Supplementary Material for On Joint Estimation of Pose, Geometry and svBRDF from a Handheld Scanner

Carolin Schmitt[1,2,*] Simon Donné[1,2,*] Gernot Riegler[3] Vladlen Koltun[3] Andreas Geiger[1,2]
[1]Max Planck Institute for Intelligent Systems, Tübingen
[2]University of Tübingen [3]Intel Intelligent Systems Lab
{firstname.lastname}@tue.mpg.de {firstname.lastname}@intel.com

Abstract

*In this **supplementary document**, we first provide additional information on our custom-built sensor. Subsequently, we discuss details of the implementation omitted from the main document due to space constraints. Finally we present quantitative and qualitative results, both photometric and geometric, for all nine objects studied in the main paper. The **supplementary video** gives a high-level overview of our method and provides additional relighting results.*

1. Hardware and capture

Our custom-built handheld capture rig is powered by a compact battery pack such that it is easily portable. It comprises two high resolution cameras, 12 high-powered LEDs, and a custom-built laser with a diffractive optical element (DOE).

The cameras have a native resolution of 4112×3008. The RGB camera uses a standard Bayer pattern (BGGR), while the IR camera is a single-channel camera with a high-pass filter that removes the visible spectrum. The cameras are hardware-synchronized with each other and with the LEDs by an Arduino board. Empirically, we found an exposure time of 15ms to be a reasonable compromise between motion blur and image noise.

The laser emits light of 830nm at up to 650mW through a DOE that diffracts it into ≈ 200000 separate beams, in a pseudorandom pattern. Its projected dot pattern is subsequently observed by the IR camera; to keep the pattern and the wavelength constant, the laser module is actively cooled. We perform OpenCV block matching on the observed pattern to estimate a depth map for each of the observations. Those are then volumetrically fused into a consistent mesh using TSDF fusion [5]. We render the depth map $\mathcal{Z}$ from this mesh into the center view and use this depth as the initial depth estimate.

The whole sensor rig is calibrated geometrically and photometrically in advance. For the geometric calibration, we determine the camera intrisics and extrinsics as well as the positions and orientations of cameras, lights and laser relative to each other. Photometrically, we calibrate the camera vignetting as well as the radiant intensities and attenuation curve per color channel of the light sources. In our method, we assume that debayering, undistorting, and devignetting of the input images is done in pre-processing.

2. Method

2.1. Geometric Consistency

To enforce consistency between depth $\{z_p\}$ and normals $\{n_p\}$, we maximize the inner product between the normal estimate and the normal implied by the cross product of the surface tangents. As the derivations for $\frac{\partial z_p}{\partial x}$ and $\frac{\partial z_p}{\partial y}$ are similar, we

Photometric Test Error	Overall	Specular	Non-Specular
GD with fixed stepsize	1.159	**2.934**	1.110
L-BFGS	2.296	4.432	2.235
Adam, default parameters	1.148	3.186	1.092
Adam, our parameters	**1.131**	3.039	**1.078**

Figure 1: **Different Optimizers.** Please refer to Section 2.2 for a detailed discussion.

show only the derivation of the horizontal tangent:

$$\begin{aligned}\frac{\partial z_p}{\partial x} &\propto \left[1, 0, \frac{\partial \mathcal{Z}_1(\pi_1(\vec{x}_p))}{\partial x}\right]^T \\ &= \left[1, 0, \frac{\partial \mathcal{Z}_1(\pi_1(\vec{x}_p))}{\partial [u,v]^T}\frac{\partial [u,v]^T}{\partial x}\right]^T \\ &= \left[1, 0, \vec{\nabla}\mathcal{Z}_1(\pi_1(\vec{x}_p))^T\,[f/z_p, 0]^T\right]^T\end{aligned} \tag{1}$$

2.2. Implementation

Due to memory constraints, we represent the inferred quantities $\mathcal{X}$ at half-resolution while utilizing the observations at full resolution. We implemented the rendering function $\mathcal{R}_i$ using PyTorch [3], exploiting PyTorch's GPU acceleration and auto-differentiation capabilities. We jointly optimize over the parameters $\mathcal{X}$ using ADAM [2] with hyperparameters $\beta_1 = 0.9$, $\beta_2 = 0.9$ and $\epsilon = 10^{-3}$. We found these settings to be a good fit for our problem. In Table 1 we quantitatively compare four different optimizer choices:

1. Gradient descent with a fixed stepsize for each parameter that we optimize over,
2. L-BFGS,
3. ADAM with the default hyperparameters $\beta_1 = 0.9$, $\beta_2 = 0.999$ and $\epsilon = 10^{-8}$, and
4. ADAM with our tuned hyperparameters $\beta_1 = 0.9$, $\beta_2 = 0.9$ and $\epsilon = 10^{-3}$.

Gradient descent struggles with the balance between bright and dark pixels. It optimizes and reconstructs bright pixels well, but the signal-to-noise ratio in dark pixels degrades performance in those areas drastically. L-BFGS is a second-order optimization algorithm that relies on a series of line searches. We found that this fails to optimize our objective function properly. We assume this is due to the strong inter-dependencies between the different variables, where the relatively low number of line searches that L-BFGS is able to perform in the same time budget as the other optimizers is simply insufficient. The default parameters for ADAM result in the well-known problem of strong perturbations around local minima, and for this reason we increased ϵ to 10^{-3}, mitigating this issue. Finally, given the relatively low number of iterations (1000), the shorter memory $\beta_2 = 0.9$ helps ADAM to adapt more quickly.

