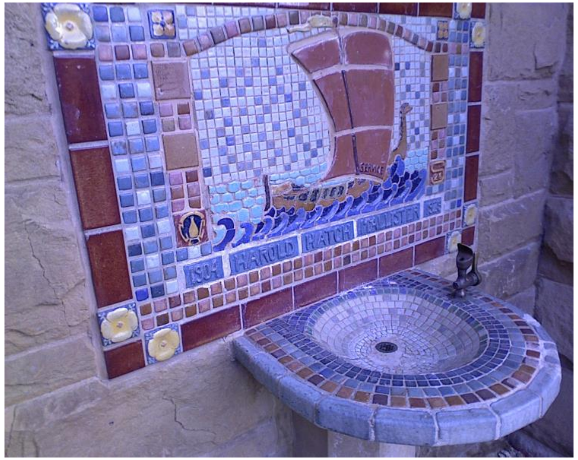
- RGB-D: color (RGB) + depth (metric!)
- Depth 640x480px @ 30 fps
- Color (up to 1280x1024px @ 10 fps)
- Structured Light / Time-of-flight
- Low-cost!

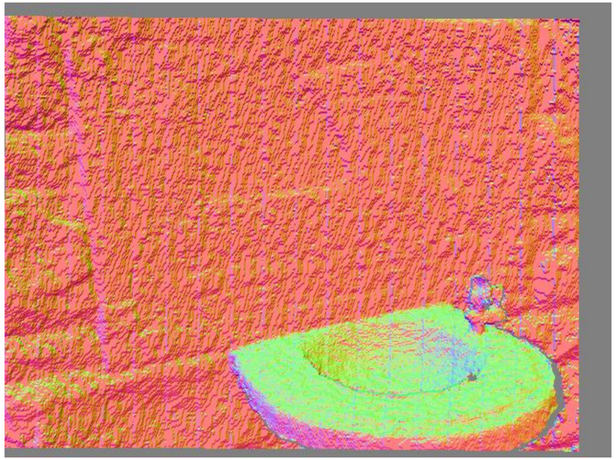
Microsoft Kinect v1

Occipital Structure Sensor

Asus Xtion

Intel RealSense R200





Google Tango
(Lenovo Phab 2 Pro)

RGB-D based 3D reconstruction

- Dense real-time 3D reconstruction of real-world objects/scenes from RGB-D data
- SLAM: Simultaneous Localization and Mapping (RGB-D-SLAM)
- Focus in this talk: methods that use Signed Distance Fields (SDF) as model
- Applications:
  - Augmented/Virtual Reality
  - Robotics
  - Industrial inspection
  - etc

KinectFusion (Newcombe et al, ISMAR 2011)

HTC Vive

Parrot AR drone

Real-Time Camera Tracking and 3D Reconstruction Using Signed Distance Functions

E. Bylow[1], J. Sturm[2], C. Kerl[2], F. Kahl[1], D. Cremers[2]

[1] Lund University

[2] Technical University of Munich

Robotics: Science and Systems (RSS)
2013, Berlin, Germany

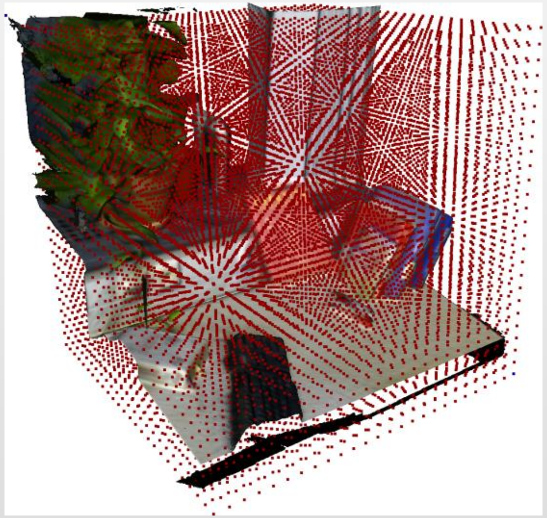
System Pipeline

- Based on KinectFusion (Newcombe et al, ISMAR 2011)

System Pipeline

- Based on KinectFusion (Newcombe et al, ISMAR 2011)

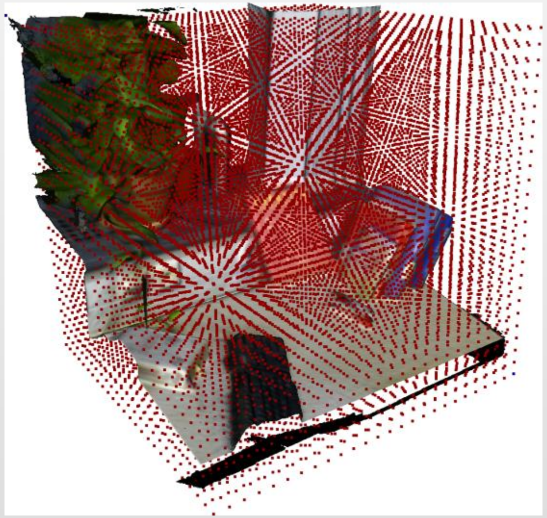
3D model: Signed Distance Field (SDF volume)



System Pipeline

- Based on KinectFusion (Newcombe et al, ISMAR 2011)

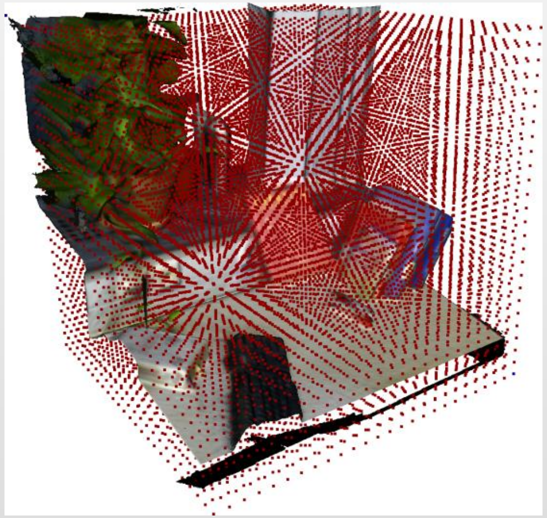
3D model: Signed Distance Field (SDF volume)



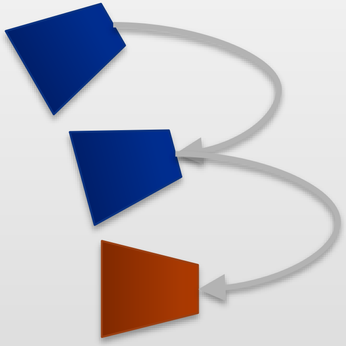
System Pipeline

- Based on KinectFusion (Newcombe et al, ISMAR 2011)

3D model: Signed Distance Field (SDF volume)

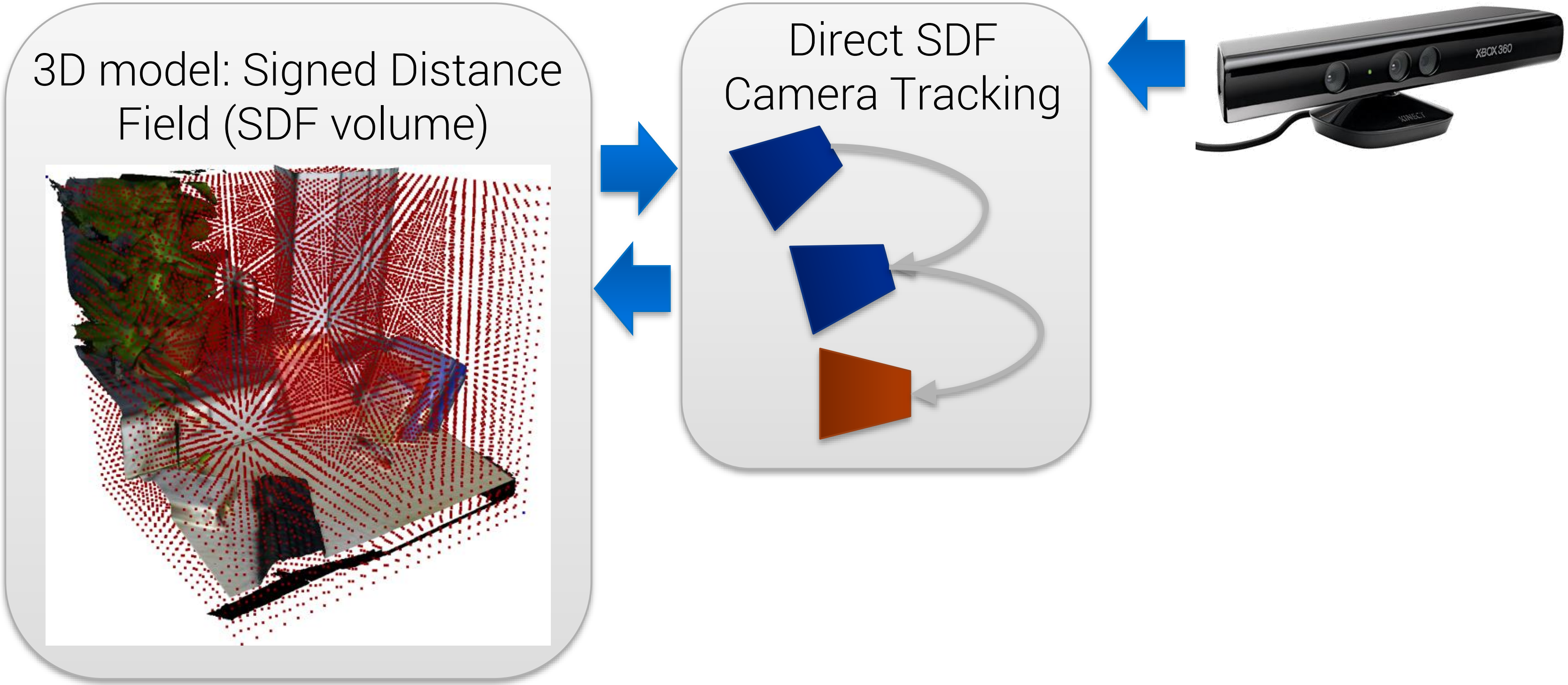


Direct SDF Camera Tracking



System Pipeline

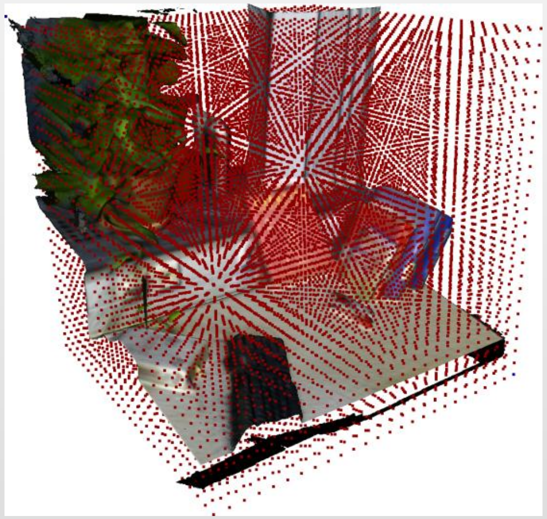
- Based on KinectFusion (Newcombe et al, ISMAR 2011)



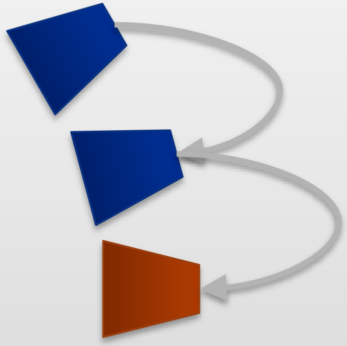
System Pipeline

- Based on KinectFusion (Newcombe et al, ISMAR 2011)

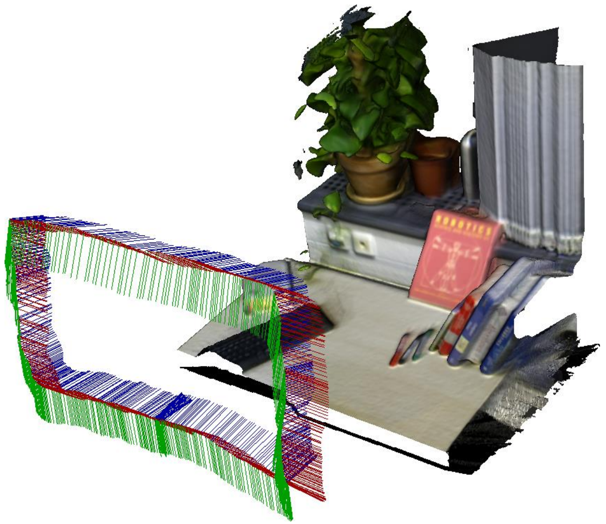
3D model: Signed Distance Field (SDF volume)



Direct SDF Camera Tracking

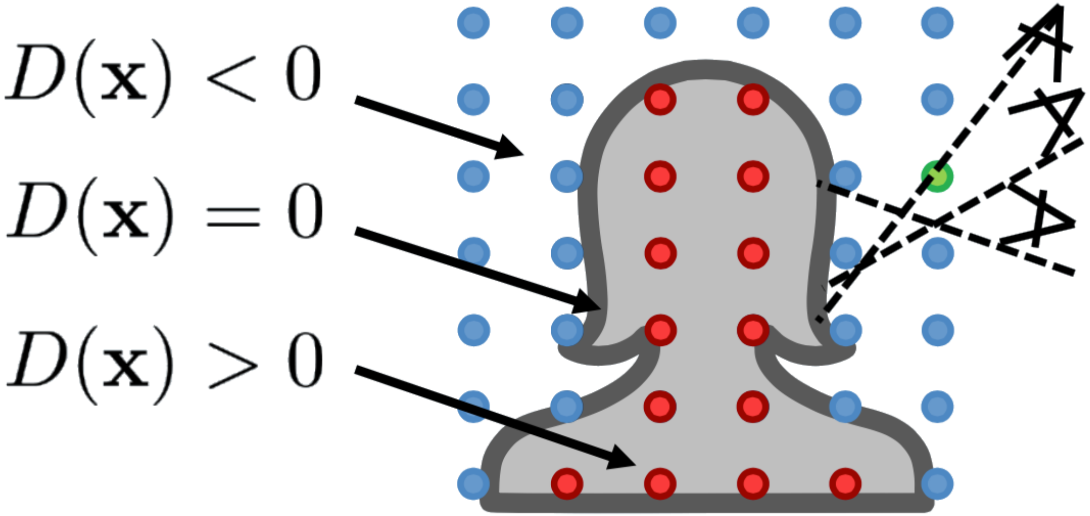


Marching Cubes: 3D Mesh



SDF Volume

- Volumetric 3D model representation: dense voxel grid
- Each voxel stores:
  - Signed Distance Function (SDF): signed distance to closest surface
  - Color values
  - Weights



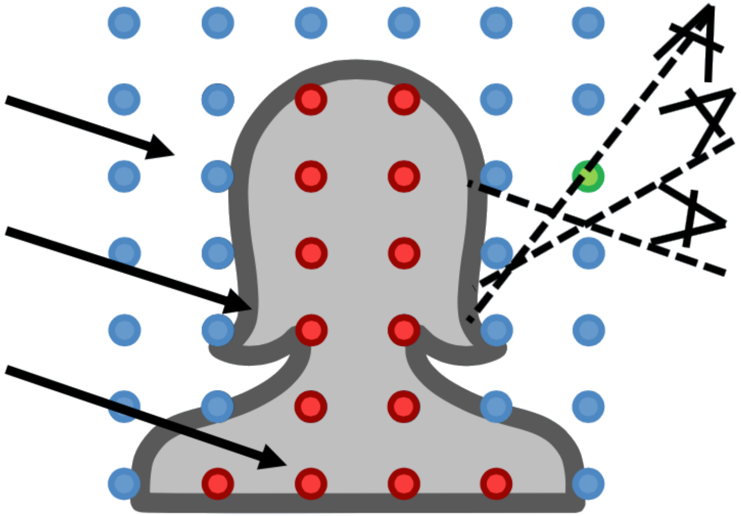
SDF Volume

- Volumetric 3D model representation: dense voxel grid
- Each voxel stores:
  - Signed Distance Function (SDF): signed distance to closest surface
  - Color values
  - Weights

$D(\mathbf{x}) < 0$

$D(\mathbf{x}) = 0$

$D(\mathbf{x}) > 0$



Update (weighted average) :

$$D \leftarrow \frac{WD + wd}{W + w}$$

$$C \leftarrow \frac{WC + wc}{W + w}$$

$$W \leftarrow W + w$$

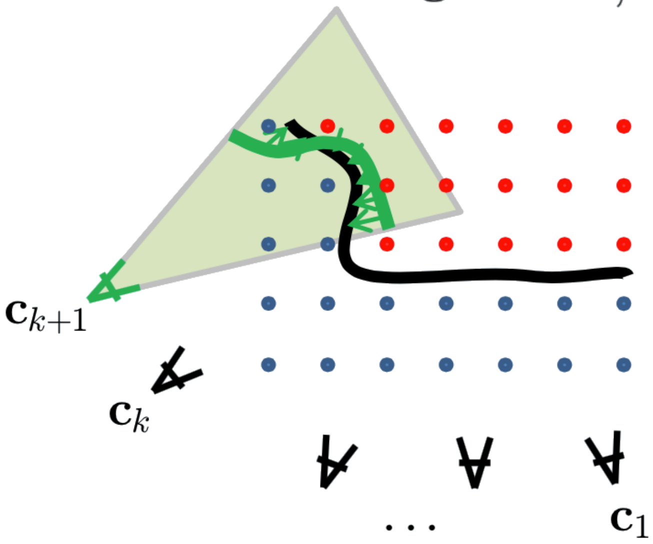
Camera tracking

- Estimate current camera pose from input RGB-D frame
- KinectFusion: synthetic depth map from SDF (raycasting) + ICP alignment

Camera tracking

- Estimate current camera pose from input RGB-D frame
- KinectFusion: synthetic depth map from SDF (raycasting) + ICP alignment
- Novel direct camera tracking against SDF: direct minimization of error between input depth map and SDF:

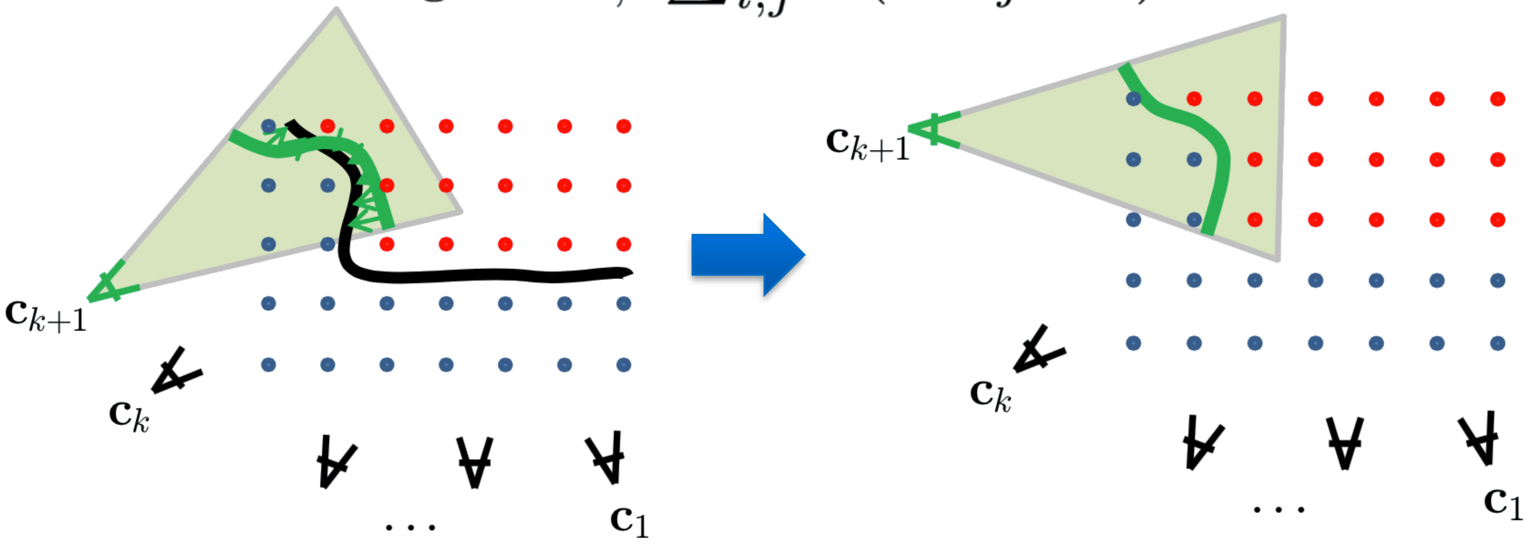
$$\arg\min_{R,\mathbf{t}} \sum_{i,j} D(R\mathbf{x}_{ij} + \mathbf{t})^2$$



Camera tracking

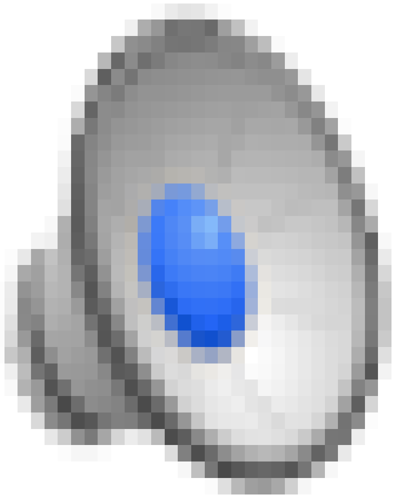
- Estimate current camera pose from input RGB-D frame
- KinectFusion: synthetic depth map from SDF (raycasting) + ICP alignment
- Novel direct camera tracking against SDF: direct minimization of error between input depth map and SDF:

$$\arg\min_{R,\mathbf{t}} \sum_{i,j} D(R\mathbf{x}_{ij} + \mathbf{t})^2$$



Extension: CopyMe3D

CopyMe3D: Scanning and Printing Persons in 3D (Sturm et al, GCPR 2013)

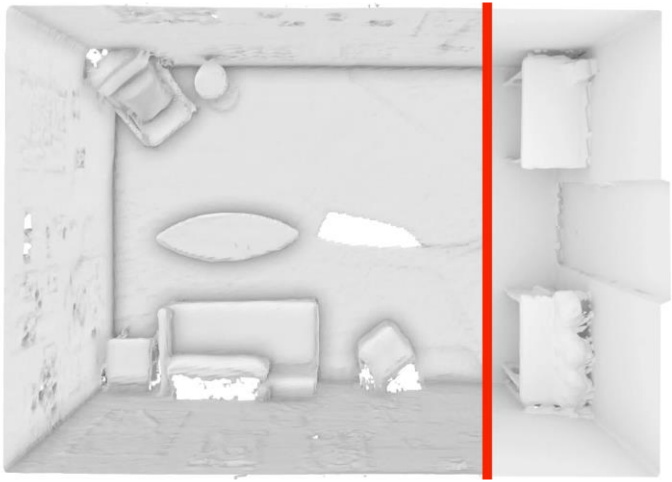


CopyMe3D

Printed 3D figures

De-noising, Stabilizing and Completing 3D Reconstructions On-the-go using Plane Priors