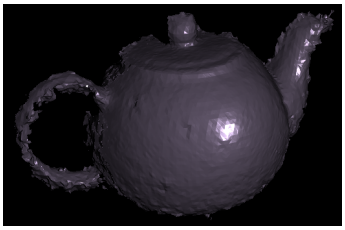
We choose the step size multipliers for each of the parameters in a physically-motivated way (because of the way ADAM works, this is a soft upper bound on the actual stepsize). In our experiments we empirically found the following values to perform well:

- Depth: 0.1mm
- Normals: 0.5 degrees
- Pose: 0.1mm and 0.5 degrees, respectively
- Material properties: 1% of the domain range

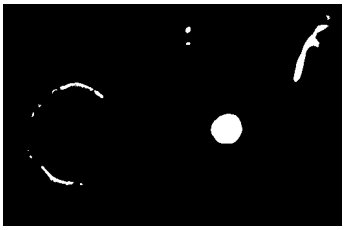
3. Additional Experimental Evaluations

We now first discuss the specular/non-specular loss function split used in the main paper. Secondly, we also provide a quantitative comparison showing the effect of the depth initialization on the final result. Lastly, we present and discuss both quantitative and qualitative results for all nine objects studied in the main paper, first geometrically and then photometrically.

Observation

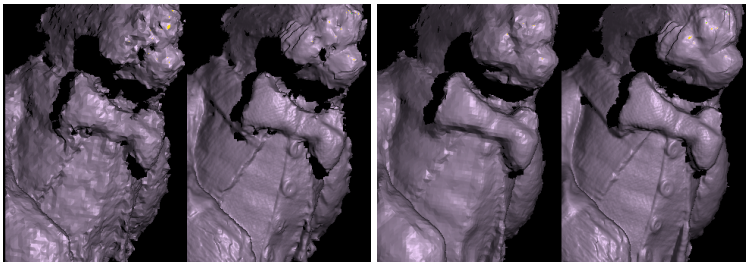


Initial Geometry (same lighting)



Specularity Map

Figure 2: **Specularity Mask.** We smooth the initial depth map and calculate the angle between normal $\{n_p\}$ and half-vector for every pixel. We determine the in-specular and non-specular image regions by thresholding the resulting angles.



Initialized from 5 depth maps

Initialized from 40 depth maps

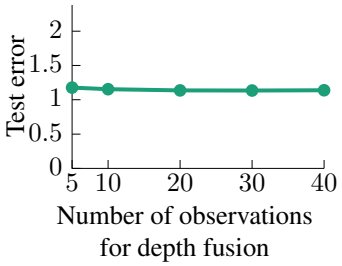


Figure 3: **Geometry Refinement.** The reconstruction quality is barely affected by the input depth estimate. The test photometric error only degrades slightly when using the worst depth initialization and details are recovered, even from a very coarse initialization.

Evaluation Metrics: We evaluate the performance of our method photometrically by calculating the mean L_1 loss between the observation and our prediction for every pixel and all input images. Since most of the BRDF information is captured in pixels that potentially observe specular highlights, we split the mean photometric test error into pixels in possibly-specular and non-specular regions using a mask as the one illustrated in Fig. 2.

To calculate this *specularity mask* for each observation, we threshold the angle between the normalized halfvector ($\propto \boldsymbol{\omega}_p^{\text{in}} + \boldsymbol{\omega}_p^{\text{out}}$) and the initial normal for each pixel and each observation. For this, the initial normals are calculated from the input depth and smoothed. The angle threshold is is set to $15°$. We empirically found that the resulting regions cover the majority of pixels in specular highlight areas.

3.1. Impact of Geometry Initialization

To calculate the initial depth map we fuse the input depth images from all observations into a single mesh. From that mesh we then render the initial depth map for the reference view. In the main paper, we already illustrated qualitatively that we are able to recover geometric detail, even in the presence of inaccurate depth initialization. Fig. 3 supports these observations quantitatively: the test photometric error is almost constant over the number of input depth images.

3.2. Geometric Results

We now show qualitative geometrical results for all objects in our test set for two reference views in Fig. 4 - 7. In addition, we also show the quantitative results reported in the main paper, separately per reference view. For these comparisons, we now also include the COLMAP [4] MVS baseline. This approach assumes, with some robustness, appearance constancy – this assumption is very much violated in our set-up. This becomes obvious for objects like the duck which has large textureless areas for which appearance is primarily dominated by shading effects. For more textured objects, such as the rabbit or the girl, COLMAP produces reasonable results.