M. Dzitsiuk[1,2], J. Sturm[2], R. Maier[1], L. Ma[1], D. Cremers[1]

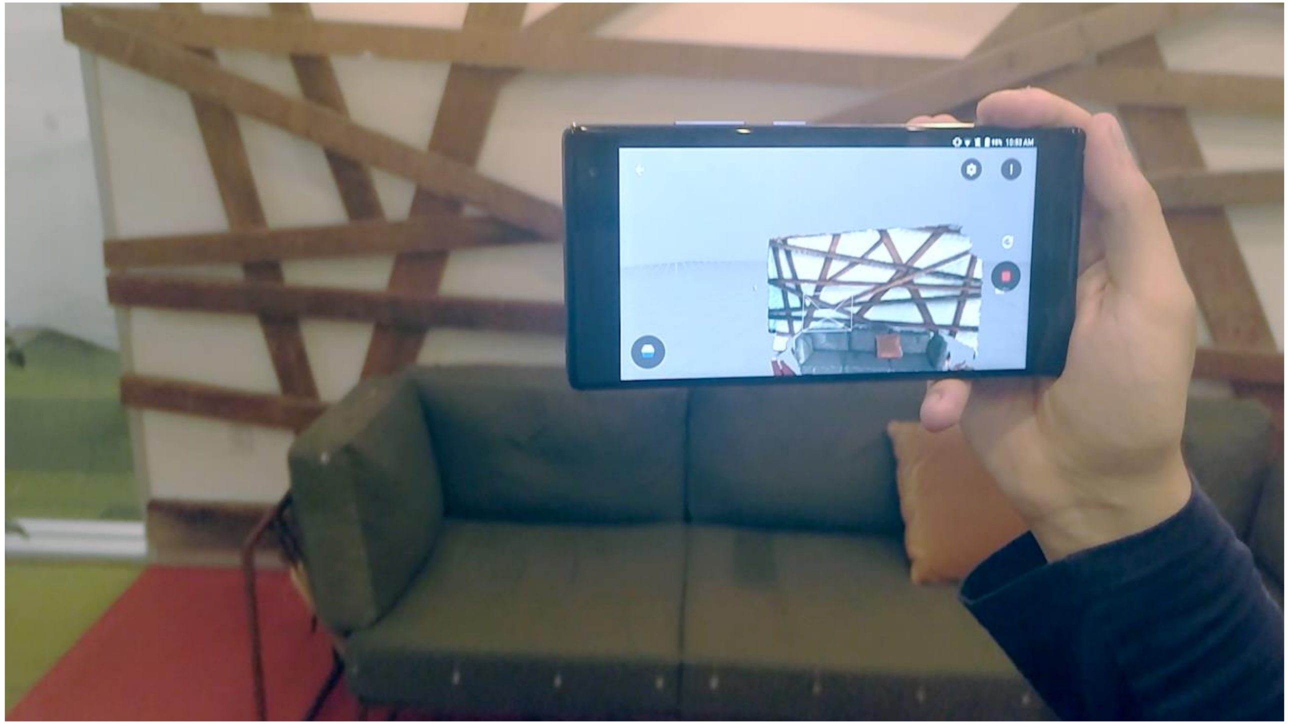


[1] Google

[2] Technical University of Munich

International Conference on Robotics and Automation (ICRA), May 2017, Singapore

Real-time 3D reconstruction on mobile device

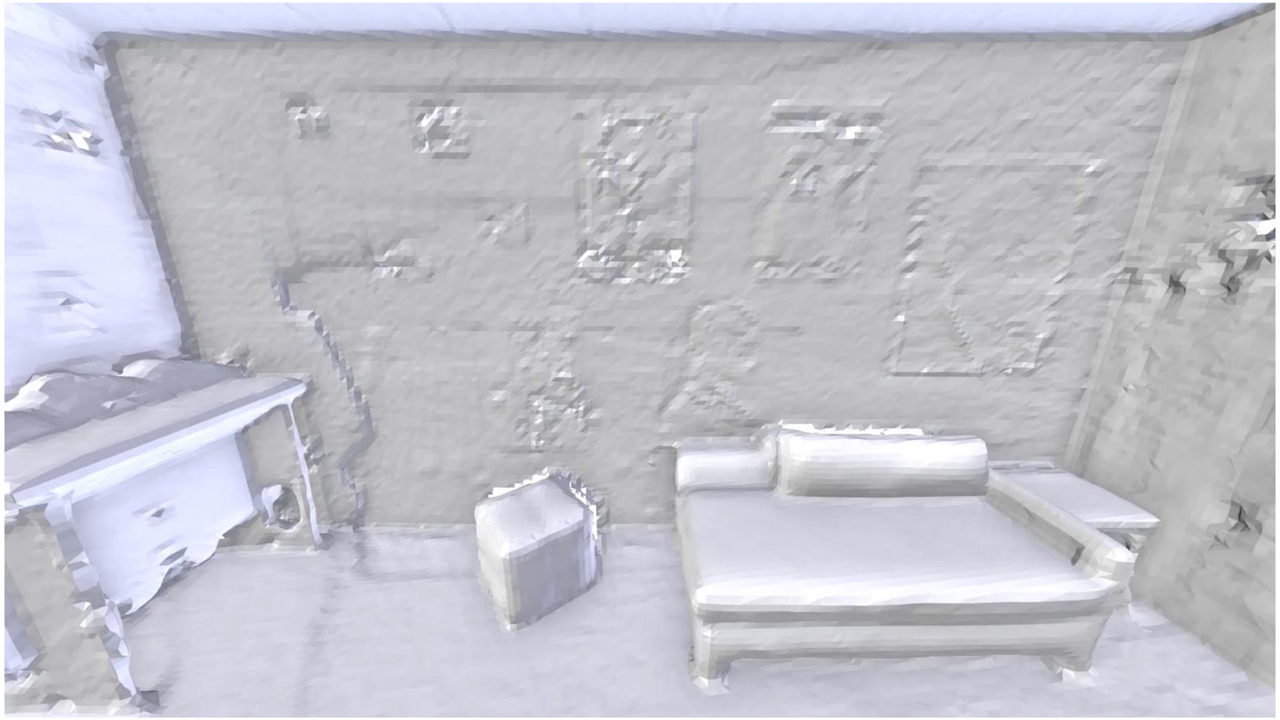


Newcombe et al.
“KinectFusion: Real-time dense surface mapping and tracking.”, 2011.

Nießner et al.
"Real-time 3D reconstruction at scale using voxel hashing.", 2013.

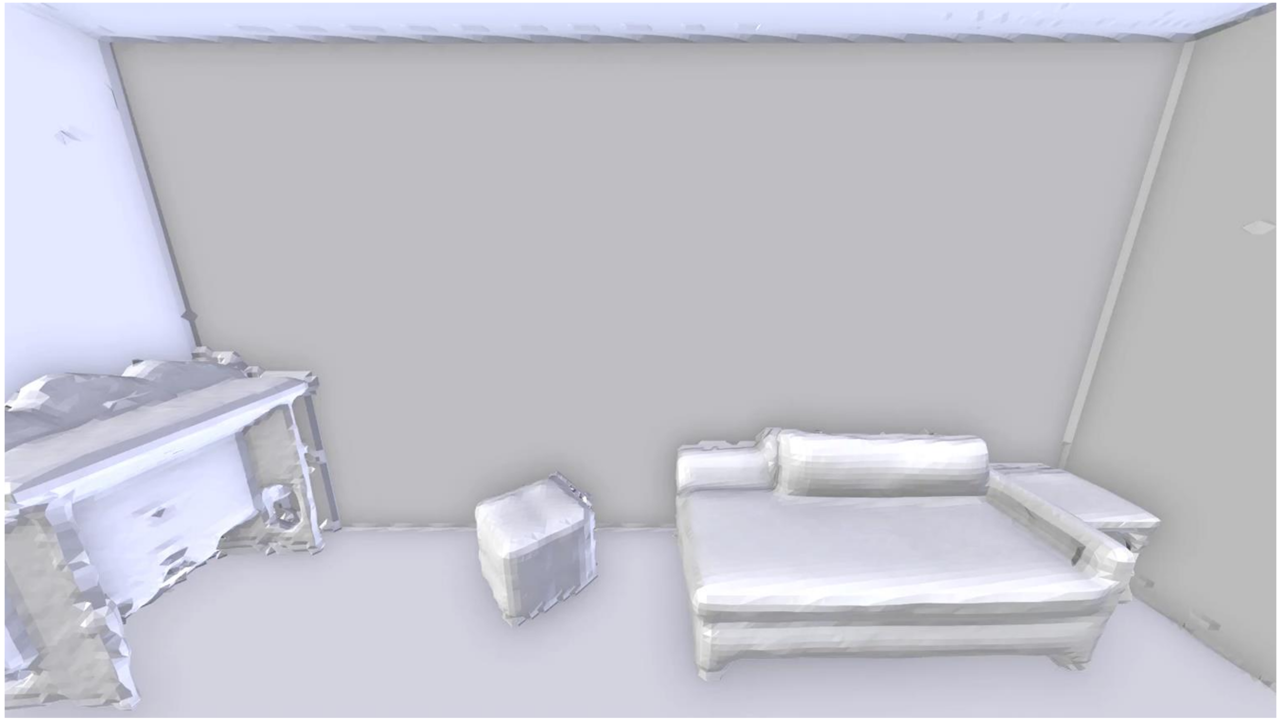
Klingensmith et al.
“Chisel: Real Time Large Scale 3D Reconstruction Onboard a Mobile Device using Spatially Hashed Signed Distance Fields.”, 2015.

Motivation



- Problems with real-time 3D reconstruction:
  - Noisy
  - Incomplete
  - No segmentation

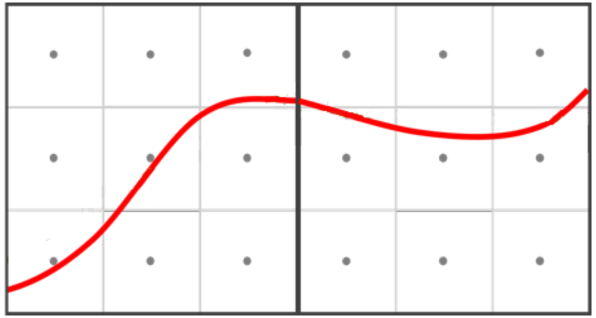
Motivation



- Solution idea: detect and use planes

Approach

- **Input:** Signed Distance Field divided into chunks



Approach

- **Input:** Signed Distance Field divided into chunks
- 1. **Find plane candidates**
  - Using robust least squares on SDF

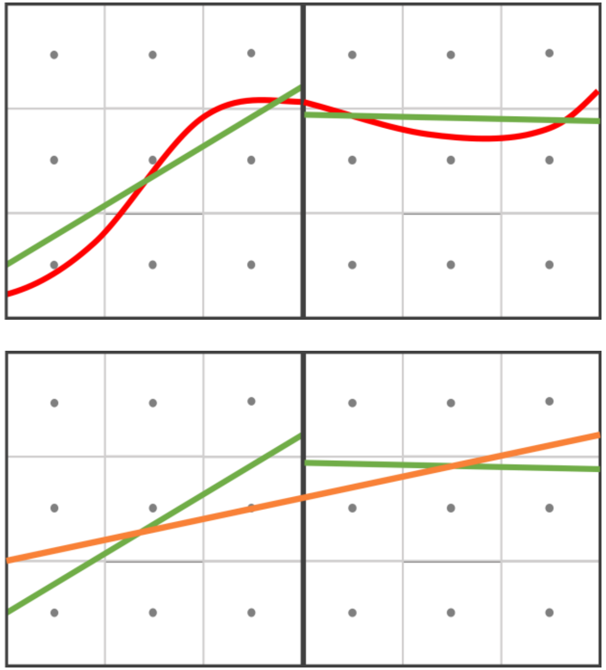
$$\arg\min_{\mathbf{p}} \sum_{\mathbf{x}_k \in \mathcal{V}} w(r_k)(r_k)^2$$

Approach

- **Input:** Signed Distance Field divided into chunks
- 1. **Find plane candidates**
  - Using robust least squares on SDF

$$\arg\min_{\mathbf{p}} \sum_{\mathbf{x}_k \in \mathcal{V}} w(r_k)(r_k)^2$$

- 2. **Merge planes**
  - RANSAC on plane candidates

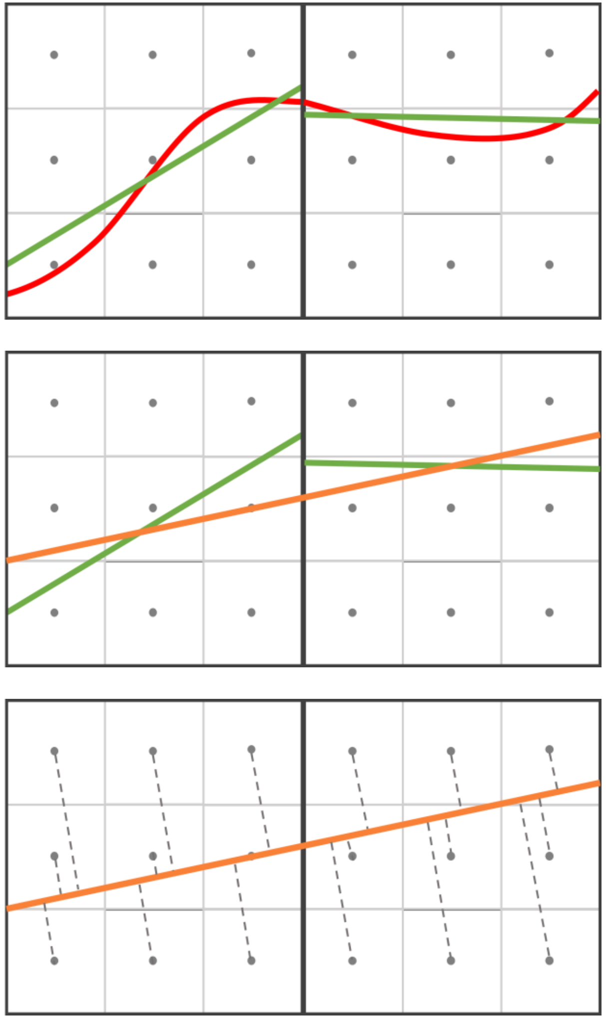


Approach

- **Input:** Signed Distance Field divided into chunks
- 1. **Find plane candidates**
  - Using robust least squares on SDF

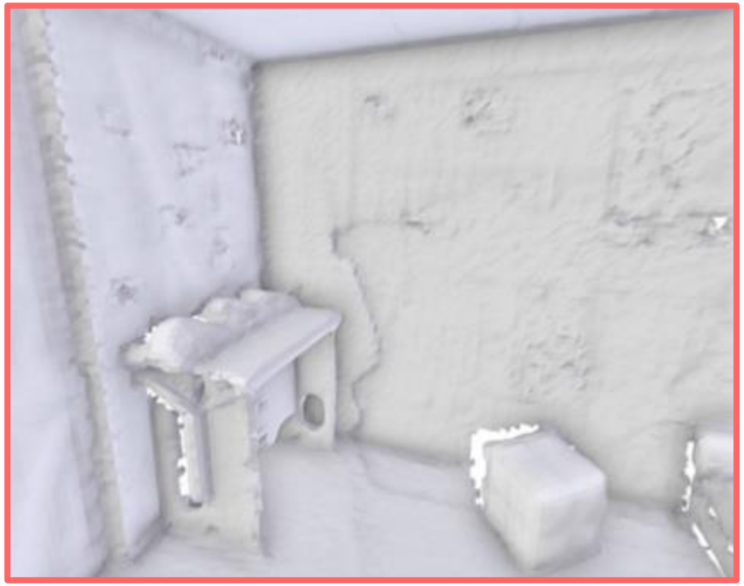
$$\underset{\mathbf{p}}{\arg\min} \sum_{\mathbf{x}_k \in \mathcal{V}} w(r_k)(r_k)^2$$

- 2. **Merge planes**
  - RANSAC on plane candidates
- 3. **De-noising**
  - Create new SDF that combines original values with planes



Results: De-noising

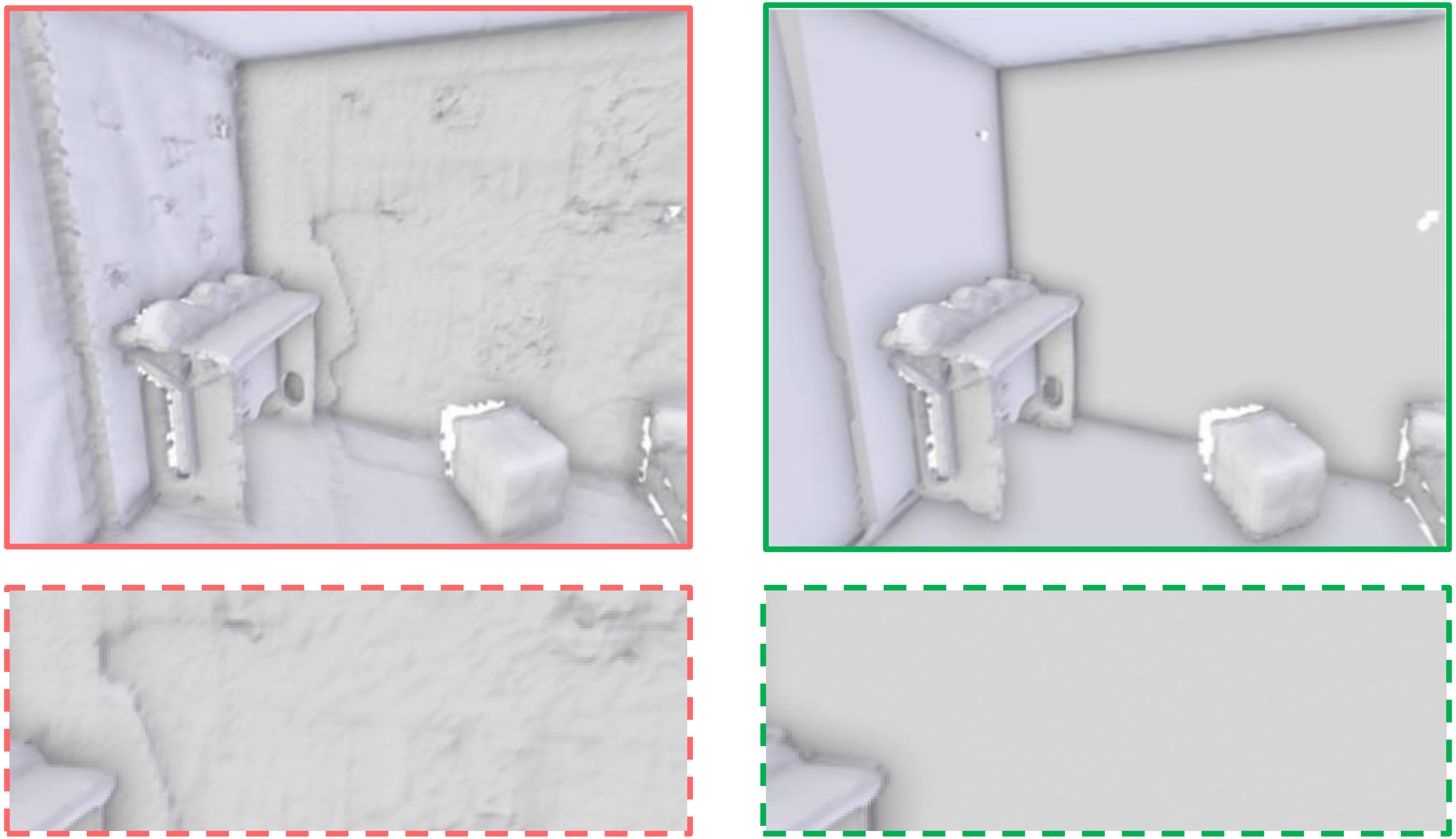
Before



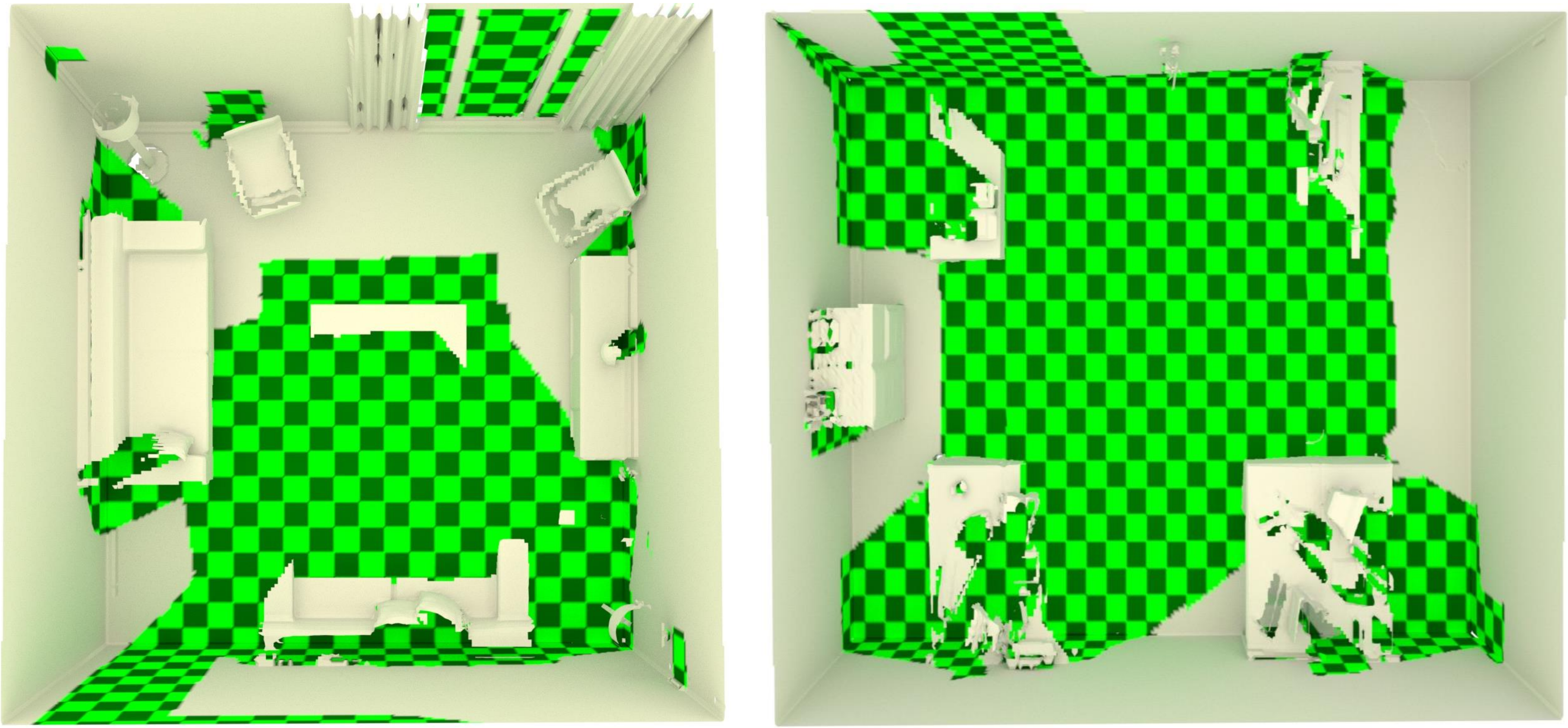
Results: De-noising

Before

After



Results: Hole filling

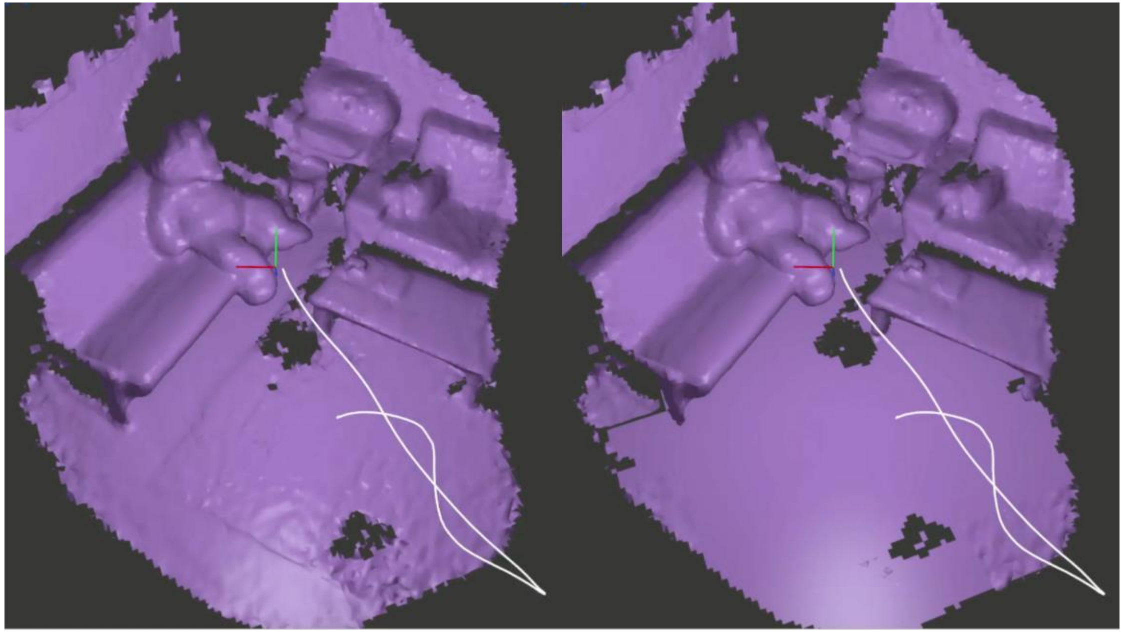


Results: Segmentation

- Classify reconstruction geometry:
  - **Floor** or **wall**, based on area and angle with gravity
  - **Object**, based on mesh connected components

Conclusions

- Real-time 3D reconstruction on mobile device
- Incorporates plane priors into Signed Distance Field
- Enables de-noising, hole filling, segmentation



Intrinsic3D: High-Quality 3D Reconstruction by Joint Appearance and Geometry Optimization with Spatially-Varying Lighting

R. Maier[1,2], K. Kim[2], M. Nießner[1,3], D. Cremers[1], J. Kautz[2]

[1] NVIDIA [2] Technical University of Munich [3] Stanford University

International Conference on Computer Vision (ICCV)
October 2017, Venice, Italy

Overview

- Motivation
- Previous Work
- Approach
- Results
- Conclusion

Motivation

Motivation

- Requirement of high-quality 3D content for Augmented Reality, Virtual Reality, Gaming, …

HTC Vive

NVIDIA VR Funhouse

Motivation

- Requirement of high-quality 3D content for Augmented Reality, Virtual Reality, Gaming, …
- Usually: manual modelling (e.g. Maya)

HTC Vive

NVIDIA VR Funhouse

Motivation

- Requirement of high-quality 3D content for Augmented Reality, Virtual Reality, Gaming, …
- Usually: manual modelling (e.g. Maya)
- Wide availability of commodity RGB-D sensors (e.g. Kinect): efficient methods for 3D reconstruction of real-word objects

HTC Vive

NVIDIA VR Funhouse

Motivation

- Requirement of high-quality 3D content for Augmented Reality, Virtual Reality, Gaming, …
- Usually: manual modelling (e.g. Maya)
- Wide availability of commodity RGB-D sensors (e.g. Kinect): efficient methods for 3D reconstruction of real-word objects
- Question: how to reconstruct high-quality 3D models with best-possible geometry and color from low-cost depth sensors?