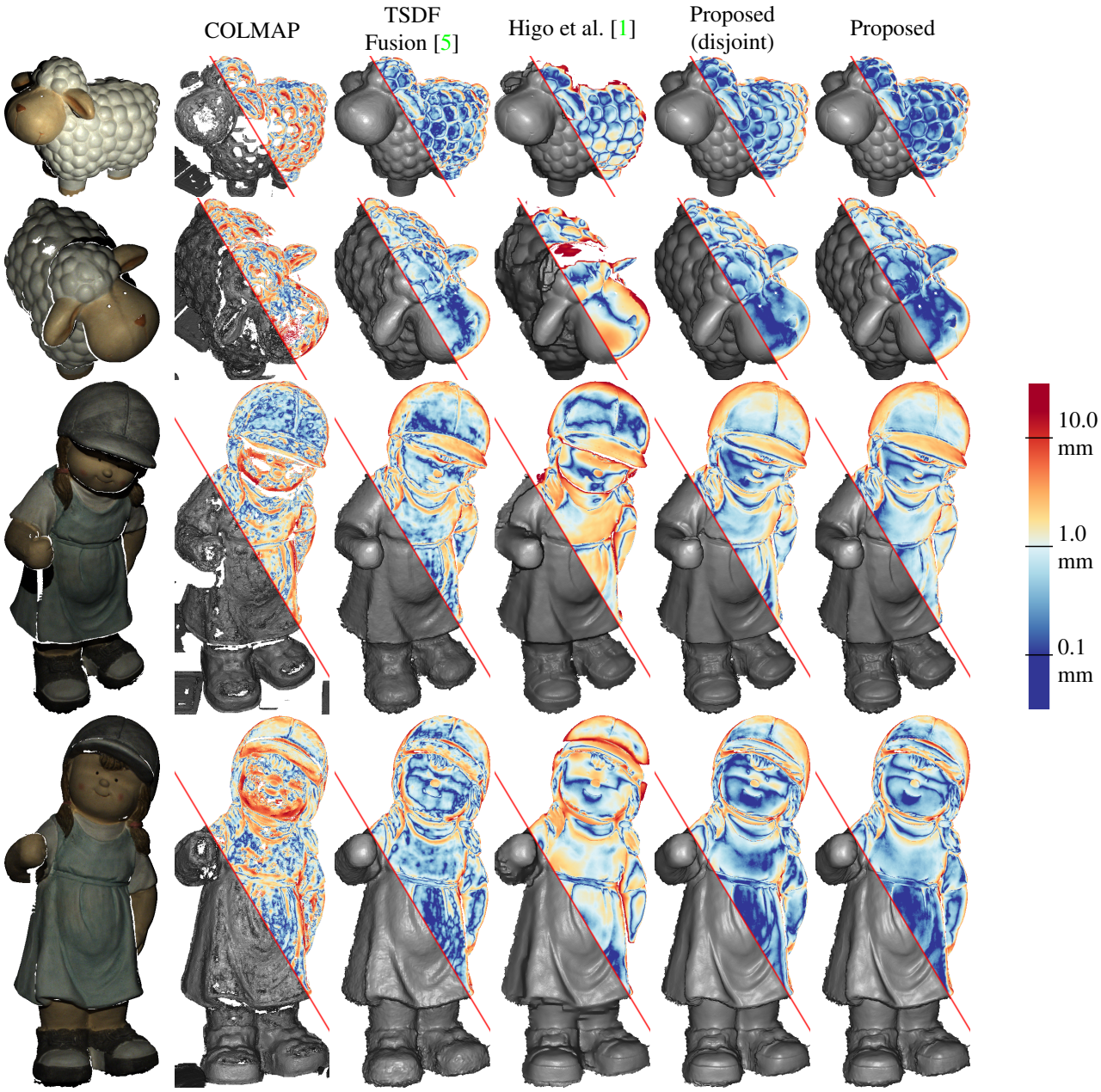


Figure 4: **Qualitative Geometry Comparison.** For both views of the girl and the sheep statues. We show, for each object, and each reference view, the rendered depth map (shaded based on the estimated surface normals) and the color-coded depth error wrt. the Artec Spider ground truth.