HTC Vive

NVIDIA VR Funhouse

Motivation

- Goal: High-Quality Reconstruction of Geometry and Color from commodity RGB-D sensors

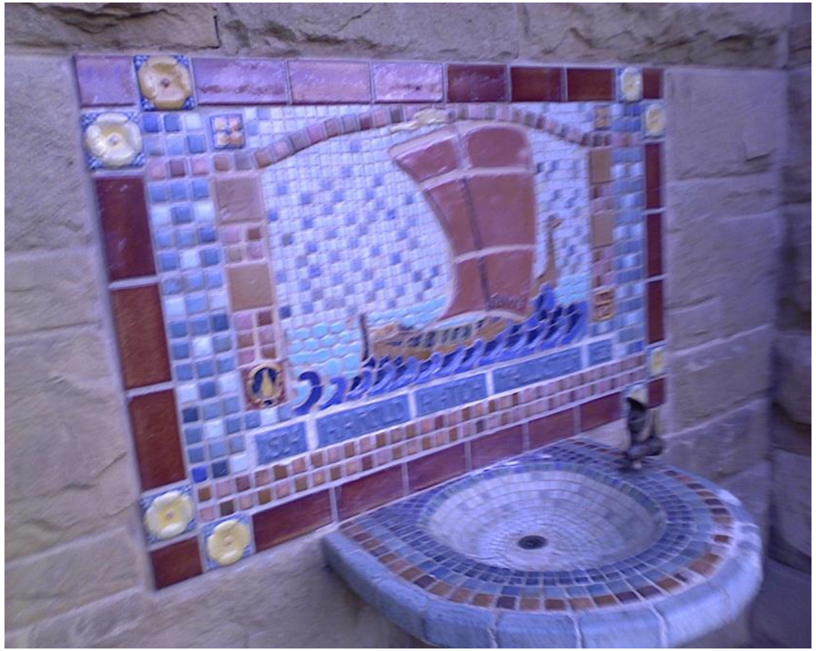
Motivation

- Goal: High-Quality Reconstruction of Geometry and Color from commodity RGB-D sensors

- Challenges:

Motivation

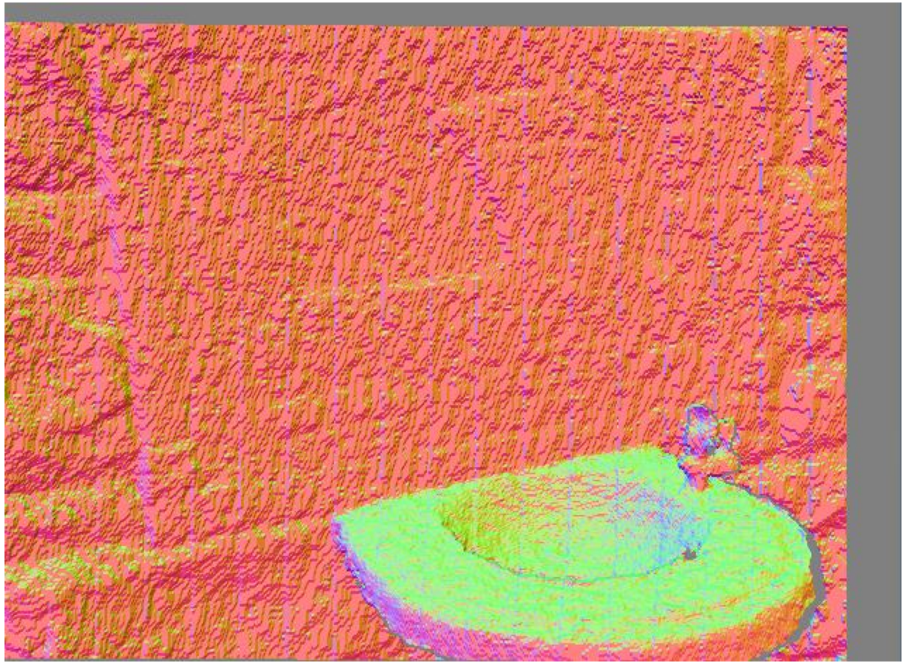
- Goal: High-Quality Reconstruction of Geometry and Color from commodity RGB-D sensors
- Challenges:
  - Input data quality (e.g. motion blur, sensor noise)



Motivation

- Goal: High-Quality Reconstruction of Geometry and Color from commodity RGB-D sensors

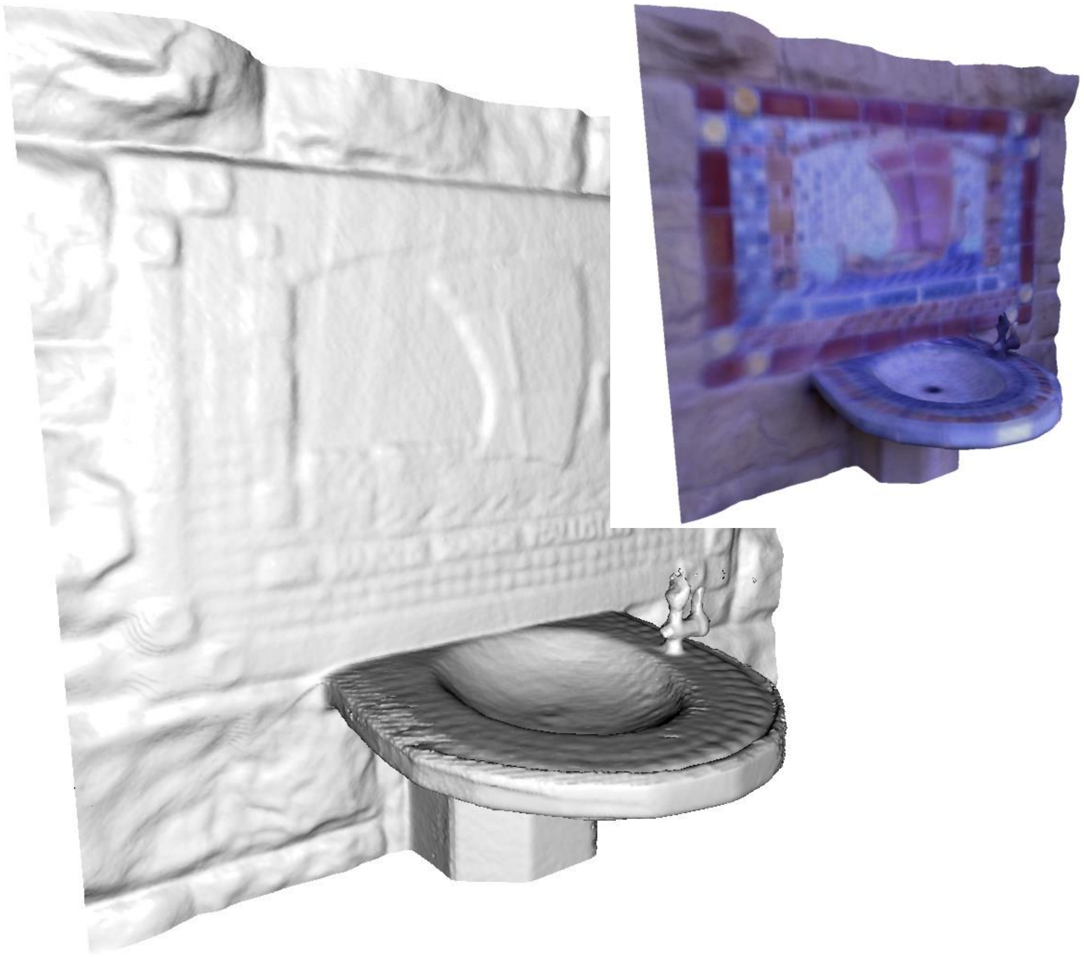
- Challenges:
  - Input data quality (e.g. motion blur, sensor noise)



Motivation

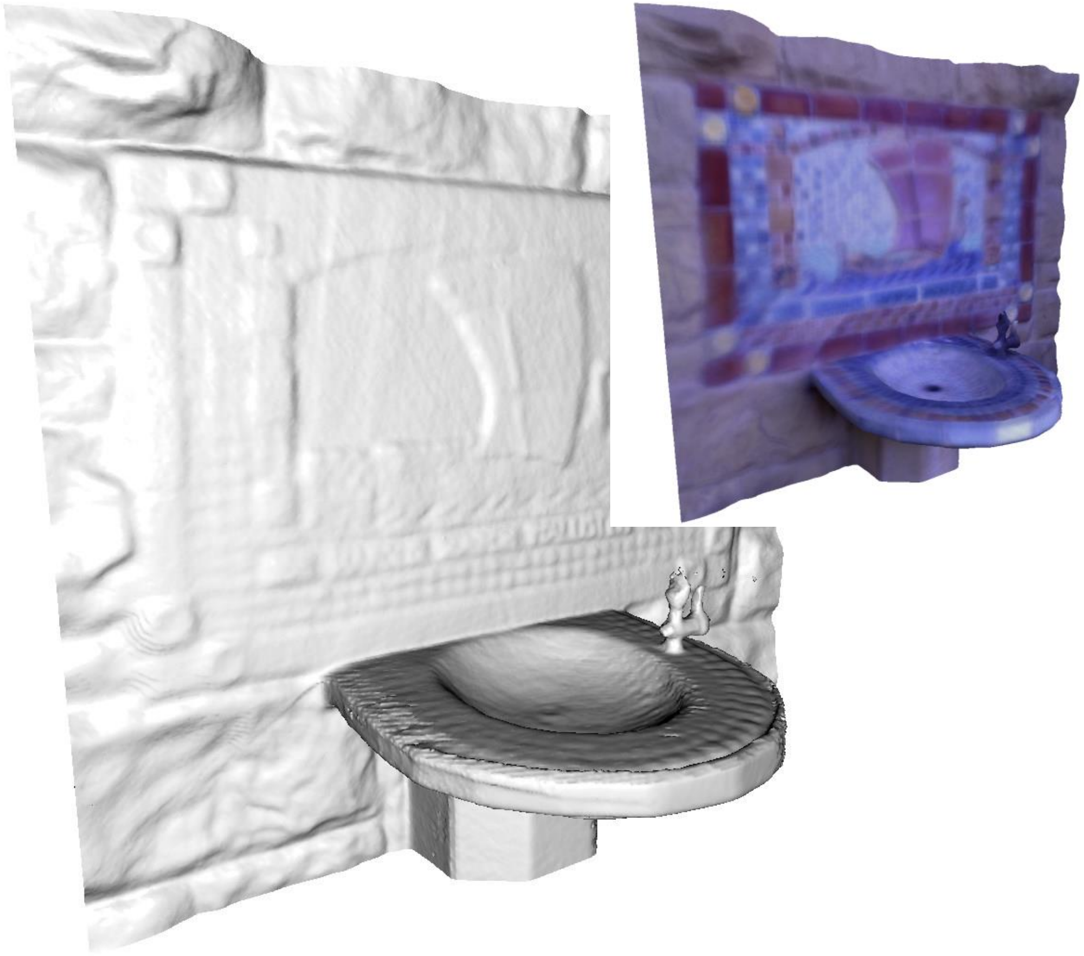
- Goal: High-Quality Reconstruction of Geometry and Color from commodity RGB-D sensors
- Challenges:
  - Input data quality (e.g. motion blur, sensor noise)
  - Inaccurate camera pose estimation
  - (Slightly) inaccurate and over-smoothed geometric reconstruction

Our Method



Our Method

- Temporal view sampling & filtering techniques (input frames)

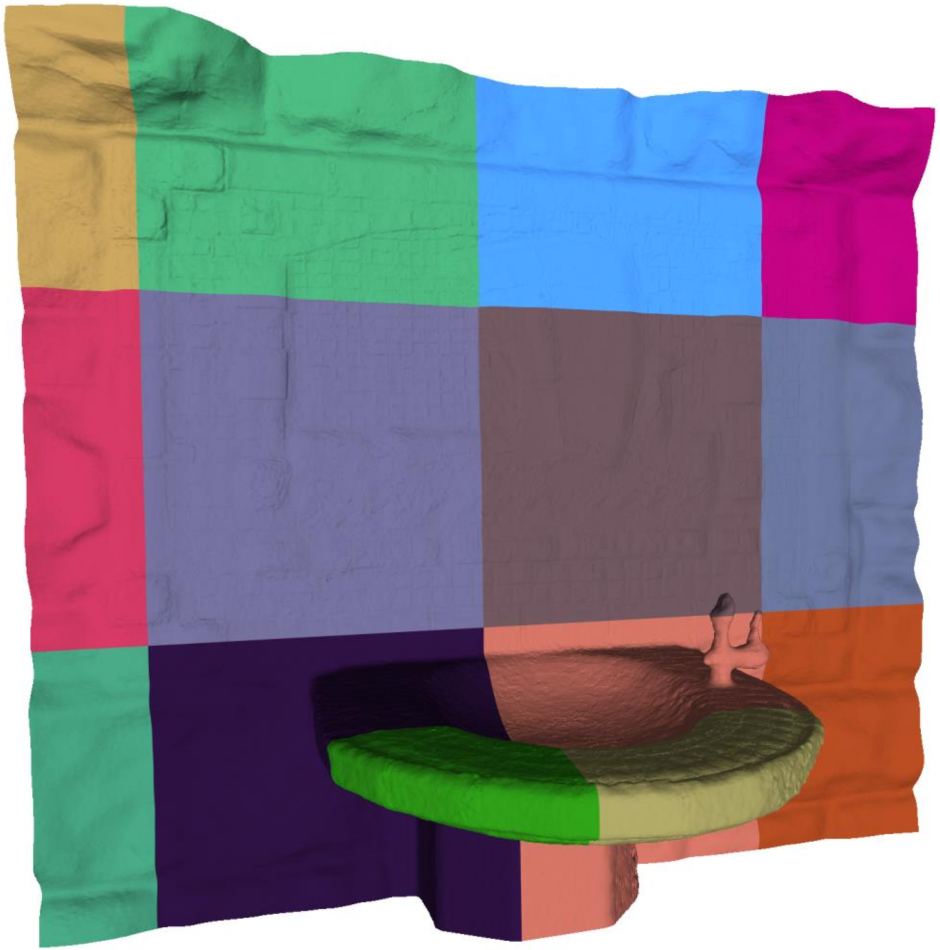


Our Method

- Temporal view sampling & filtering techniques (input frames)
- Joint optimization of
  - surface & albedo (Signed Distance Field)
  - image formation model (camera poses & intrinsics)

Our Method

- Temporal view sampling & filtering techniques (input frames)
- Joint optimization of
  - surface & albedo (Signed Distance Field)
  - image formation model (camera poses & intrinsics)
- Lighting estimation using Spatially-Varying Spherical Harmonics (SVSH)



Our Method

- Temporal view sampling & filtering techniques (input frames)
- Joint optimization of
  - surface & albedo (Signed Distance Field)
  - image formation model (camera poses & intrinsics)
- Lighting estimation using Spatially-Varying Spherical Harmonics (SVSH)

Our Method

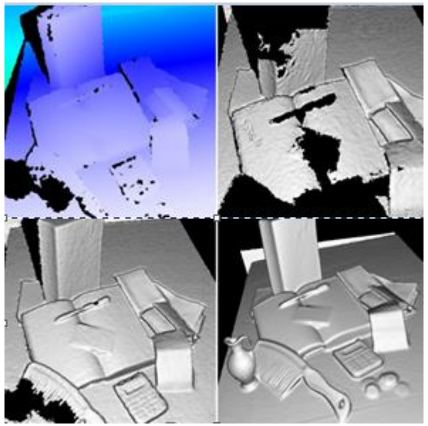
- Temporal view sampling & filtering techniques (input frames)
- Joint optimization of
  - surface & albedo (Signed Distance Field)
  - image formation model (camera poses & intrinsics)
- Lighting estimation using Spatially-Varying Spherical Harmonics (SVSH)
- Optimized colors (by-product)

Previous Work

State-of-the-art

RGB-D based 3D Reconstruction

- Given a stream of RGB-D frames of an object/scene, compute its 3D shape that maximizes the geometric-consistency
- Real-time, robust, fairly accurate geometric reconstructions



KinectFusion, 2011

DynamicFusion, 2015

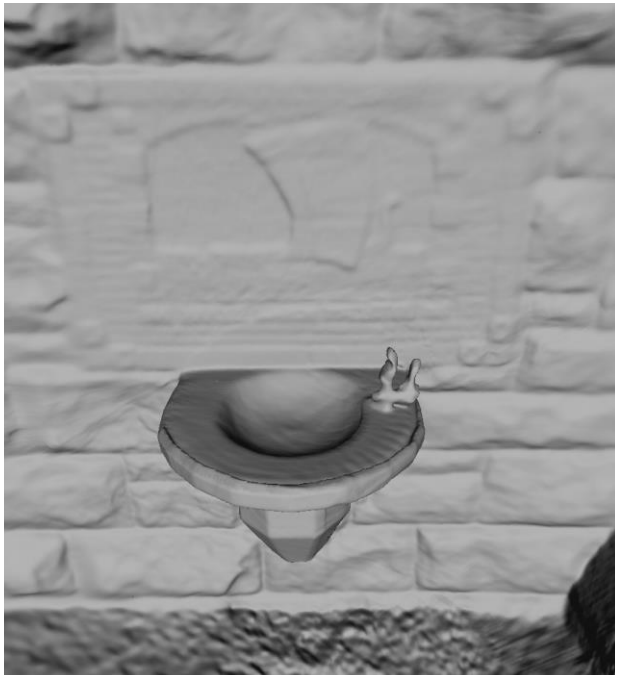
BundleFusion, 2016

State-of-the-art

Voxel Hashing

- Baseline RGB-D based 3D reconstruction framework (initial camera poses and sparse SDF reconstruction): accurate geometric reconstruction, bad colors

Input Frames



Geometry

Colors ☹

"Real-time 3D Reconstruction at Scale using Voxel Hashing", Nießner et al., ToG 2013.

State-of-the-art

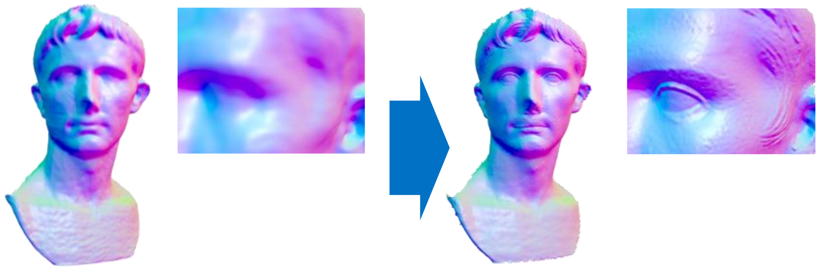
High-Quality Colors [Zhou2014]

Optimize camera poses and image deformations to optimally fit initial (maybe wrong) reconstruction

But: Need HQ image, no geometry refinement involved

“Color Map Optimization for 3D Reconstruction with Consumer Depth Cameras”
Zhou and Koltun, ToG 2014

High-Quality Geometry [Zollhoefer2015]



Adjust camera poses in advance (bundle adjustment) to improve color

Use shading cues (RGB) to refine geometry (shading based refinement of surface & albedo)

But: RGB is fixed (no color refinement based on refined geometry)

“Shading-based Refinement on Volumetric Signed Distance Functions”
Zollhoefer et al., ToG 2015

State-of-the-art

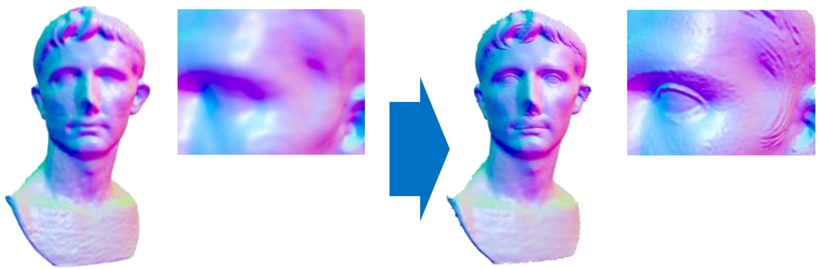
High-Quality Colors [Zhou2014]

Optimize camera poses and image deformations to optimally fit initial (maybe wrong) reconstruction

But: Need HQ image, no geometry refinement involved

"Color Map Optimization for 3D Reconstruction with Consumer Depth Cameras"
Zhou and Koltun, ToG 2014

High-Quality Geometry [Zollhoefer2015]



Adjust camera poses in advance (bundle adjustment) to improve color

Use shading cues (RGB) to refine geometry (shading based refinement of surface & albedo)

But: RGB is fixed (no color refinement based on refined geometry)

"Shading-based Refinement on Volumetric Signed Distance Functions"
Zollhoefer et al., ToG 2015

Idea: jointly optimize for geometry, albedo and image formation model to simultaneously obtain high-quality geometry and appearance!