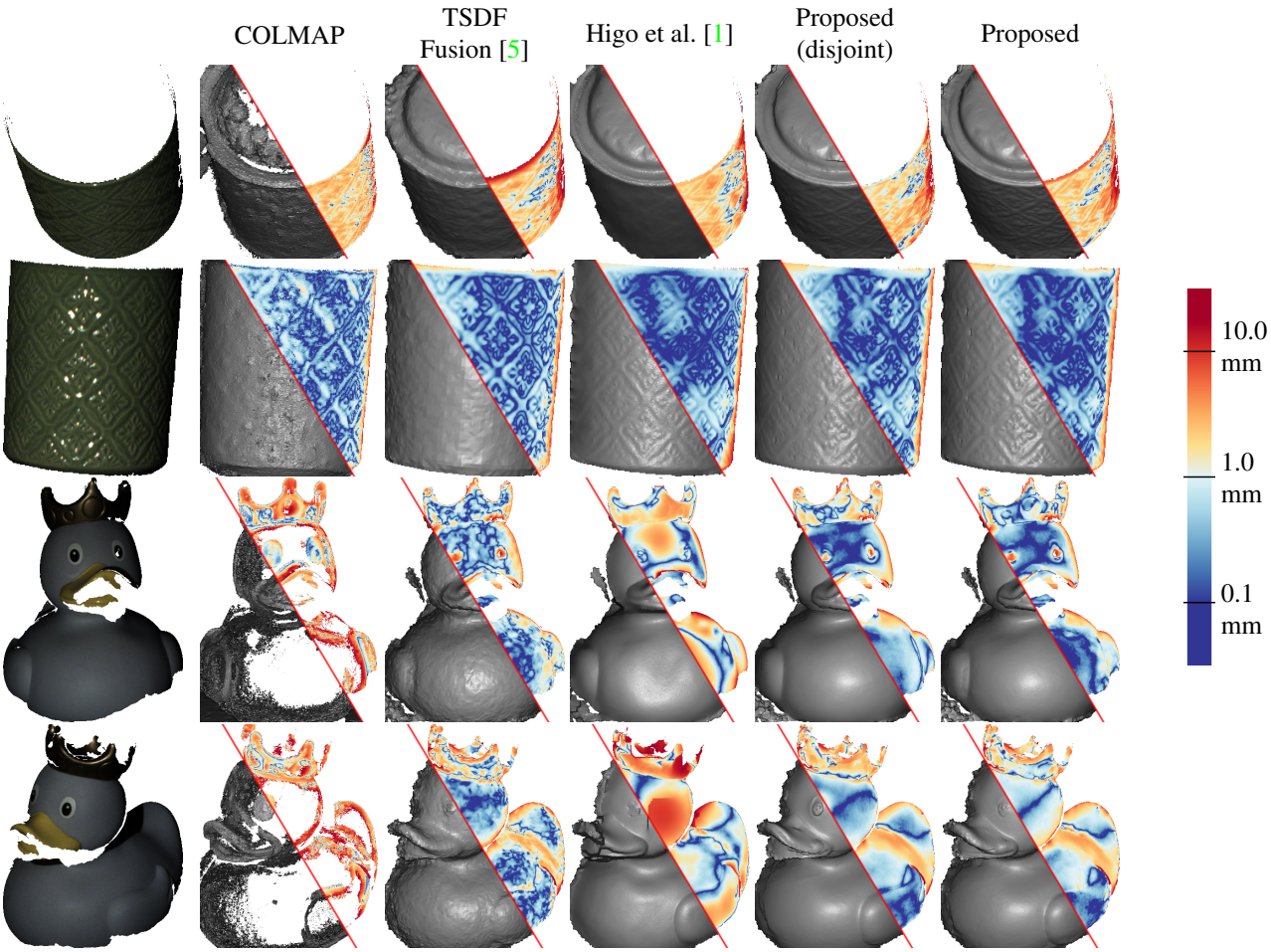


Figure 5: **Qualitative Geometry Comparison.** For both views of the vase and the duck. We show, for each object, and each reference view, the rendered depth map (shaded based on the estimated surface normals) and the color-coded depth error wrt. the Artec Spider ground truth.

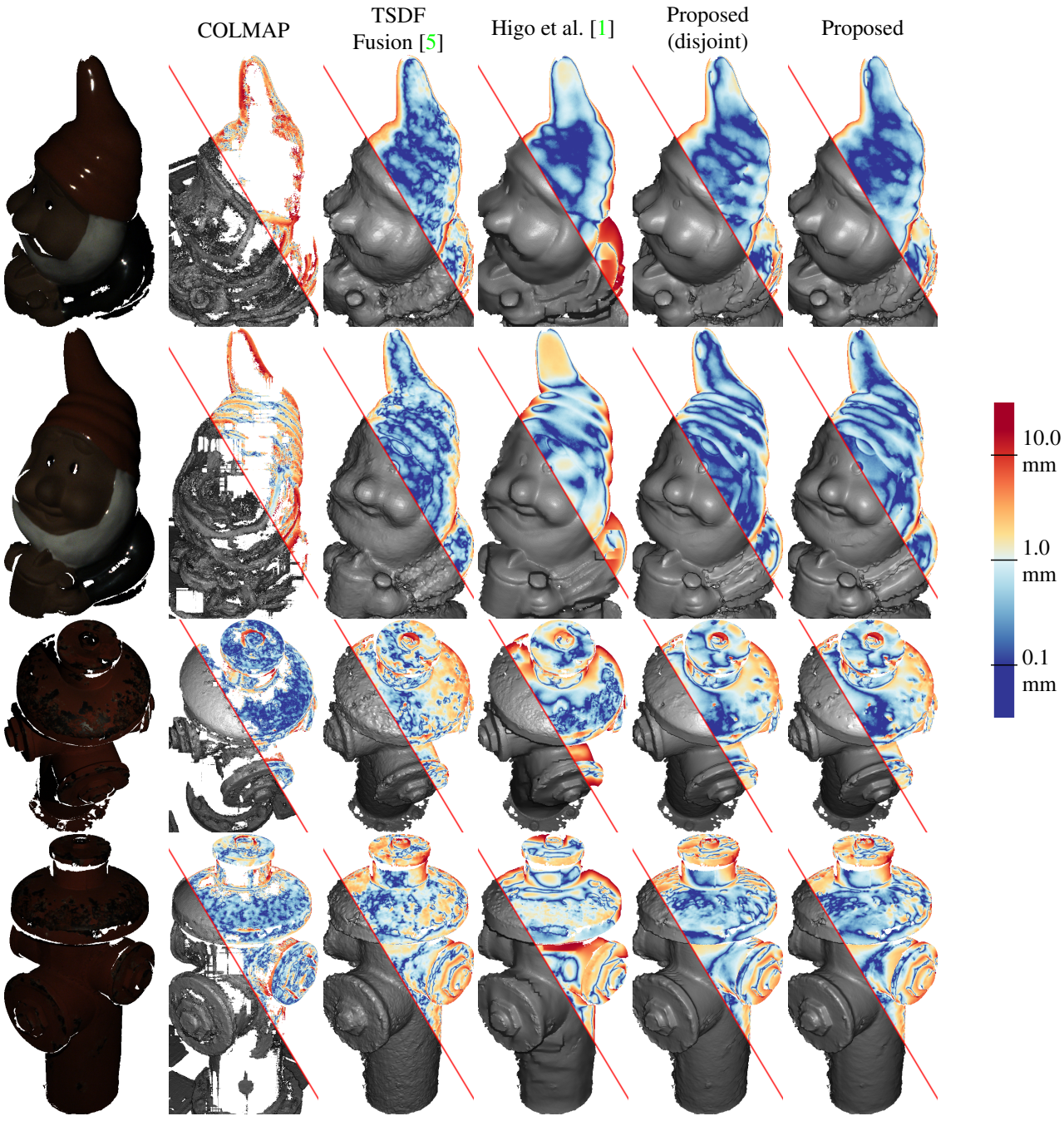


Figure 6: **Qualitative Geometry Comparison.** For both views of the gnome and the hydrant statues. We show, for each object, and each reference view, the rendered depth map (shaded based on the estimated surface normals) and the color-coded depth error wrt. the Artec Spider ground truth.