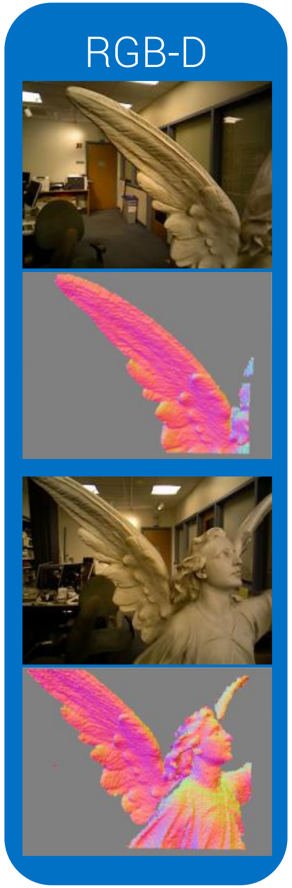
Approach

Approach

Schematic Overview

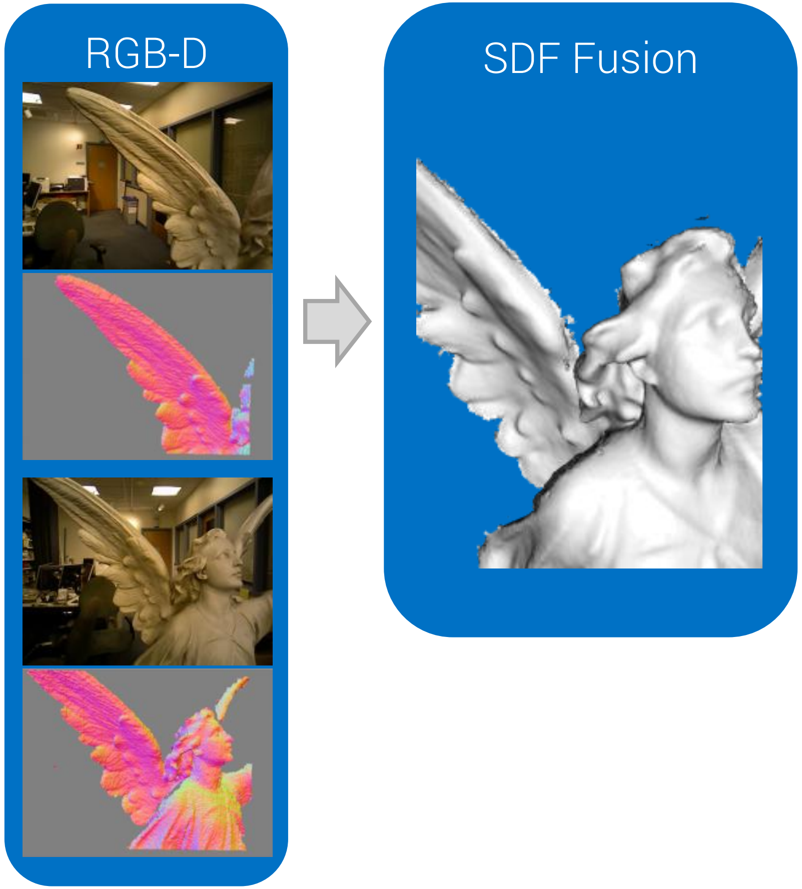
Approach

Schematic Overview



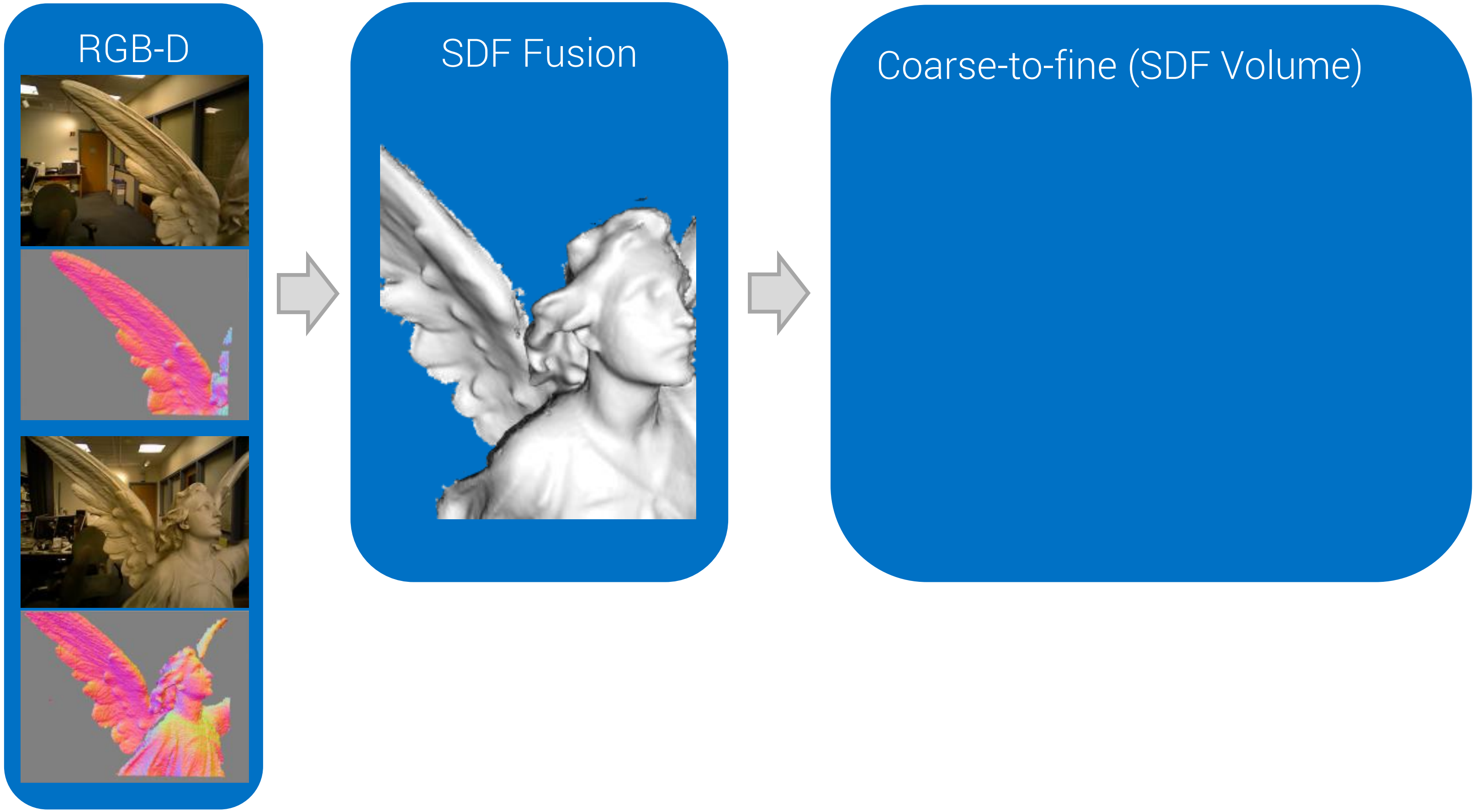
Approach

Schematic Overview



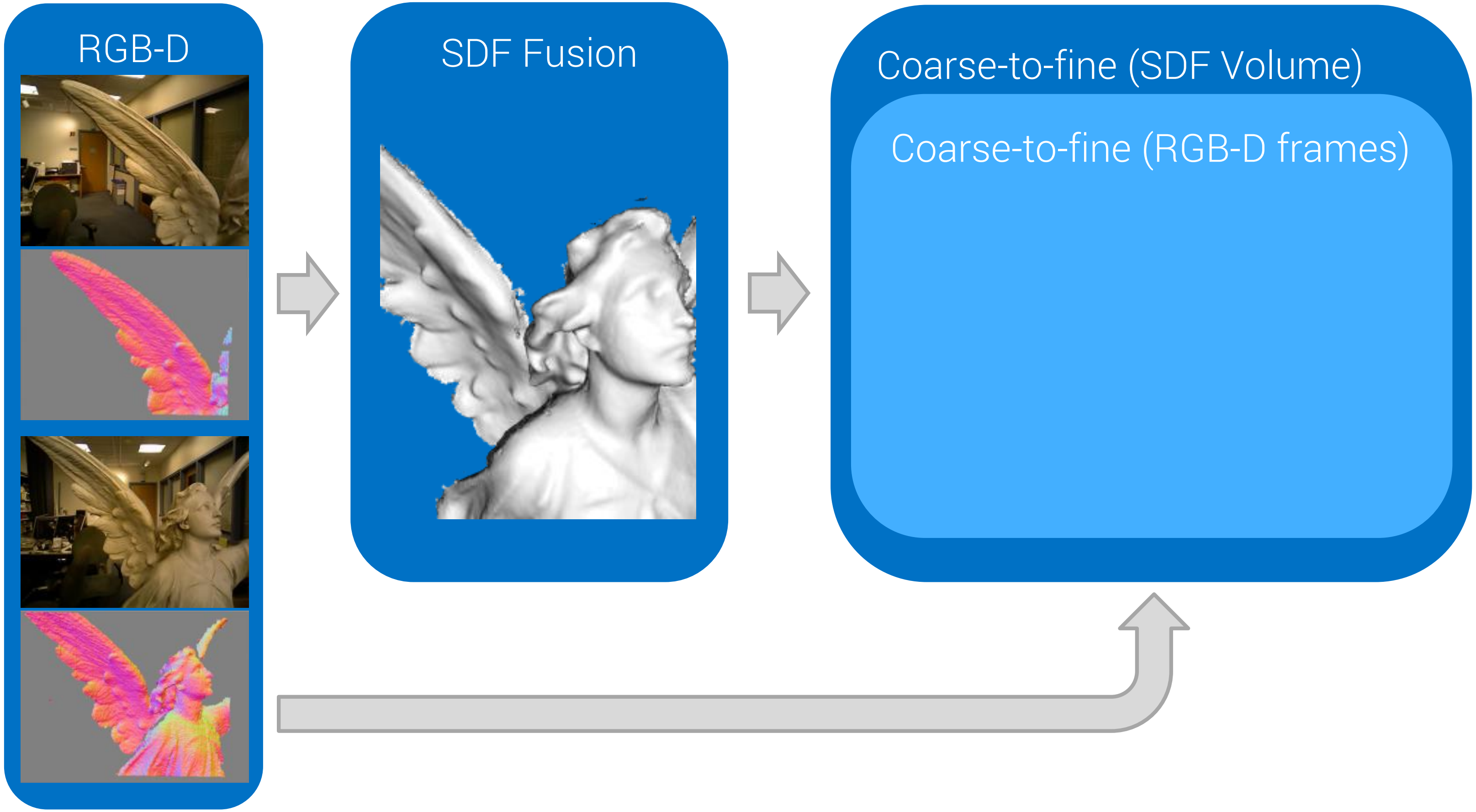
Approach

Schematic Overview



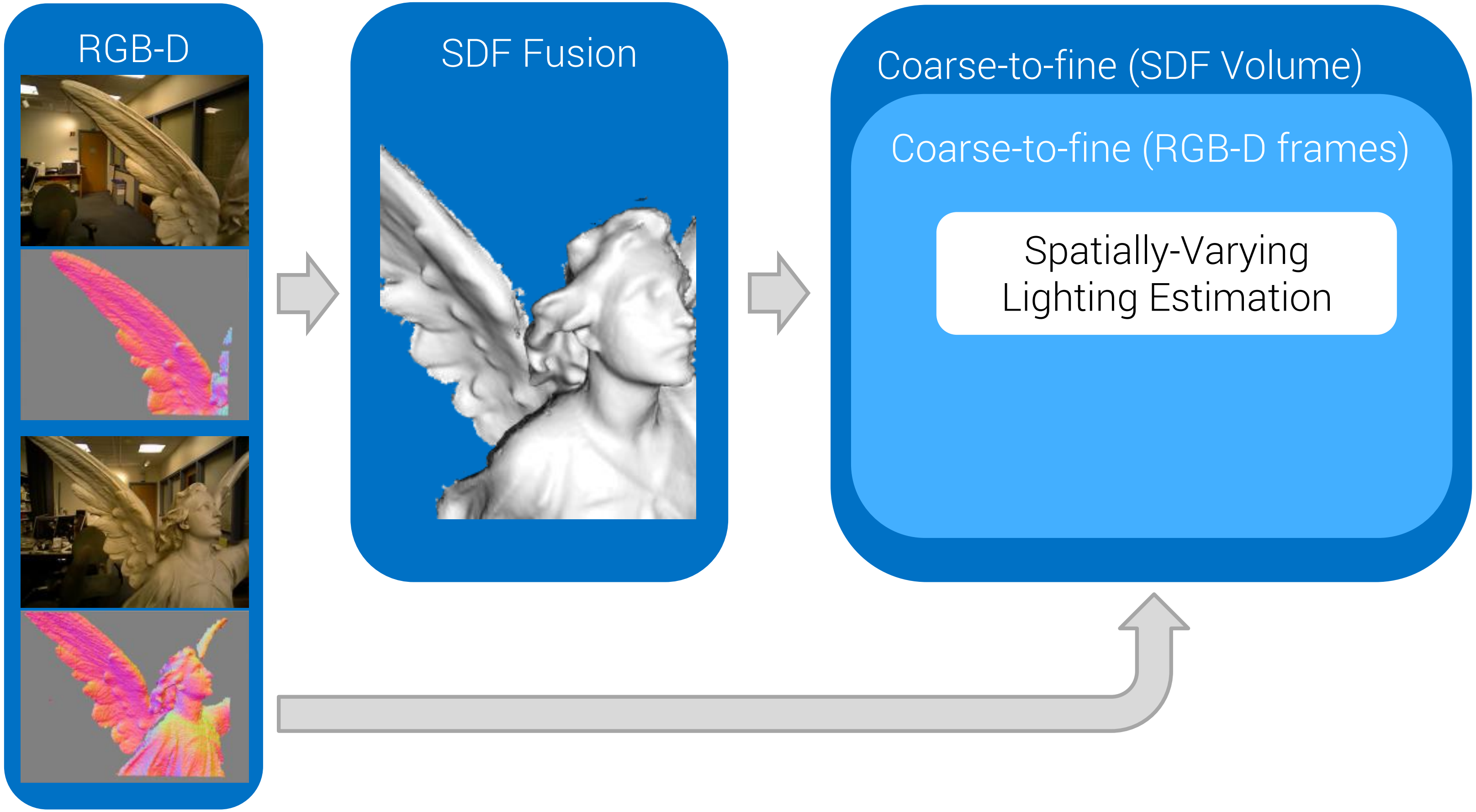
Approach

Schematic Overview



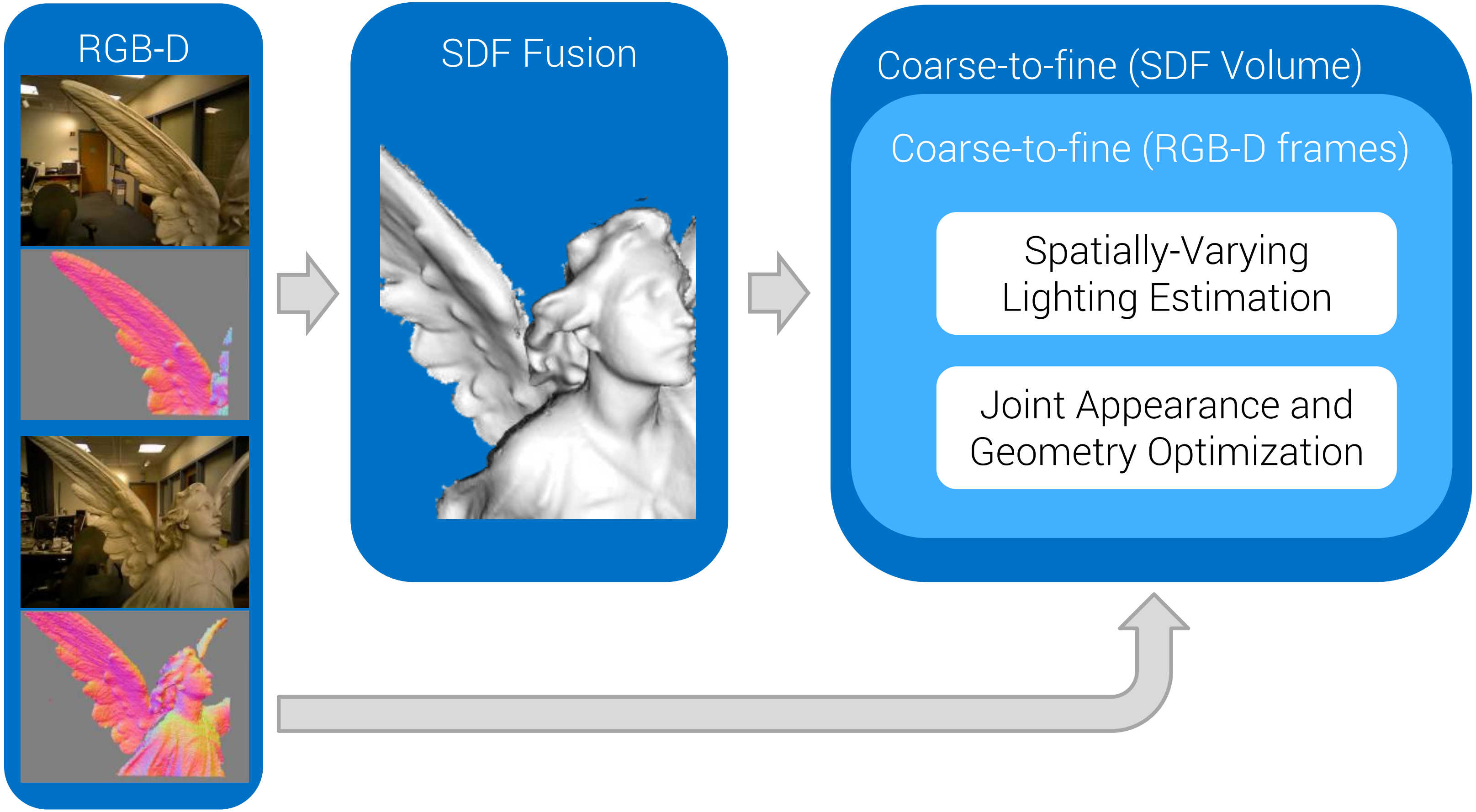
Approach

Schematic Overview



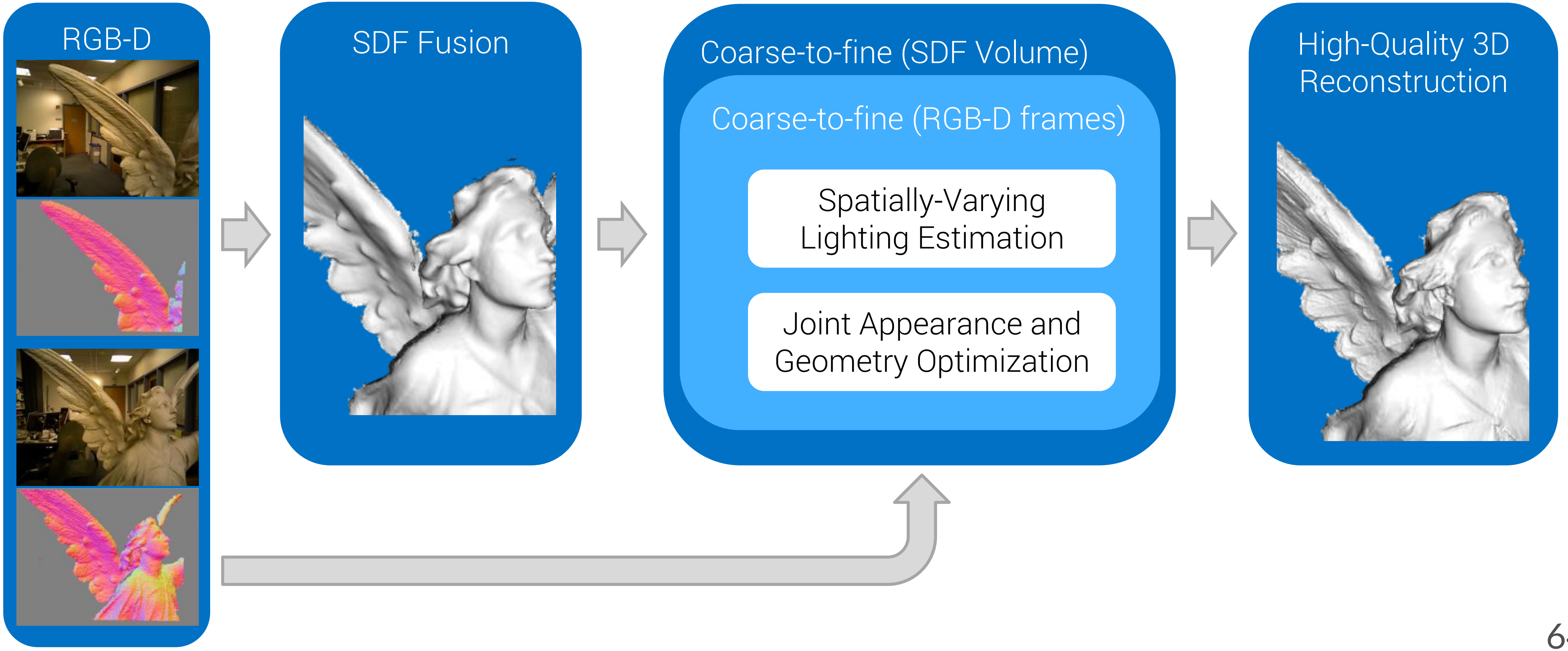
Approach

Schematic Overview



Approach

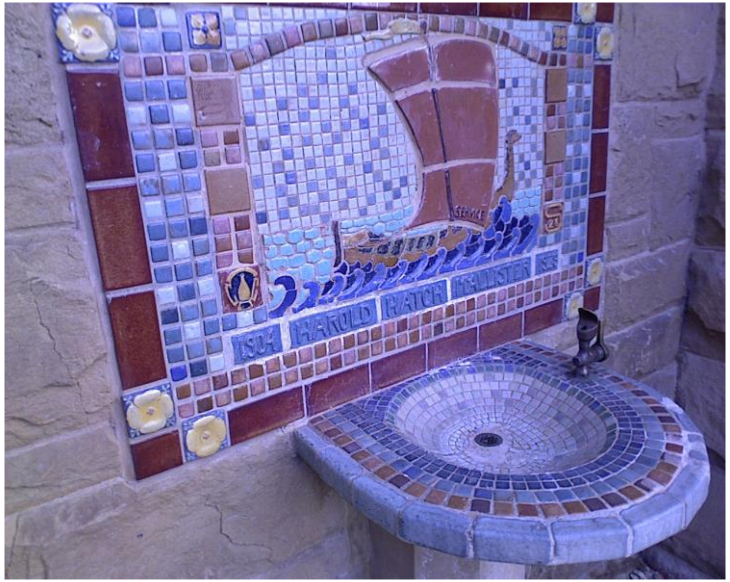
Schematic Overview

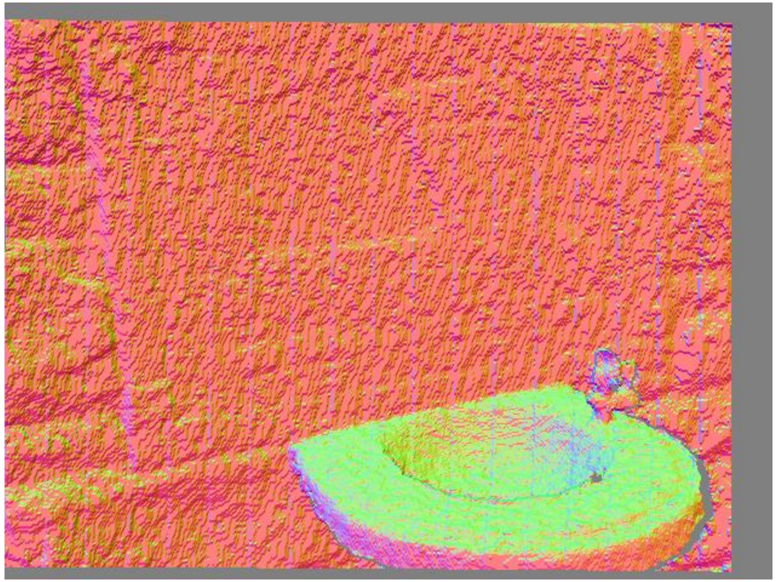


RGB-D Data

Fountain

- 1086 RGB-D frames
- Sensor:
  - Depth 640x480px
  - Color 1280x1024px
  - ~10 Hz
  - Primesense
- Poses estimated using Voxel Hashing





Source: http://qianyi.info/scenedata.html

SDF Fusion

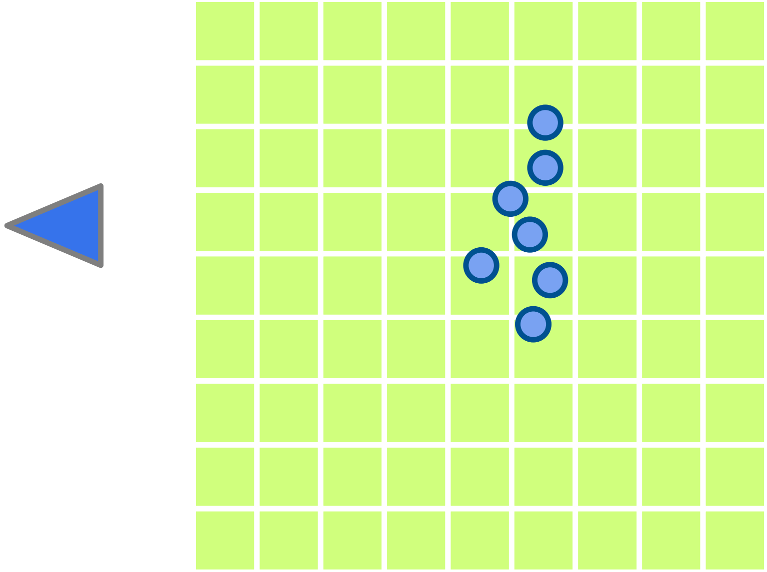
- Volumetric Signed Distance Field (SDF)[1]: 3D voxel grid that stores signed distance to closest surface at each voxel

[1] "A volumetric method for building complex models from range images", Curless and Levoy, SIGGRAPH 1996.