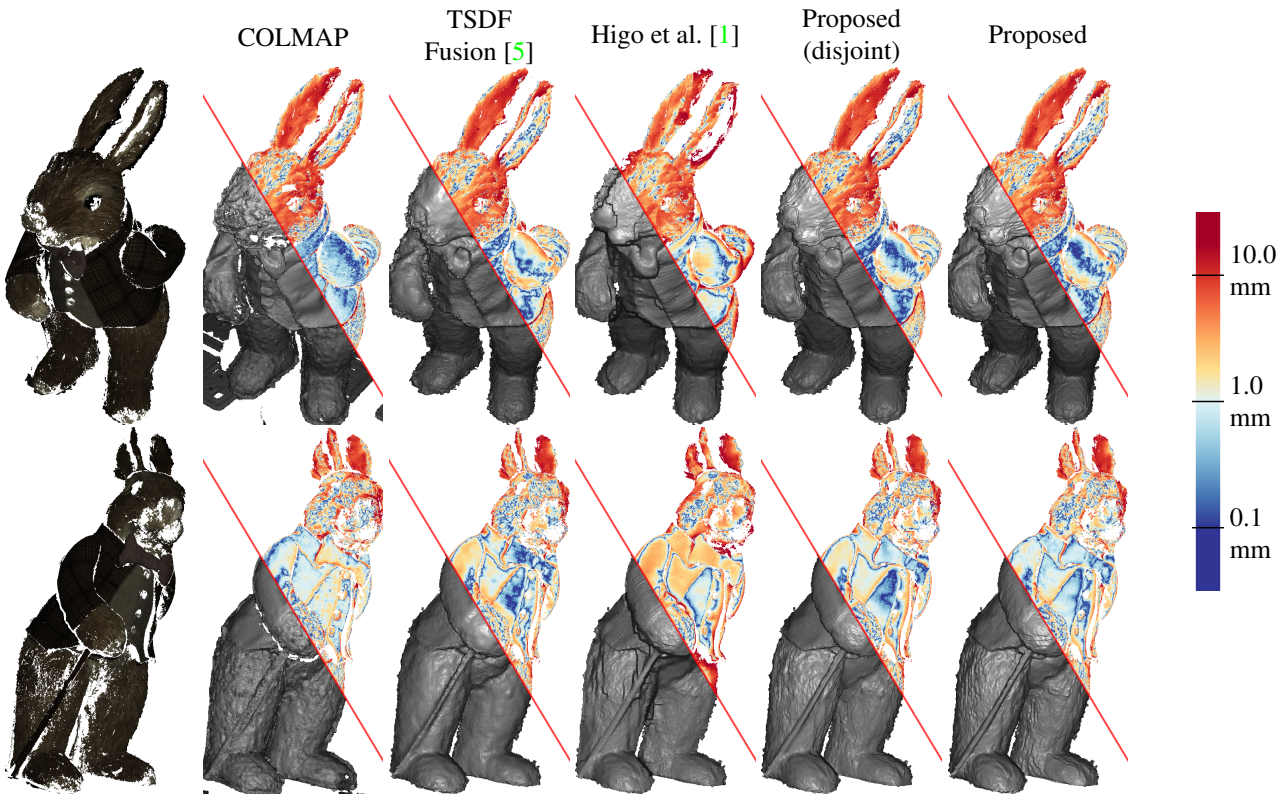


Figure 7: **Qualitative Geometry Comparison.** For both views of the rabbit statue. We show, for each object, and each reference view, the rendered depth map (shaded based on the estimated surface normals) and the color-coded depth error wrt. the Artec Spider ground truth.

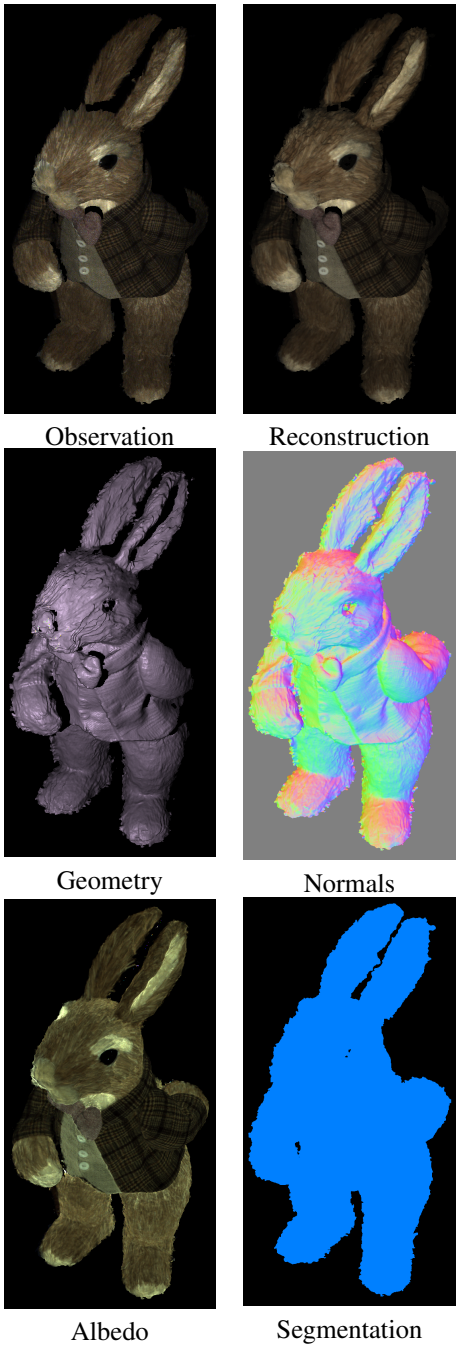
		duck		pineapple		girl		gnome		sheep		hydrant		rabbit	
Metric	Method	View 1	View 2	View 1	View 2	View 1	View 2	View 1	View 2	View 1	View 2	View 1	View 2	View 1	View 2
AEA	COLMAP refined	4.96	4.80	1.78	0.44	1.62	1.62	4.01	2.84	3.11	2.12	**0.93**	1.14	**2.42**	1.94
	TDSF fusion (40 frames)	0.71	**0.91**	1.96	0.52	1.28	0.95	0.84	0.63	0.97	0.60	1.49	1.22	2.67	**1.66**
	Higo COLMAP poses	2.65	2.65	1.65	0.44	1.61	1.57	1.81	1.39	2.98	1.20	1.88	1.74	3.01	2.69
	Disjoint	0.67	0.96	**1.54**	0.47	1.23	0.88	0.75	0.56	0.74	0.54	1.25	1.12	2.75	1.75
	Proposed	**0.60**	0.99	1.59	**0.41**	**1.20**	**0.80**	**0.73**	**0.55**	**0.71**	**0.43**	1.24	**1.08**	2.63	1.68
AAE	COLMAP refined	39.56	56.19	12.93	12.72	24.46	23.27	42.78	38.85	38.80	30.97	20.31	18.95	27.48	24.64
	TDSF fusion (40 frames)	6.41	7.09	14.50	9.67	11.61	11.19	7.84	7.44	9.29	7.47	12.90	10.44	25.90	22.94
	Higo COLMAP poses	7.79	7.75	11.54	9.69	10.76	11.84	9.22	9.26	10.05	7.26	14.66	15.87	29.59	25.75
	Disjoint	6.24	5.77	**9.95**	8.31	10.13	9.49	6.54	6.29	6.89	6.42	9.26	9.92	**24.77**	**21.25**
	Proposed	**5.50**	**4.85**	10.53	**7.42**	**9.03**	**8.43**	**6.15**	**5.34**	**6.15**	**5.05**	**9.02**	**7.99**	25.15	21.86