[2] "Marching cubes: A high resolution 3D surface construction algorithm", Lorensen and Cline, SIGGRAPH 1987.

SDF Fusion

- Volumetric Signed Distance Field (SDF)[1]: 3D voxel grid that stores signed distance to closest surface at each voxel
- Integrate depth maps into (sparse) SDF with their estimated camera poses

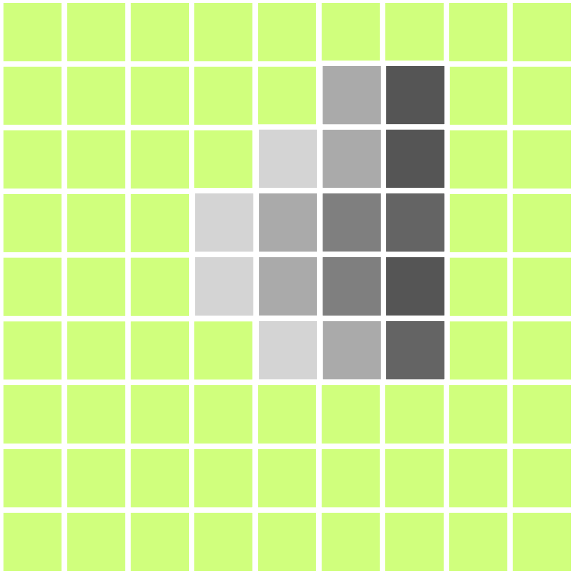


[1] "A volumetric method for building complex models from range images", Curless and Levoy, SIGGRAPH 1996.

[2] "Marching cubes: A high resolution 3D surface construction algorithm", Lorensen and Cline, SIGGRAPH 1987.

SDF Fusion

- Volumetric Signed Distance Field (SDF)[1]: 3D voxel grid that stores signed distance to closest surface at each voxel
- Integrate depth maps into (sparse) SDF with their estimated camera poses

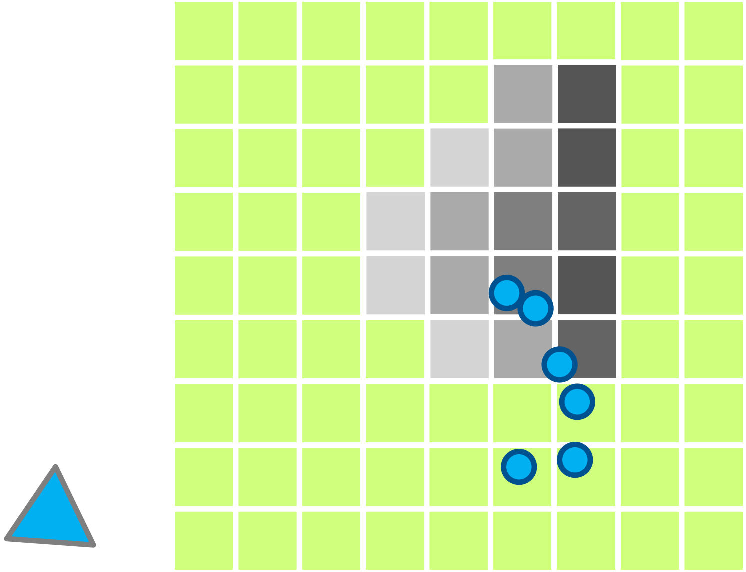


[1] "A volumetric method for building complex models from range images", Curless and Levoy, SIGGRAPH 1996.

[2] "Marching cubes: A high resolution 3D surface construction algorithm", Lorensen and Cline, SIGGRAPH 1987.

SDF Fusion

- Volumetric Signed Distance Field (SDF)[1]: 3D voxel grid that stores signed distance to closest surface at each voxel
- Integrate depth maps into (sparse) SDF with their estimated camera poses

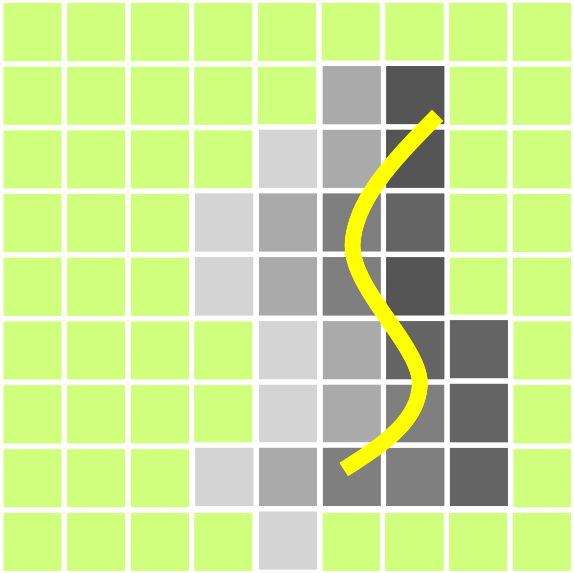


[1] "A volumetric method for building complex models from range images", Curless and Levoy, SIGGRAPH 1996.

[2] "Marching cubes: A high resolution 3D surface construction algorithm", Lorensen and Cline, SIGGRAPH 1987.

SDF Fusion

- Volumetric Signed Distance Field (SDF)[1]: 3D voxel grid that stores signed distance to closest surface at each voxel
- Integrate depth maps into (sparse) SDF with their estimated camera poses
- Extract ISO-surface with Marching Cubes[2] (triangle mesh)



[1] "A volumetric method for building complex models from range images", Curless and Levoy, SIGGRAPH 1996.

[2] "Marching cubes: A high resolution 3D surface construction algorithm", Lorensen and Cline, SIGGRAPH 1987.

Shape-from-Shading

- Shading equation: $$\mathbf{B}(\boldsymbol{v}) = \mathbf{a}(\boldsymbol{v}) \sum_{m=1}^{b^2} l_m H_m(\mathbf{n}(\boldsymbol{v})),$$

Shape-from-Shading

surface normal

- Shading equation: $$\mathbf{B}(\boldsymbol{v}) = \mathbf{a}(\boldsymbol{v}) \sum_{m=1}^{b^2} l_m H_m\left(\boxed{\mathbf{n}(\boldsymbol{v})}\right),$$

Shape-from-Shading

- Shading equation: $$\mathbf{B}(\boldsymbol{v}) = \mathbf{a}(\boldsymbol{v}) \sum_{m=1}^{b^2} \underbrace{\boxed{l_m H}}_{\text{lighting}}_m \overbrace{\boxed{(\mathbf{n}(\boldsymbol{v}))}}^{\text{surface normal}},$$

Shape-from-Shading

- Shading equation: $$\mathbf{B}(\boldsymbol{v}) = \underbrace{\boxed{\mathbf{a}(\boldsymbol{v})}}_{\text{albedo}} \sum_{m=1}^{b^2} \underbrace{\boxed{l_m}}_{\text{lighting}} H_m \underbrace{\boxed{\mathbf{n}(\boldsymbol{v})}}_{\text{surface normal}},$$

Shape-from-Shading

- Shading equation: $$\overset{\text{Shading}}{\boxed{\mathbf{B}(\boldsymbol{v})}} = \overset{\text{albedo}}{\boxed{\mathbf{a}(\boldsymbol{v})}} \sum_{m=1}^{b^2} \underset{\text{lighting}}{\boxed{l_m}} H_m \overset{\text{surface normal}}{\boxed{(\mathbf{n}(\boldsymbol{v}))}},$$

Shape-from-Shading

- Shading equation: $$\overbrace{\boxed{\mathbf{B}(\boldsymbol{v})}}^{\text{Shading}} = \overbrace{\boxed{\mathbf{a}(\boldsymbol{v})}}^{\text{albedo}} \sum_{m=1}^{b^2} \underbrace{\boxed{l_m H}}_{\text{lighting}}{}_m \overbrace{\boxed{(\mathbf{n}(\boldsymbol{v}))}}^{\text{surface normal}},$$

- Shading-based refinement:
  - Estimate lighting given surface and albedo (intrinsic material properties)

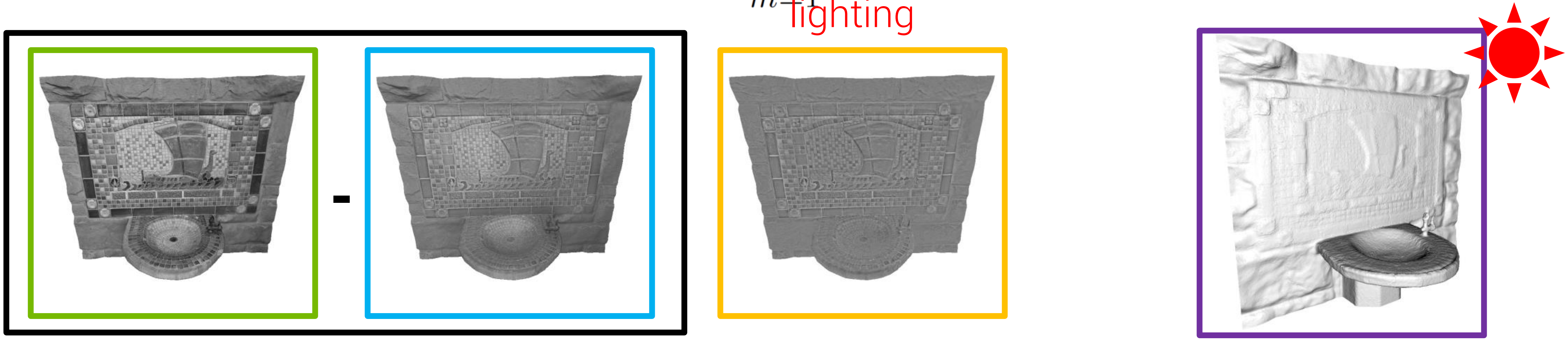
Shape-from-Shading

- Shading equation:

Shading albedo surface normal

$$\boxed{\mathbf{B}(\boldsymbol{v})} = \boxed{\mathbf{a}(\boldsymbol{v})} \sum_{m=1}^{b^2} \boxed{l_m} H_m \boxed{(\mathbf{n}(\boldsymbol{v}))},$$

lighting



- Shading-based refinement:
  - Estimate lighting given surface and albedo (intrinsic material properties)
  - Estimate surface and albedo given the lighting: minimize difference between estimated shading and input luminance

Spatially-Varying Lighting

Subvolume Partitioning

Spatially-Varying Lighting

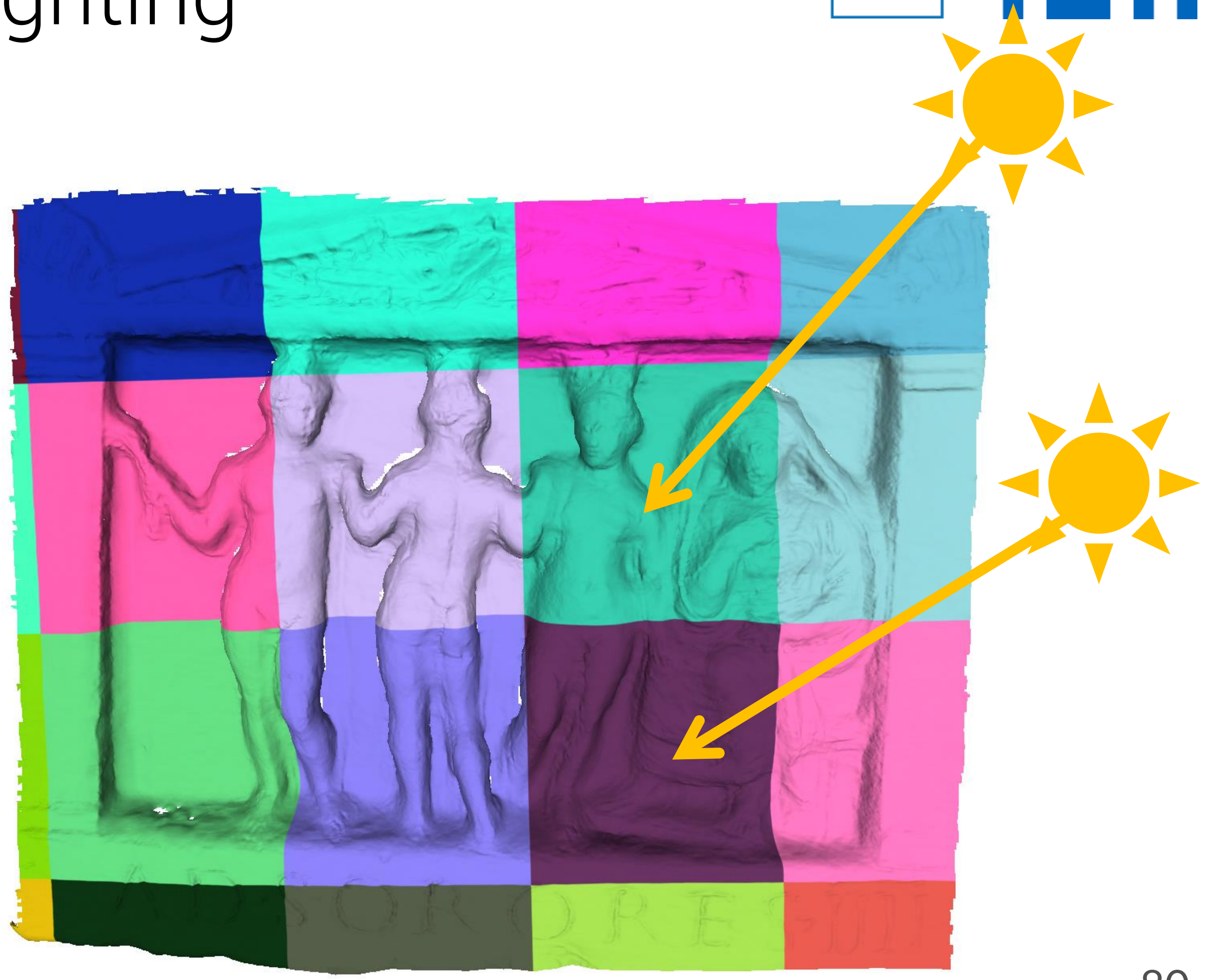
Subvolume Partitioning

- Partition SDF volume into subvolumes of fixed size

Spatially-Varying Lighting

Subvolume Partitioning

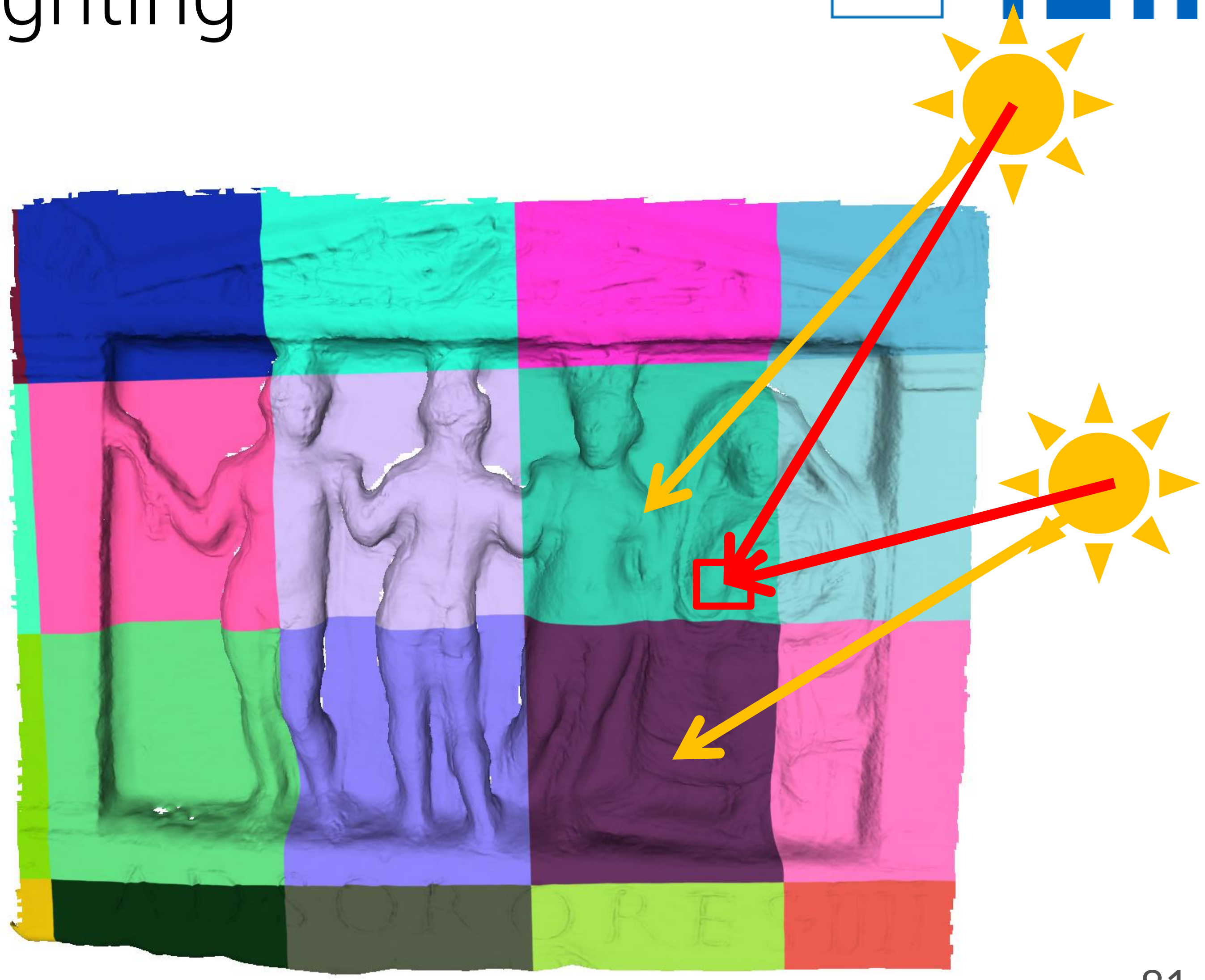
- Partition SDF volume into subvolumes of fixed size
- Estimate independent Spherical Harmonics (SH coefficients) for each subvolume



Spatially-Varying Lighting

Subvolume Partitioning

- Partition SDF volume into subvolumes of fixed size
- Estimate independent Spherical Harmonics (SH coefficients) for each subvolume
- Obtain per-voxel SH coefficients through tri-linear interpolation



Spatially-Varying Lighting

Spatially-Varying Lighting

- Estimate SVSH Coefficients for all subvolumes jointly:

$$E_{\text{lighting}}(\boldsymbol{l}_1, \ldots, \boldsymbol{l}_K) = E_{\text{appearance}} + \lambda_{\text{diffuse}} E_{\text{diffuse}}.$$

Spatially-Varying Lighting

- Estimate SVSH Coefficients for all subvolumes jointly:

$$E_{\text{lighting}}(\boldsymbol{l}_1, \ldots, \boldsymbol{l}_K) = \boxed{E_{\text{appearance}}} + \lambda_{\text{diffuse}} E_{\text{diffuse}}.$$

Data term:

$$E_{\text{appearance}} = \sum_{\boldsymbol{v} \in \mathbf{D}_0} (\mathbf{B}(\boldsymbol{v}) - \mathbf{I}(\boldsymbol{v}))^2.$$

Similarity between estimated shading and input luminance

Spatially-Varying Lighting

- Estimate SVSH Coefficients for all subvolumes jointly:

$$E_{\text{lighting}}(\boldsymbol{l}_1, \ldots, \boldsymbol{l}_K) = \boxed{E_{\text{appearance}}} + \lambda_{\text{diffuse}} \boxed{E_{\text{diffuse}}.}$$

Data term:

$$E_{\text{appearance}} = \sum_{\boldsymbol{v} \in \mathbf{D}_0} (\mathbf{B}(\boldsymbol{v}) - \mathbf{I}(\boldsymbol{v}))^2.$$

Similarity between estimated shading and input luminance

Laplacian regularizer:

$$E_{\text{diffuse}} = \sum_{s \in \mathcal{S}} \sum_{r \in \mathcal{N}_s} (\boldsymbol{l}_s - \boldsymbol{l}_r)^2.$$

Smooth illumination changes

Joint Optimization

Shading-based SDF optimization

- Joint optimization of geometry, albedo and image formation model (camera poses and camera intrinsics):

$$E_{\text{scene}}(\tilde{\mathbf{D}}, \mathbf{a}, \mathcal{T}, f_x, f_y, c_x, c_y) = \sum_{\boldsymbol{v} \text{ s.t. } |\tilde{\mathbf{D}}(\boldsymbol{v})| < t_{\text{shell}}} \lambda_g E_g + \lambda_v E_v + \lambda_s E_s + \lambda_a E_a,$$

Joint Optimization

Shading-based SDF optimization

- Joint optimization of geometry, albedo and image formation model (camera poses and camera intrinsics):

$$E_{\text{scene}}(\tilde{\mathbf{D}}, \mathbf{a}, \mathcal{T}, f_x, f_y, c_x, c_y) = \sum_{\boldsymbol{v} \text{ s.t. } |\tilde{\mathbf{D}}(\boldsymbol{v})| < t_{\text{shell}}} \lambda_g \boxed{E_g} + \lambda_v E_v + \lambda_s E_s + \lambda_a E_a,$$

Gradient-based shading constraint (data term)

Joint Optimization

Shading-based SDF optimization

- Joint optimization of geometry, albedo and image formation model (camera poses and camera intrinsics):

$$E_{\text{scene}}(\tilde{\mathbf{D}}, \mathbf{a}, \mathcal{T}, f_x, f_y, c_x, c_y) = \sum_{\boldsymbol{v} \text{ s.t. } |\tilde{\mathbf{D}}(\boldsymbol{v})| < t_{\text{shell}}} \lambda_g \boxed{E_g} + \lambda_v \boxed{E_v} + \lambda_s E_s + \lambda_a E_a ,$$

Gradient-based shading constraint (data term)

Volumetric regularizer: smoothness in distance values (Laplacian)

Joint Optimization

Shading-based SDF optimization

- Joint optimization of geometry, albedo and image formation model (camera poses and camera intrinsics):

$$E_{\text{scene}}(\tilde{\mathbf{D}}, \mathbf{a}, \mathcal{T}, f_x, f_y, c_x, c_y) = \sum_{\boldsymbol{v} \text{ s.t. } |\tilde{\mathbf{D}}(\boldsymbol{v})| < t_{\text{shell}}} \lambda_g \boxed{E_g} + \lambda_v \boxed{E_v} + \lambda_s \boxed{E_s} + \lambda_a E_a ,$$

Gradient-based shading constraint (data term)

Volumetric regularizer: smoothness in distance values (Laplacian)

Surface Stabilization constraint: stay close to initial distance values

Joint Optimization

Shading-based SDF optimization

- Joint optimization of geometry, albedo and image formation model (camera poses and camera intrinsics):

$$E_{\text{scene}}(\tilde{\mathbf{D}}, \mathbf{a}, \mathcal{T}, f_x, f_y, c_x, c_y) = \sum_{\boldsymbol{v} \text{ s.t. } |\tilde{\mathbf{D}}(\boldsymbol{v})| < t_{\text{shell}}} \lambda_g \boxed{E_g} + \lambda_v \boxed{E_v} + \lambda_s \boxed{E_s} + \lambda_a \boxed{E_a},$$

Gradient-based shading constraint (data term)

Volumetric regularizer: smoothness in distance values (Laplacian)

Surface Stabilization constraint: stay close to initial distance values

Albedo regularizer: constrain albedo changes based on chromacity (Laplacian)

Shading Constraint

(1) Keyframe Selection

- Compute per-frame blur score (for color image) [Crete2007])

Frame 81

Frame 84

- Select frame with best score within a fixed size window as keyframe

[Crete2007] "The blur effect: perception and estimation with a new no-reference perceptual blur metric", Crete et al., SPIE 2007.