Figure 8: **Quantitative Geometry Comparison.** We report both the average Euclidean accuracy and average angular error.

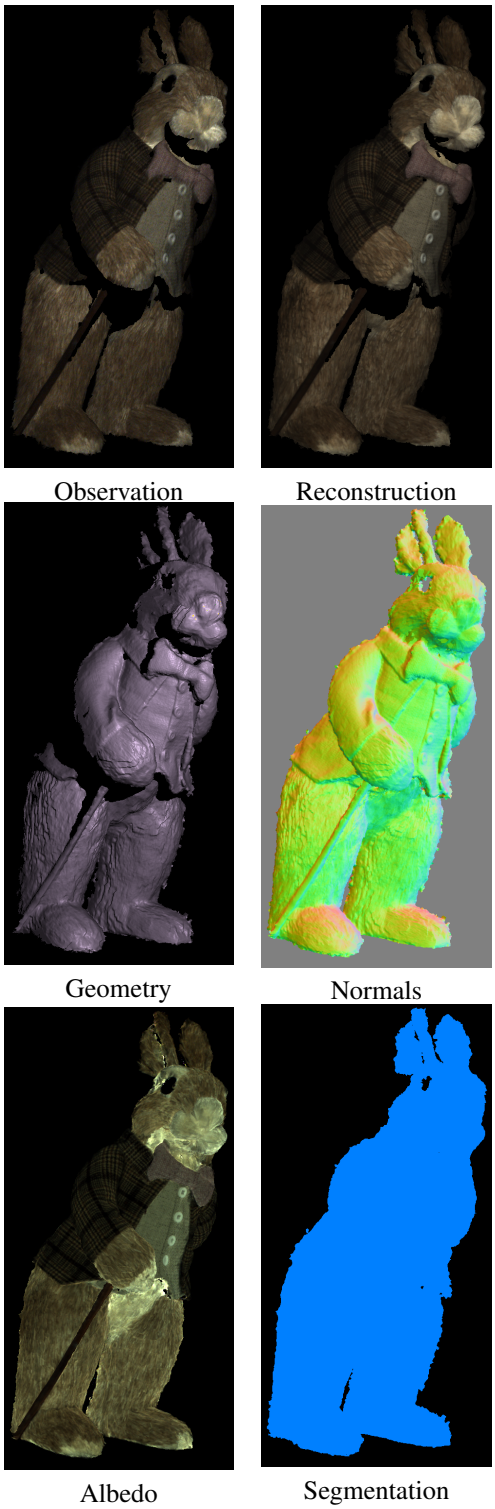
3.3. Photometric Results

On the remaining pages we provide the reconstructions of all nine objects shown in the main paper. Per object, we chose 2 reference views. In each figure, the first row shows one randomly chosen observation plus the corresponding reconstruction by the proposed method. The following two rows contain the predicted geometry, normal and albedo maps as well as the specular material weight map. The normals used for rendering the geometry map are calculated from the predicted depth map.

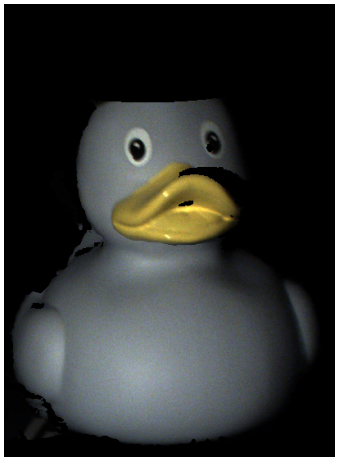
Rabbit: The jacket of the rabbit is made of a rough woolen fabric and its structure is well captured in the geometry and normal maps. The fur of the rabbit is made of 1-2mm thick straws, which corresponds to approximately 2 pixels in the observation images (the rabbit is one of our largest statues). This makes them hard to represent at this input resolution.



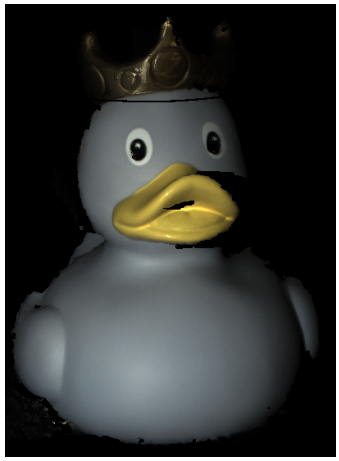
Rabbit: Note the fine details recovered in the geometry (the buttons or the structure of the bow tie). The eyes are missing in the initial depth map so they can not get reconstructed by the proposed method.