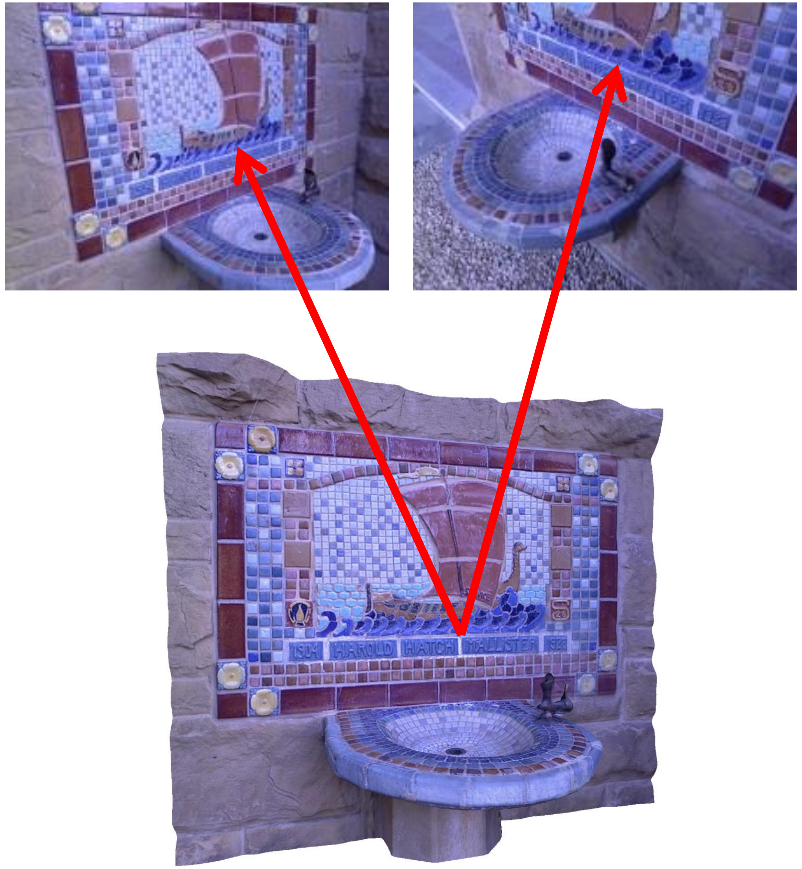
Shading Constraint

(2) Sampling & Colorization

- Sample from selected keyframes only
- Collect observations for voxel in input views:

$$c_i^v = \mathcal{C}_i(\pi(\mathcal{T}_i^{-1} v_{\text{iso}})).$$

Input keyframes



Reconstruction

Shading Constraint

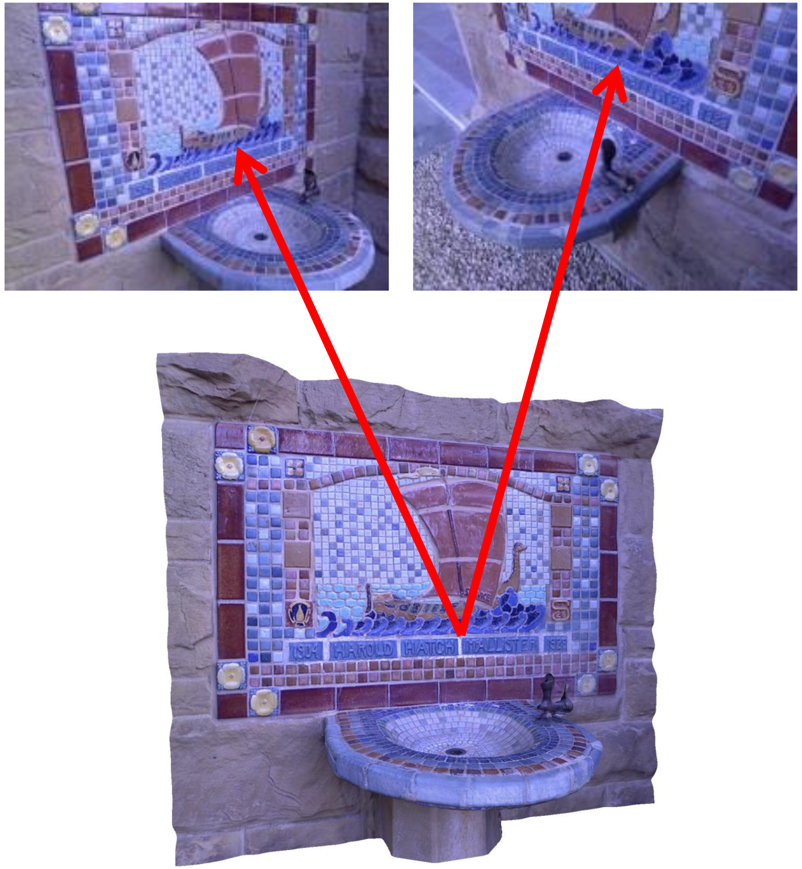
(2) Sampling & Colorization

- Sample from selected keyframes only
- Collect observations for voxel in input views:

$$c_i^v = \mathcal{C}_i\left(\boxed{\pi\left(\mathcal{T}_i^{-1} v_{\text{iso}}\right)}\right).$$

Voxel center transformed and projected into input view

Input keyframes



Reconstruction

Shading Constraint

(2) Sampling & Colorization

- Sample from selected keyframes only
- Collect observations for voxel in input views:

$$c_i^v = \mathcal{C}_i\left(\boxed{\pi\left(\mathcal{T}_i^{-1} \boldsymbol{v}_{\text{iso}}\right)}\right).$$

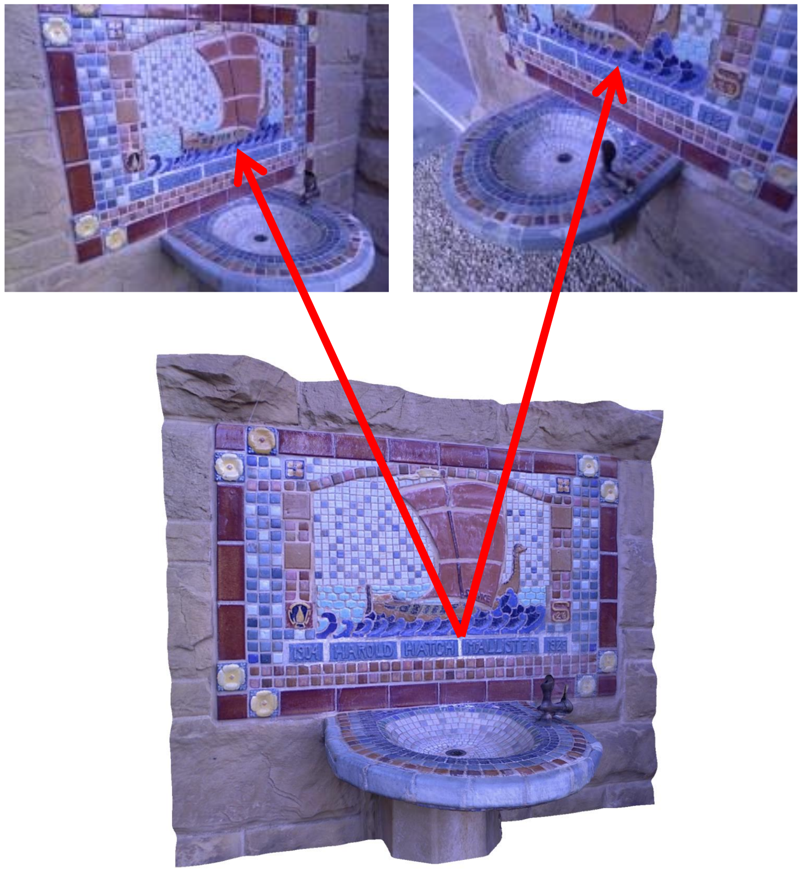
Voxel center transformed and projected into input view

- Observation weights: view-dependent on normal and depth

$$w_i^{\boldsymbol{v}} = \frac{\cos(\theta)}{d^2}$$

- Filter observations: keep only best 5 observations by weight

Input keyframes



Reconstruction

Shading Constraint

(3) Data Term

- Intuition: high-frequency changes in surface geometry result in shading cues in input images

Shading Constraint

(3) Data Term

- Intuition: high-frequency changes in surface geometry result in shading cues in input images
- Idea: maximize consistency between estimated voxel shading and sampled intensities in input luminance images (gradient for robustness)

$$E_g(\boldsymbol{v}) = \sum_{\mathcal{I}_i \in \varsigma} w_i^{\boldsymbol{v}} \|\nabla \mathbf{B}(\boldsymbol{v}) - \nabla \mathcal{I}_i(\pi(v_c))\|_2^2$$

Shading Constraint

(3) Data Term

- Intuition: high-frequency changes in surface geometry result in shading cues in input images
- Idea: maximize consistency between estimated voxel shading and sampled intensities in input luminance images (gradient for robustness)

$$E_g(\boldsymbol{v}) = \sum_{\boxed{\mathcal{I}_i \in \varsigma}} \boxed{w_i^{\boldsymbol{v}} \|} \nabla \mathbf{B}(\boldsymbol{v}) - \nabla \mathcal{I}_i(\pi(v_c)) \|_2^2$$

Best views for voxel and respective view-dependent weights

Shading Constraint

(3) Data Term

- Intuition: high-frequency changes in surface geometry result in shading cues in input images
- Idea: maximize consistency between estimated voxel shading and sampled intensities in input luminance images (gradient for robustness)

$$E_g(\boldsymbol{v}) = \sum_{\boxed{\mathcal{I}_i \in \varsigma}} \boxed{w_i^{\boldsymbol{v}}} \| \nabla \mathbf{B}(\boldsymbol{v}) - \nabla \mathcal{I}_i \boxed{(\pi(v_c))} \|_2^2$$

Best views for voxel and respective view-dependent weights

Voxel center transformed and projected into input view

Shading Constraint

(3) Data Term

- Intuition: high-frequency changes in surface geometry result in shading cues in input images
- Idea: maximize consistency between estimated voxel shading and sampled intensities in input luminance images (gradient for robustness)

$$E_g(\boldsymbol{v}) = \sum_{\boxed{\mathcal{I}_i \in \varsigma}} \boxed{w_i^{\boldsymbol{v}}} \| \nabla \boxed{\mathbf{B}(\boldsymbol{v})} - \nabla \mathcal{I}_i \boxed{(\pi(v_c))} \|_2^2$$

Best views for voxel and respective view-dependent weights

Voxel center transformed and projected into input view

Shading: allows for optimization of surface (through normal) and albedo

Shading Constraint

(3) Data Term

- Intuition: high-frequency changes in surface geometry result in shading cues in input images
- Idea: maximize consistency between estimated voxel shading and sampled intensities in input luminance images (gradient for robustness)

$$E_g(\boldsymbol{v}) = \sum_{\boxed{\mathcal{I}_i \in \varsigma}} \boxed{w_i^{\boldsymbol{v}}} \| \nabla \boxed{\mathbf{B}(\boldsymbol{v})} - \nabla \boxed{\mathcal{I}_i \boxed{(\pi(v_c))}} \|_2^2$$

Best views for voxel and respective view-dependent weights

Voxel center transformed and projected into input view

Shading: allows for optimization of surface (through normal) and albedo

Sampling: allows for optimization of camera poses and camera intrinsics

Recolorization

Optimal colors

- Recompute voxel colors after optimization at each level

Recolorization

Optimal colors

- Recompute voxel colors after optimization at each level
- Sampling (see Shading Constraint)
  - Sample from keyframes only
  - Collect, weight and filter observations

Recolorization

Optimal colors

- Recompute voxel colors after optimization at each level
- Sampling (see Shading Constraint)
  - Sample from keyframes only
  - Collect, weight and filter observations
- Weighted average of observations:

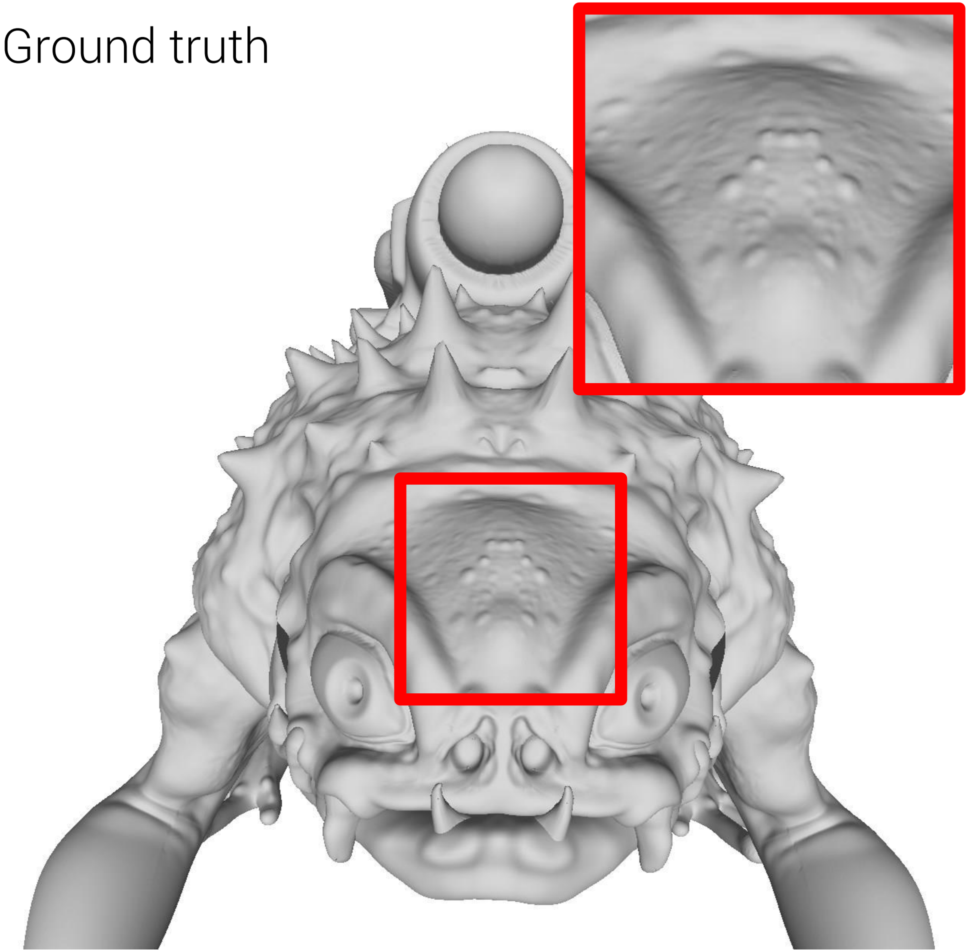
$$c_{\boldsymbol{v}}^{*} = \arg\min_{c_{\boldsymbol{v}}} \sum_{(c_i^{\boldsymbol{v}}, w_i^{\boldsymbol{v}}) \in \mathcal{O}_v} w_i^{\boldsymbol{v}} (c_{\boldsymbol{v}} - c_i^{\boldsymbol{v}})^2 .$$

Results

Geometry

Frog (synthetic)

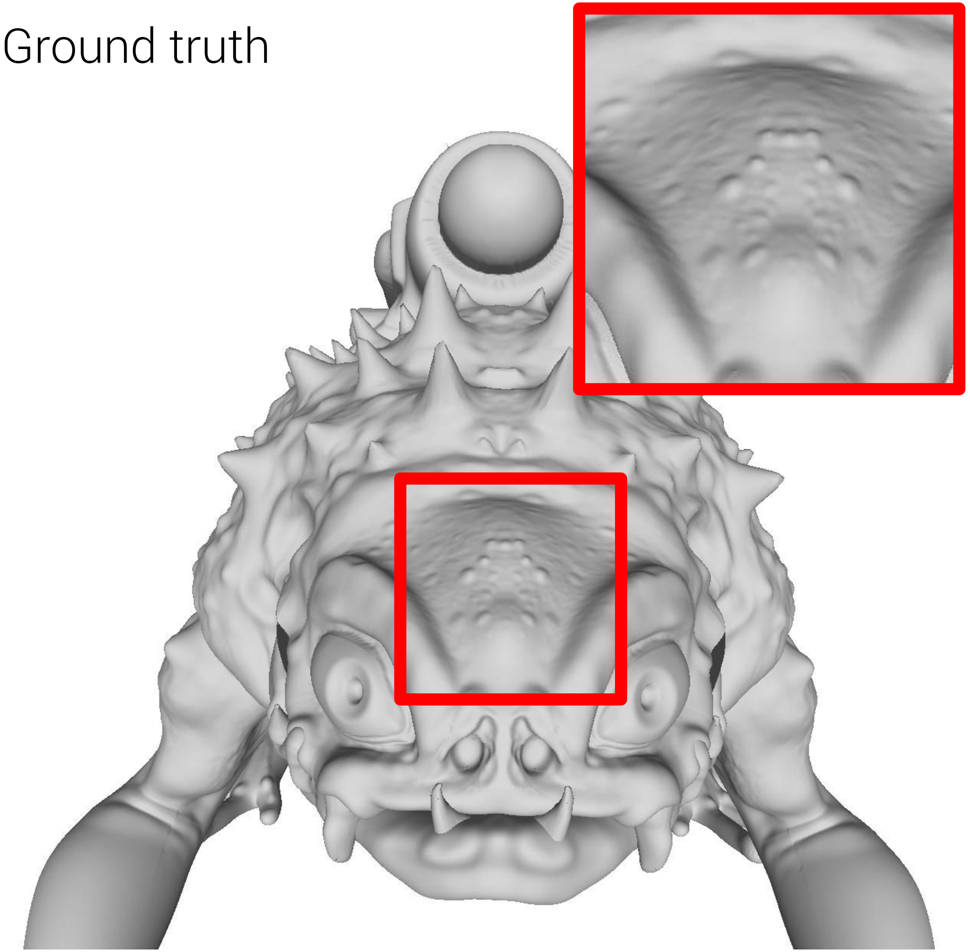
Ground truth



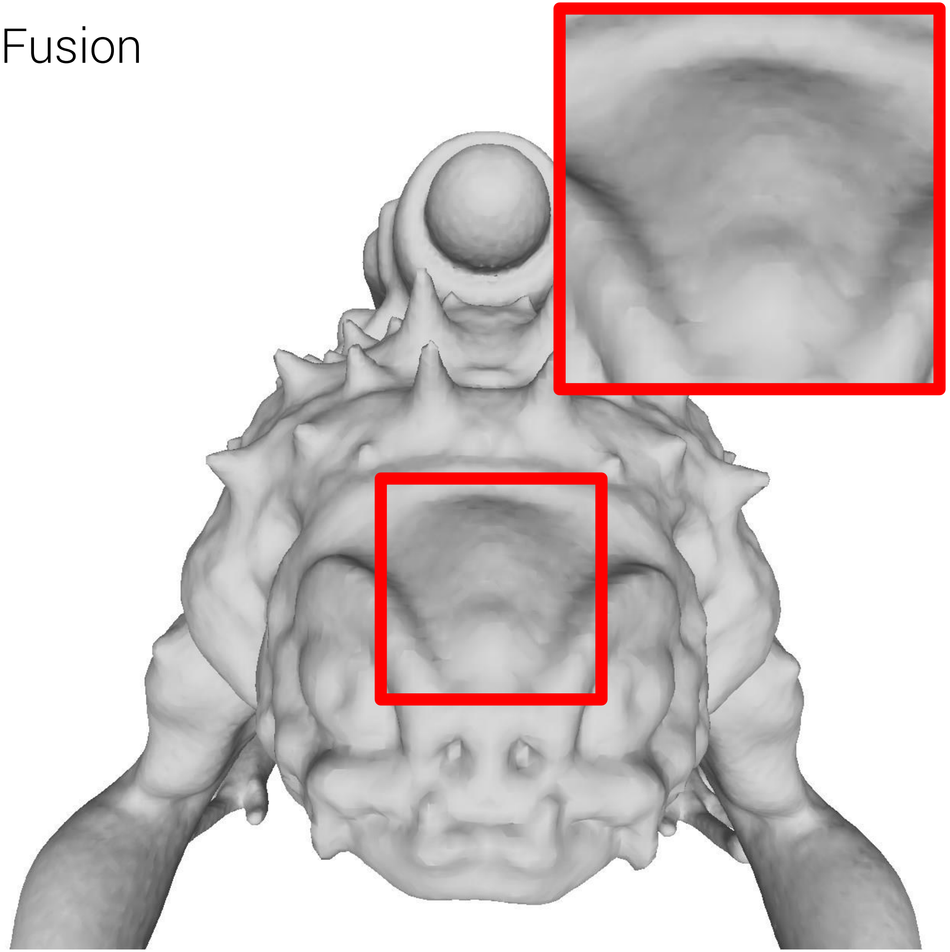
Geometry

Frog (synthetic)

Ground truth

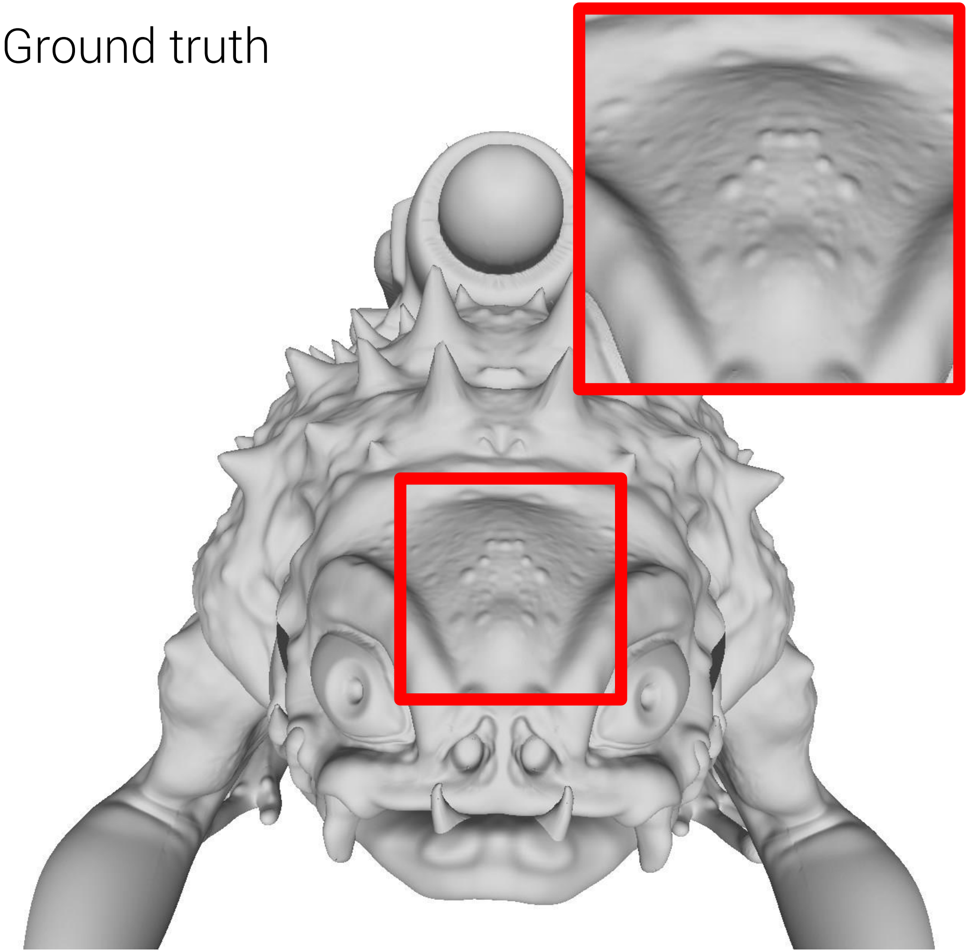


Fusion

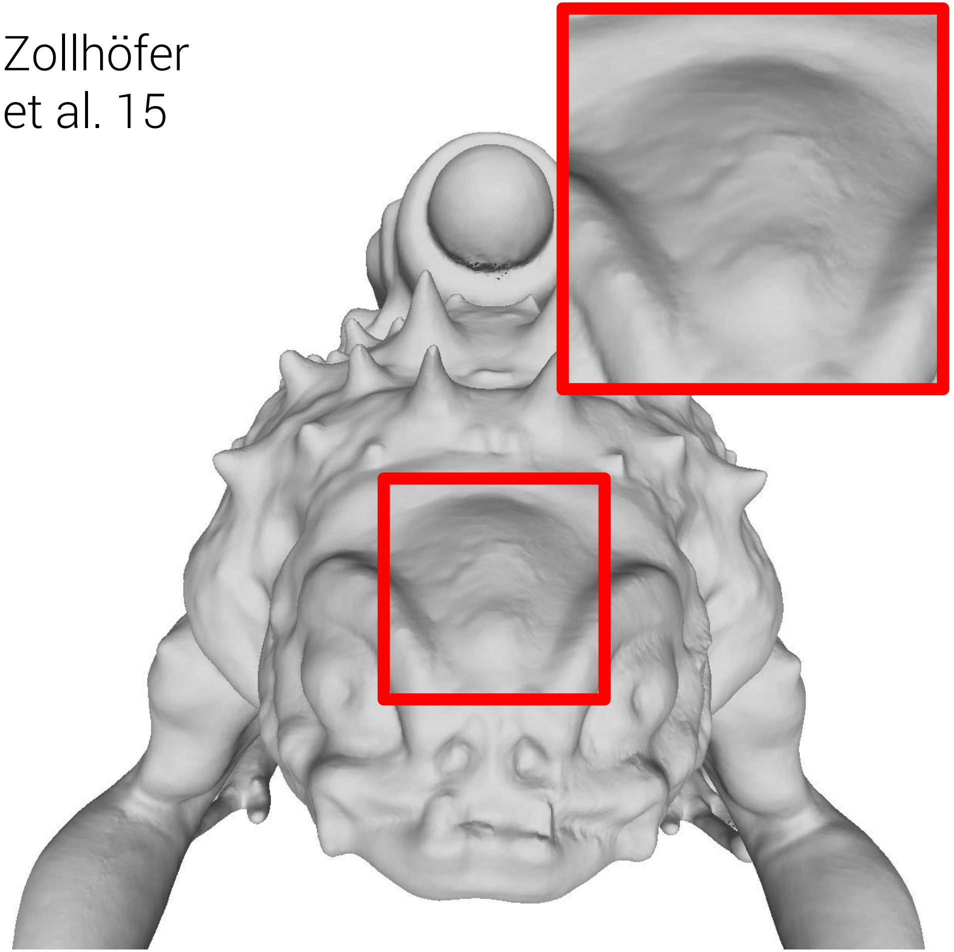


Geometry
Frog (synthetic)

Ground truth



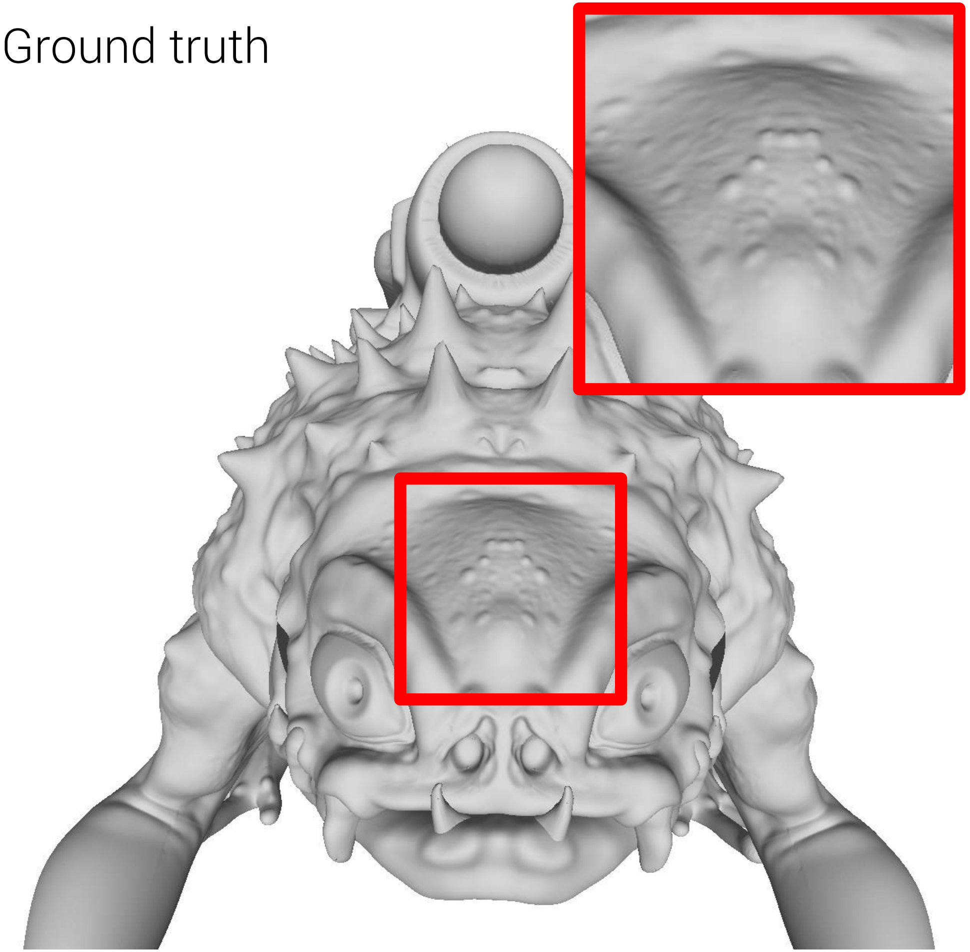
Zollhöfer
et al. 15



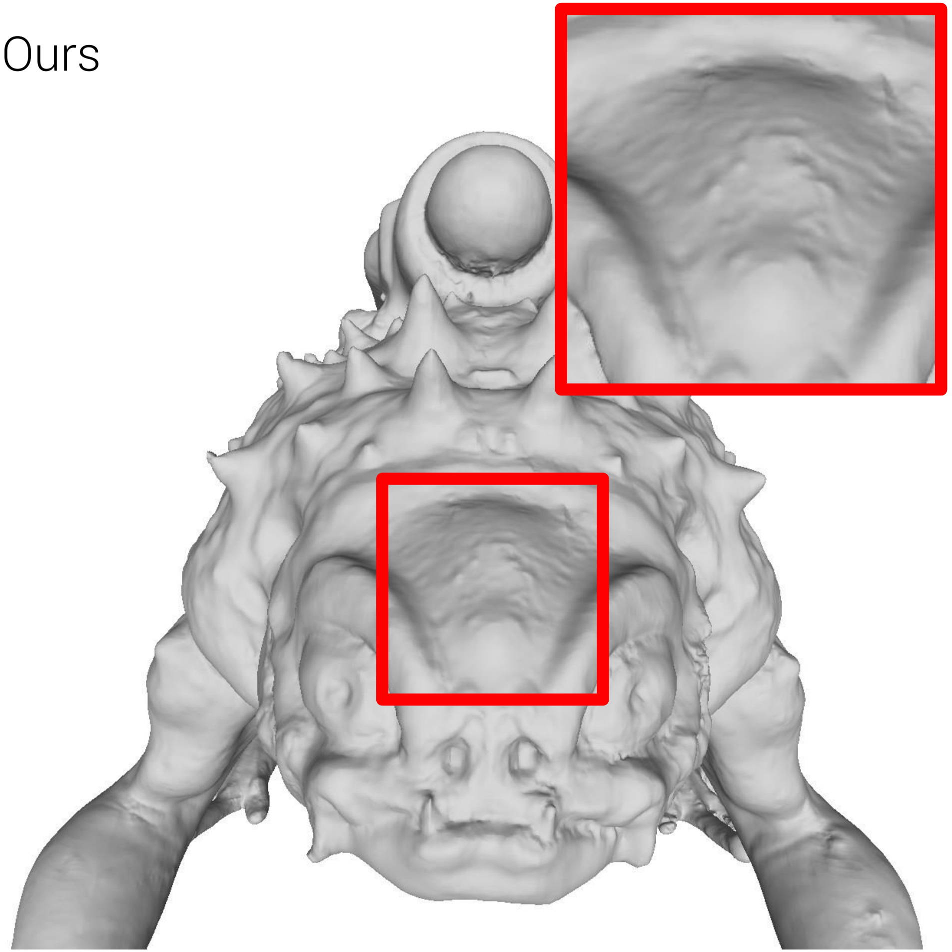
Geometry

Frog (synthetic)

Ground truth



Ours

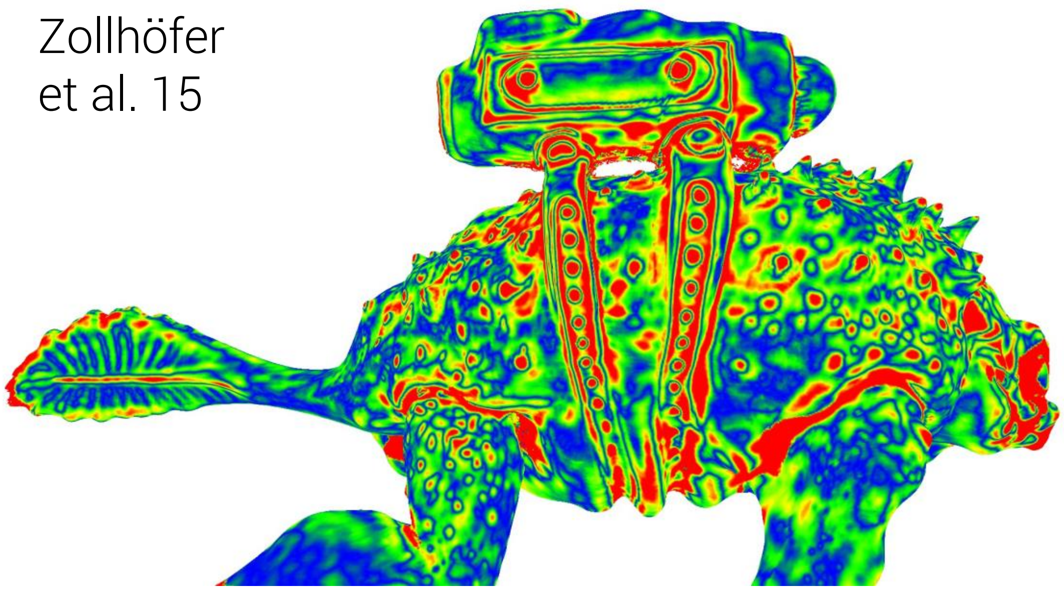


Geometry

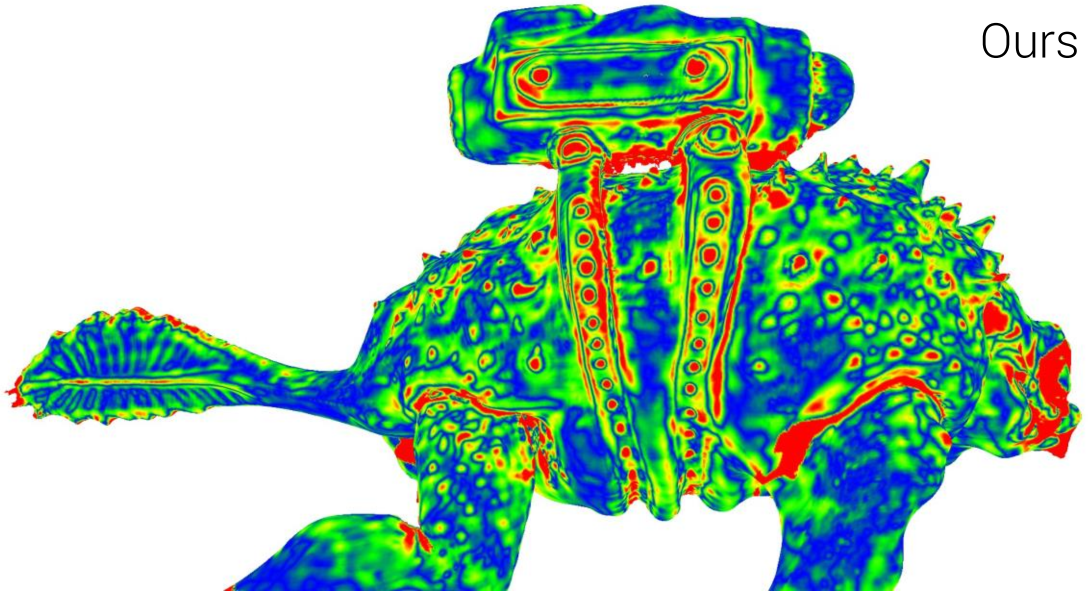
Frog (synthetic)

- Quantitative surface accuracy evaluation
- Color coding: absolute distances (ground truth)

Zollhöfer
et al. 15



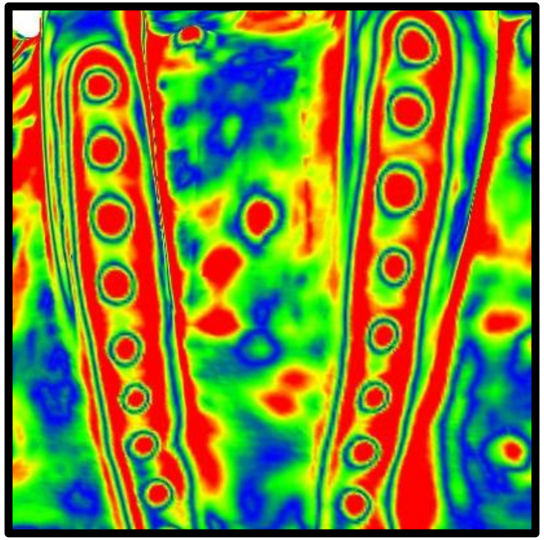
Ours



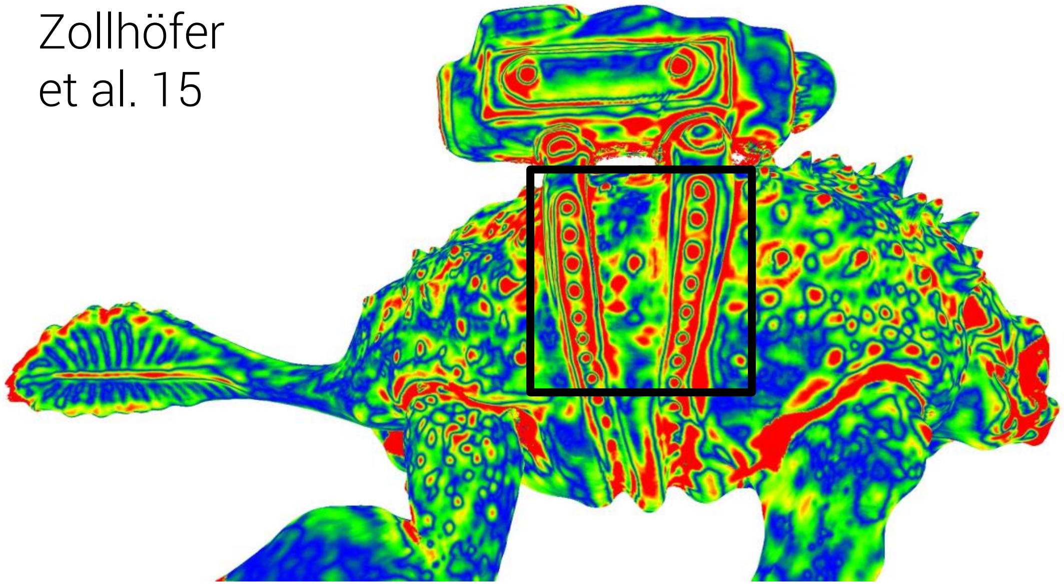
Geometry

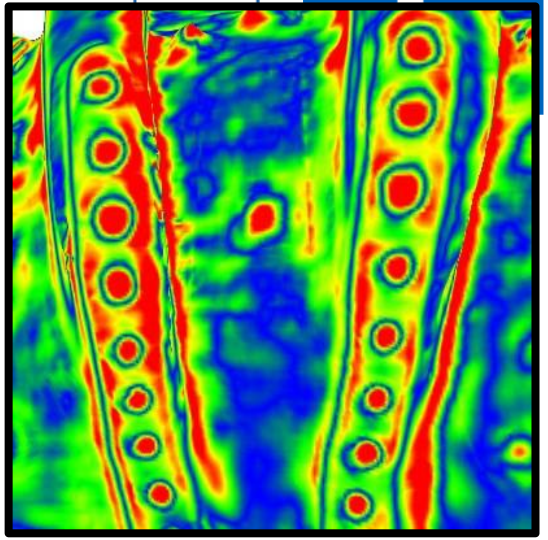
Frog (synthetic)

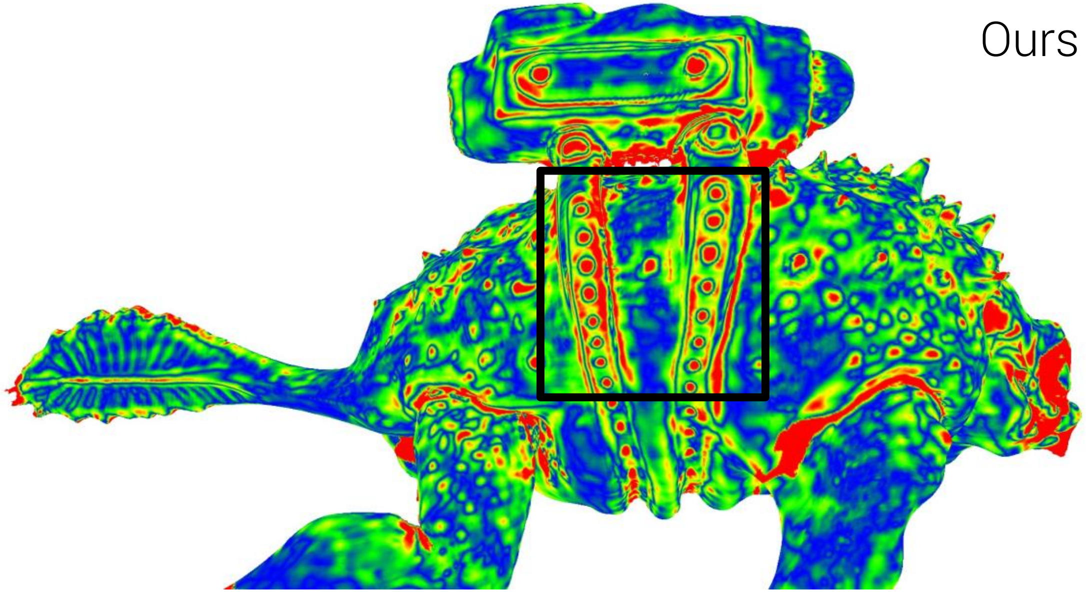
- Quantitative surface accuracy evaluation
- Color coding: absolute distances (ground truth)