Duck: The crown is highly specular and hard to reconstruct correctly. Due to our hand-held capture rig, the objects are not entirely visible in all observations. In this view, the crown is outside of the image boundaries. The proposed method cleanly separates the very specular black pupil and the non-specular white of the eye as visible in the segmentation image.



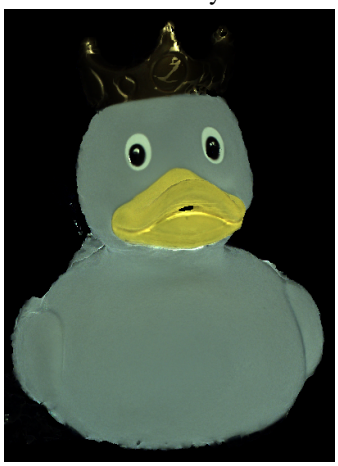
Observation



Reconstruction

Geometry

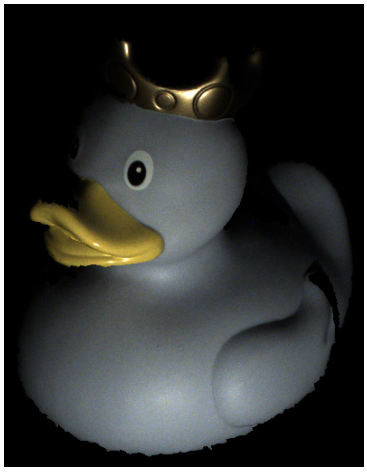
Normals



Albedo

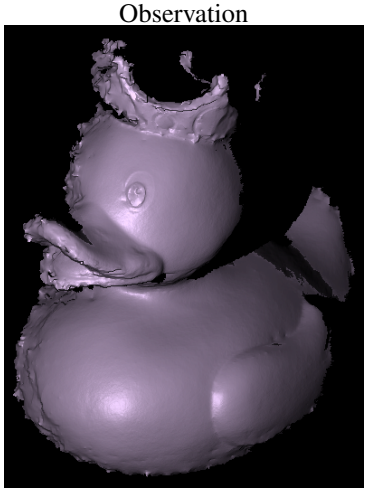
Segmentation

Duck: Especially the left side of the duck is not well lit in the observations used for this reference view. That leads to very limited guidance for the optimization and noticeable artifacts at the geometry boundaries. In the 40 observations of this reference view, the pixels of the eyes do not see any specular highlights and can not get reconstructed as cleanly as in the previous reference view.

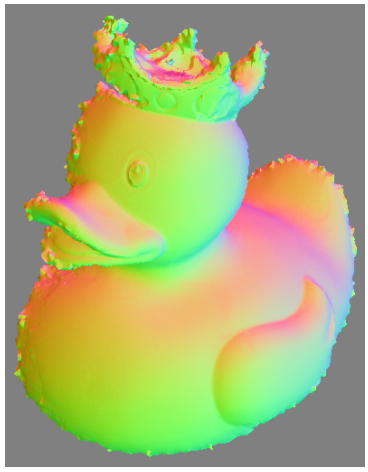


Observation

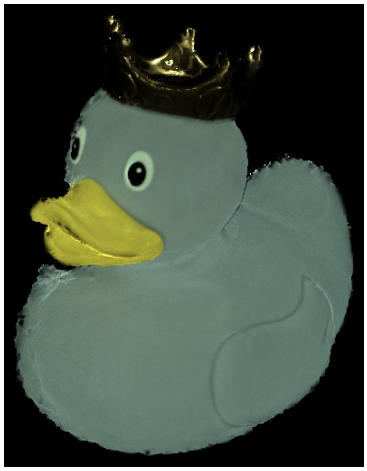
Reconstruction



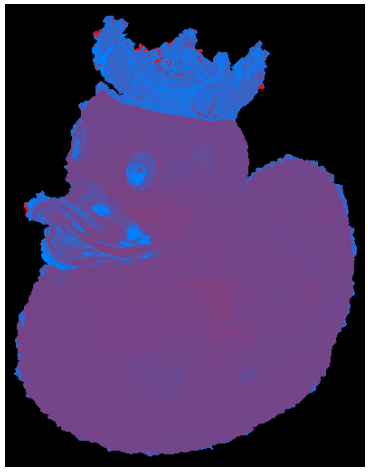
Geometry



Normals

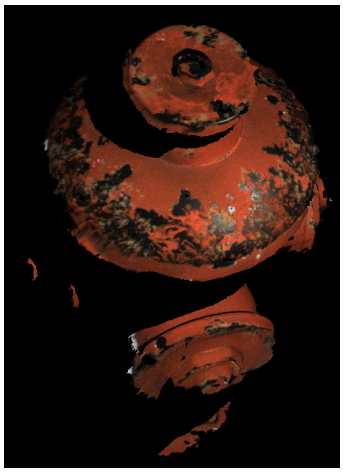


Albedo

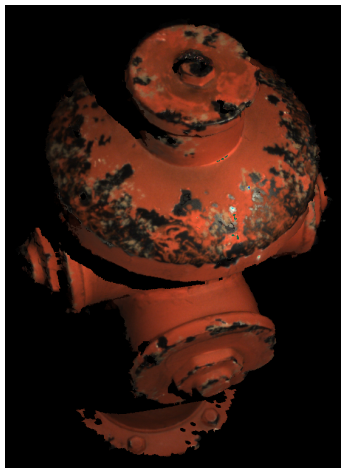


Segmentation

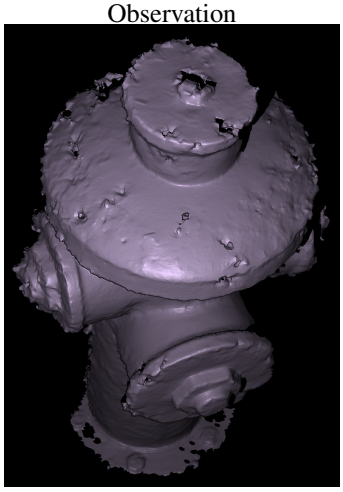
Hydrant: The hydrant primarily consists of two different materials, the diffuse red color and the specular dark patches that mimic rust. Both materials are well captured by the segmentation.