Zollhöfer
et al. 15





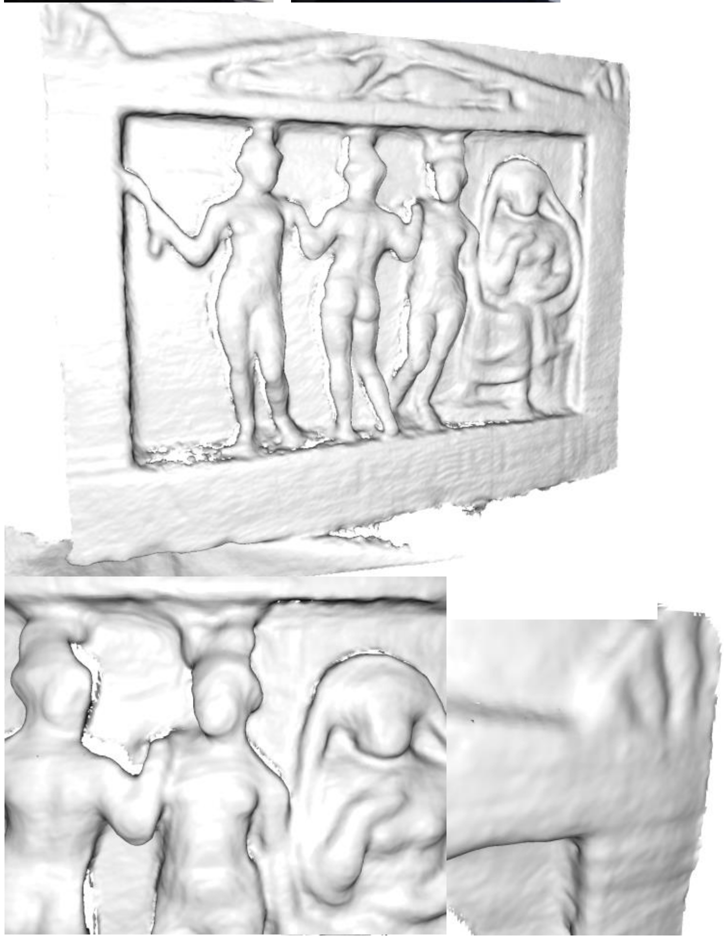


Ours

Input Color

Geometry

Relief

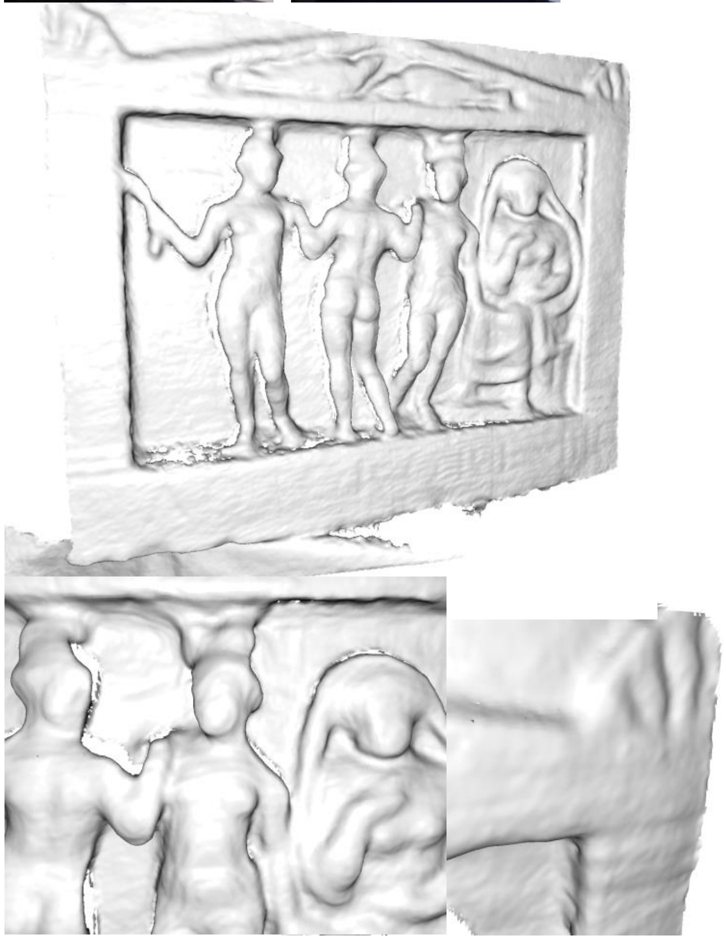


Fusion

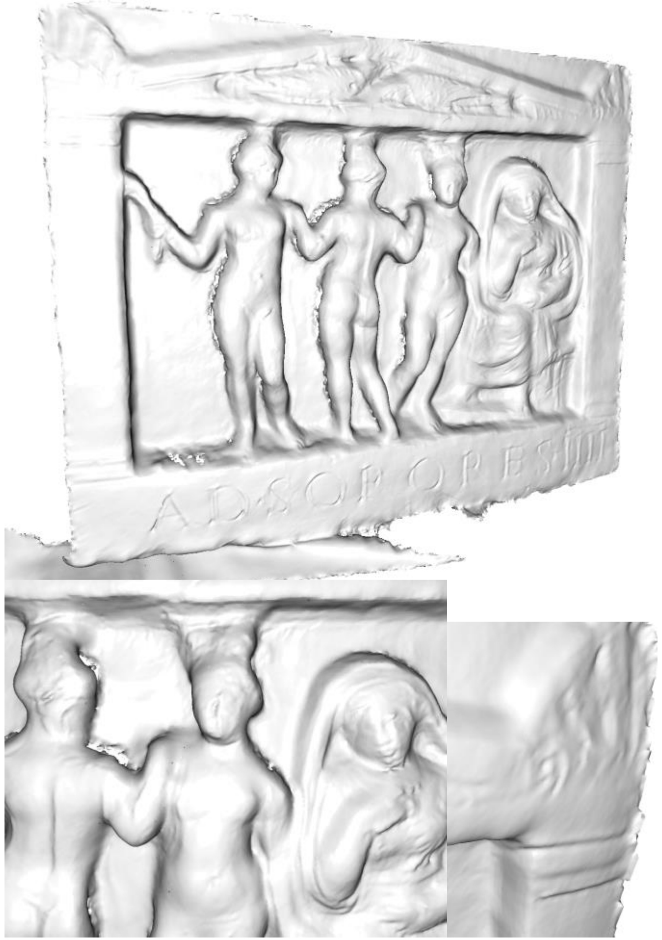
Geometry

Relief

Input Color



Fusion

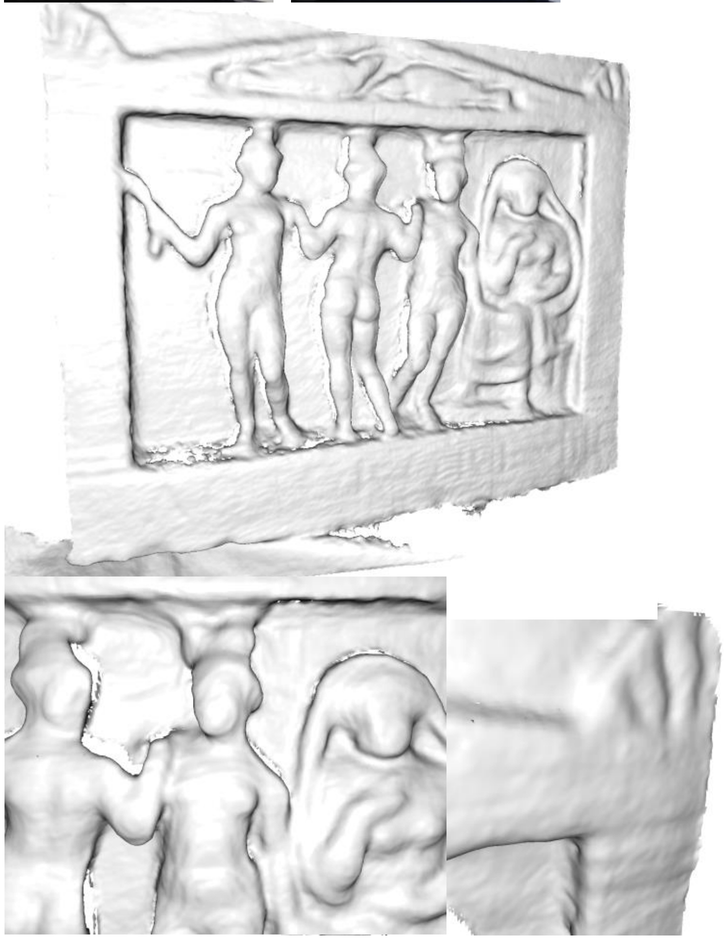


Zollhöfer et al. 15

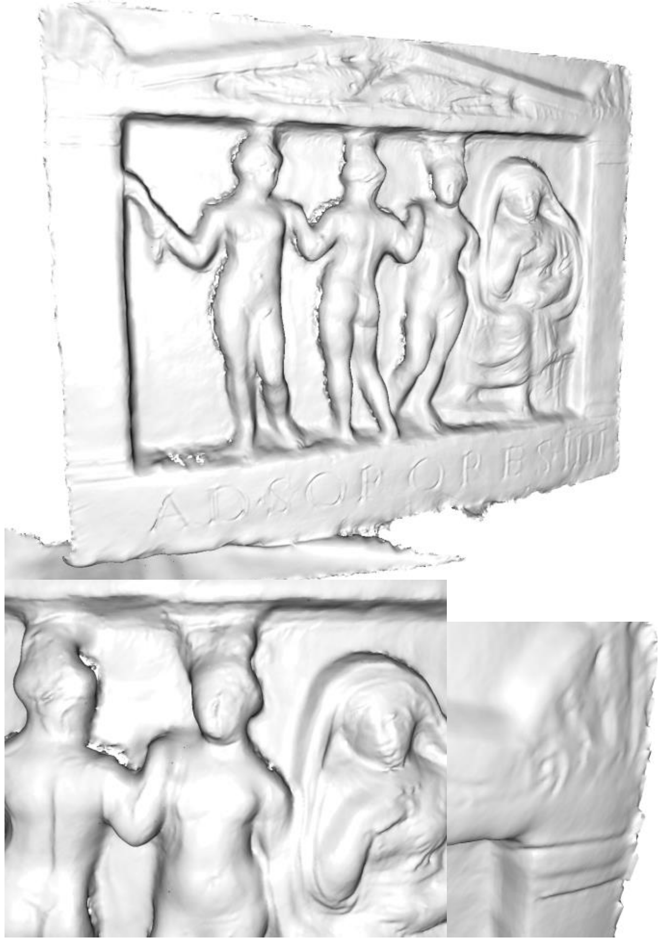
Input Color

Geometry

Relief



Fusion



Zollhöfer et al. 15

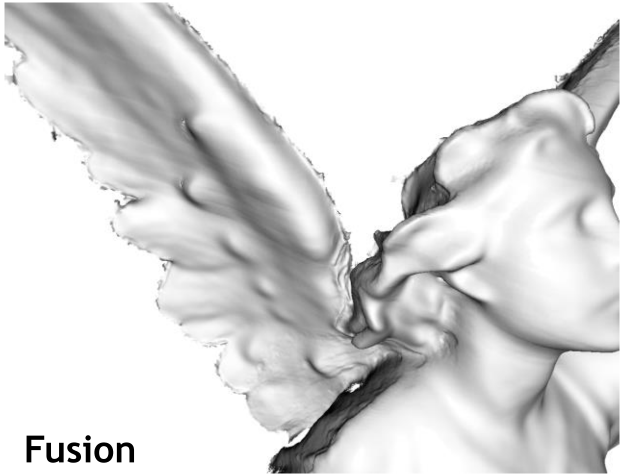
Ours

Geometry

Lucy

Input Color

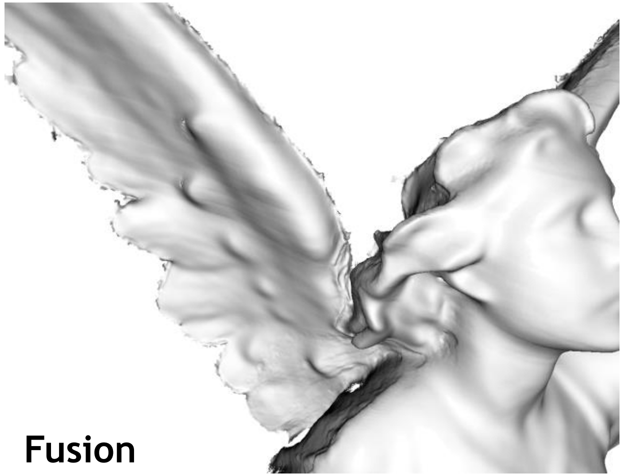
Fusion



Geometry

Lucy

Input Color



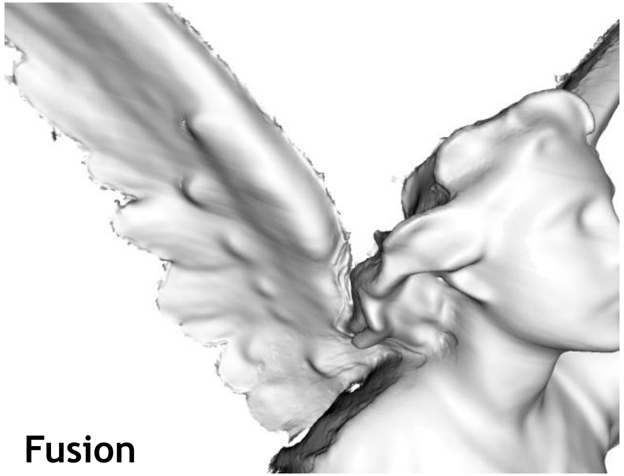
Fusion

Zollhöfer et al. 15

Geometry

Lucy

Input Color



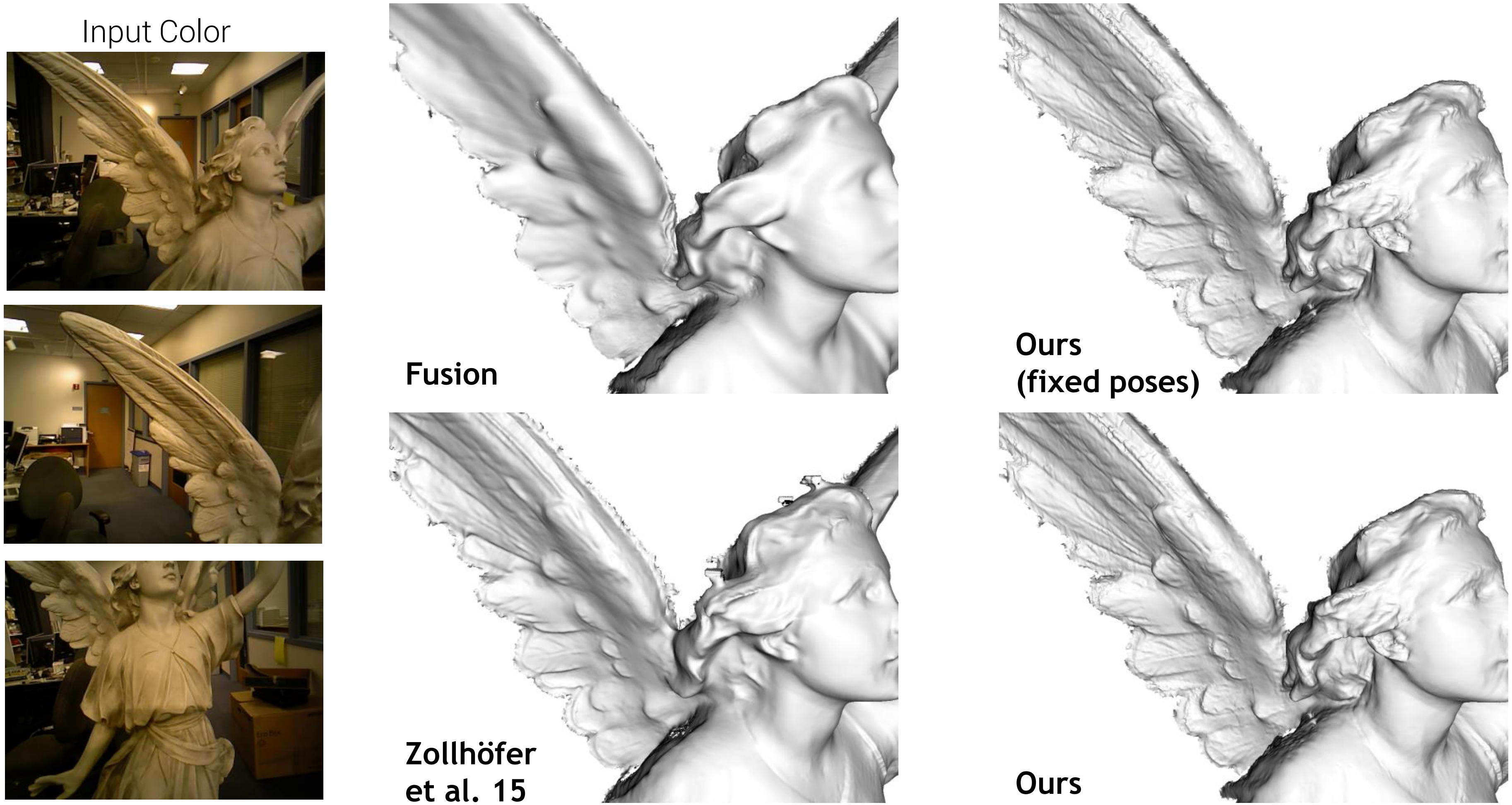
Fusion

Ours (fixed poses)

Zollhöfer et al. 15

Geometry

Lucy



Input Color

Ours

Appearance

Fountain

Appearance

Fountain

Input Color

Ours

Fusion

Appearance

Fountain

Input Color

Ours

Fusion

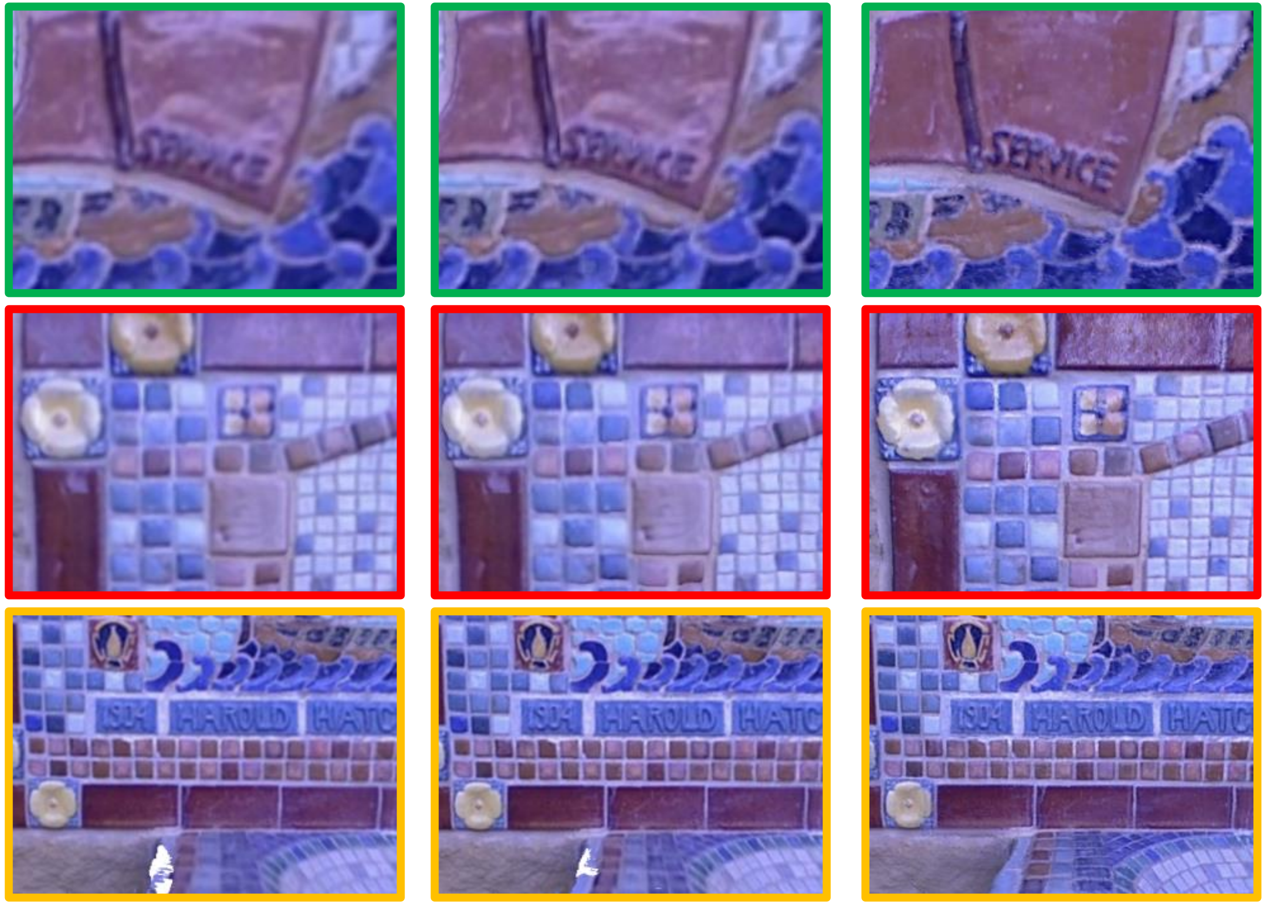
Zollhöfer et al. 15

Appearance

Fountain

Input Color

Ours



Fusion | Zollhöfer et al. 15 | Ours

Appearance
Relief

Fusion

Appearance

Relief

Fusion

Zollhöfer et al. 15

Appearance
Relief

Fusion

Zollhöfer et al. 15

Ours

Shading: Global SH vs. SVSH

Fountain

Luminance

Shading: Global SH vs. SVSH

Fountain

Luminance

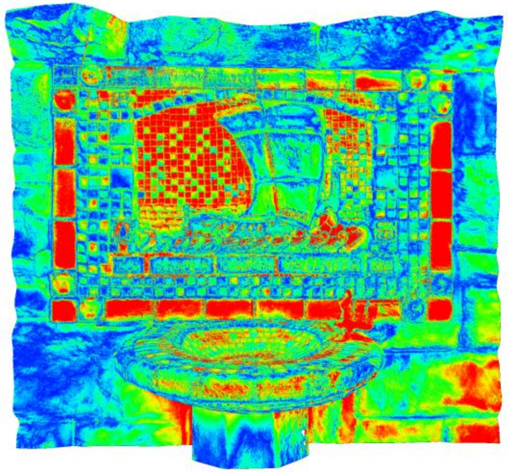
Albedo

Shading: Global SH vs. SVSH

Fountain

Luminance

Shading



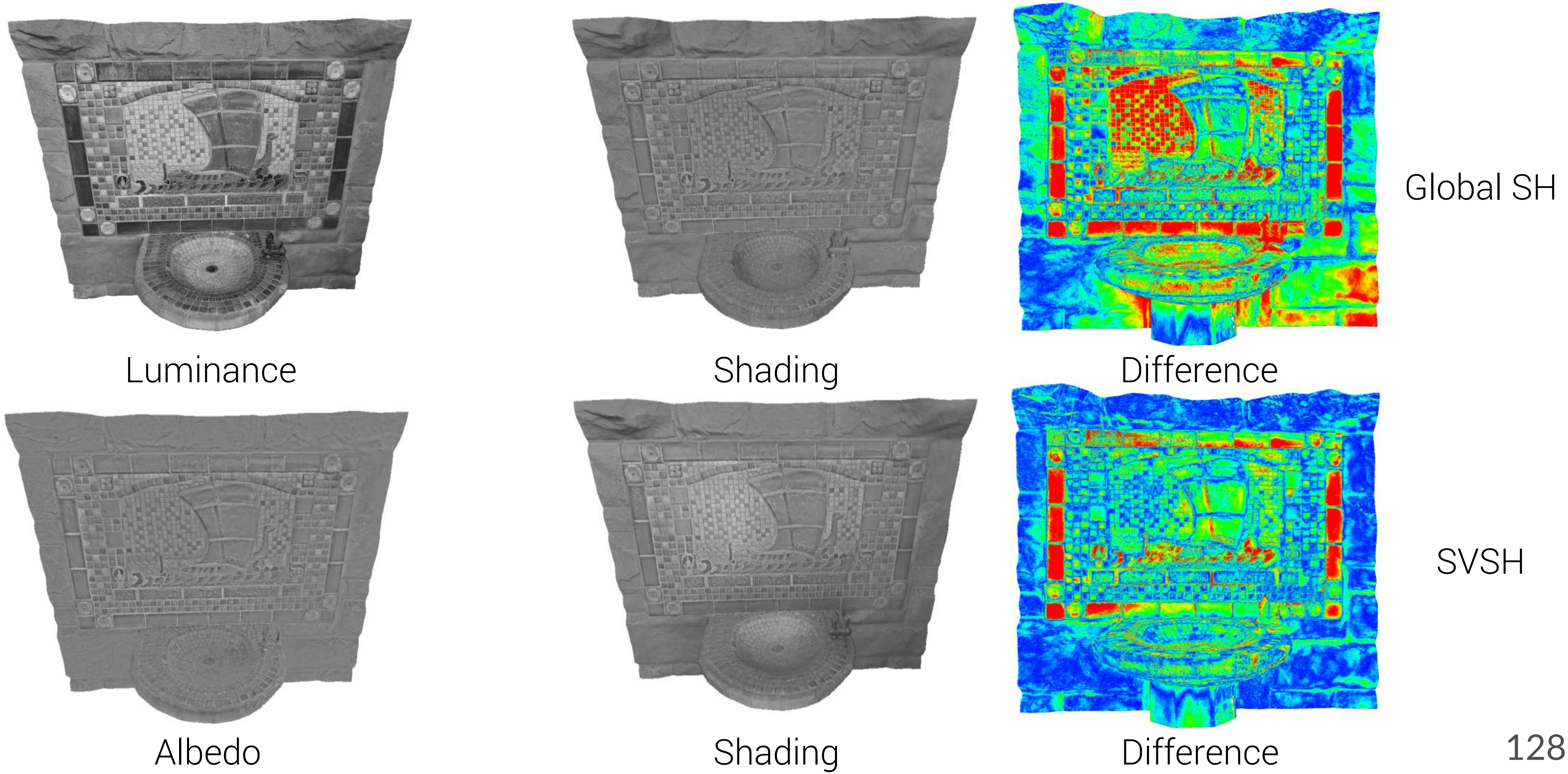
Difference

Global SH

Albedo

Shading: Global SH vs. SVSH

Fountain



Conclusion

Conclusion

High-Quality 3D Reconstruction of Geometry and Appearance

- Temporal view sampling & filtering techniques
- Spatially-Varying Lighting estimation
- Joint optimization of surface & albedo (SDF) and image formation model
- Optimized texture as by-product

Student Projects

- We offer: Master’s Thesis, IDP, Guided Research
- Topics: (RGB-D based) 3D Reconstruction, SLAM, Visual Odometry, Shape from Shading, Photometric Stereo, …
- Interested? Please contact:
  - Yvain Queau (yvain.queau@in.tum.de)
  - Robert Maier (robert.maier@in.tum.de)