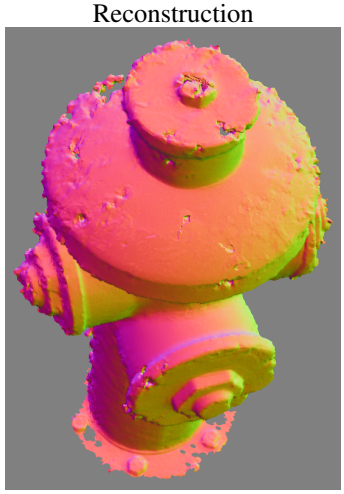
Observation



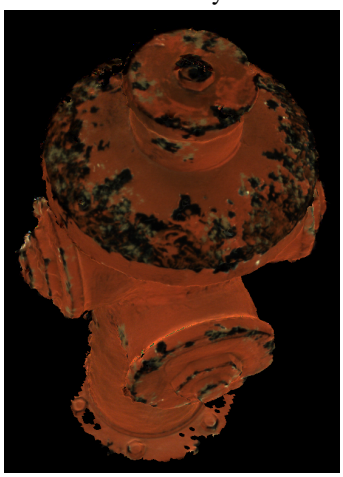
Reconstruction



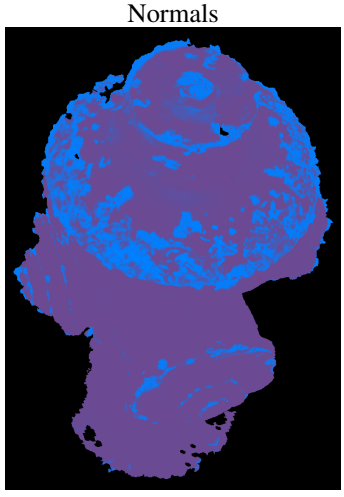
Geometry



Normals

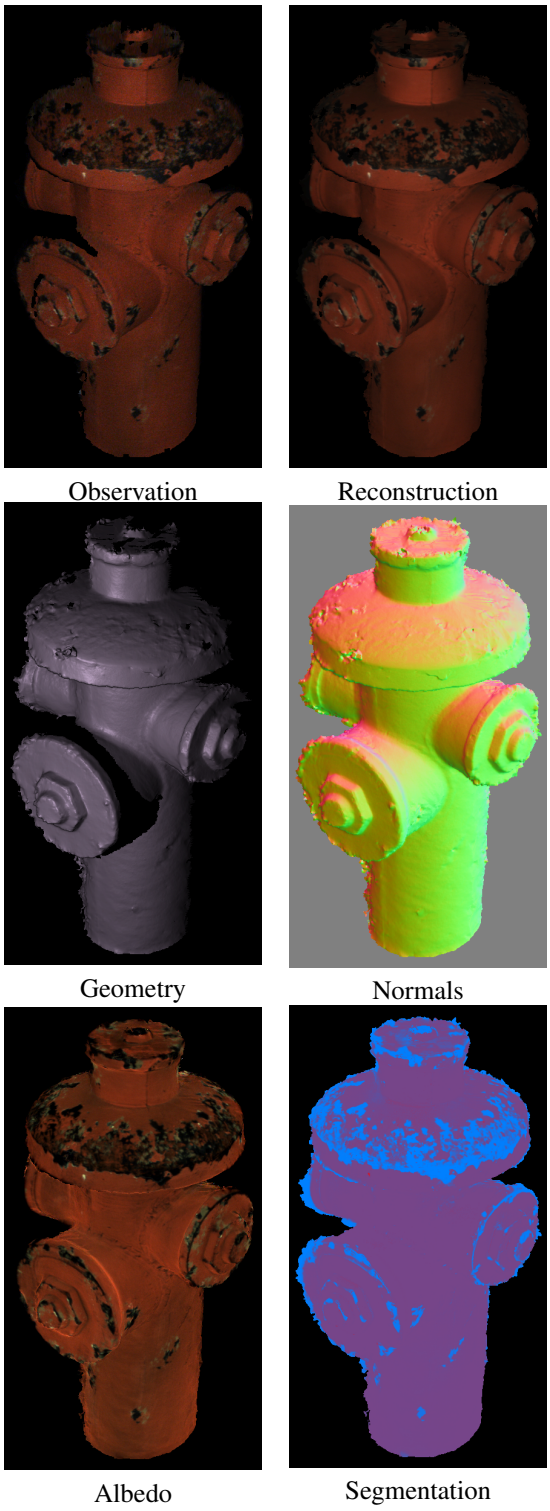


Albedo



Segmentation

Hydrant: In terms of geometry, we reconstruct both the smooth cylindrical surface and the sharp bolts accurately.

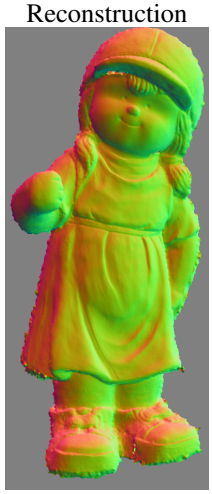


Girl: The girl shows a very uniform specular appearance, and accordingly the proposed method prefers a single specular base. Contrary to previous objects, there is a smooth transition between colors, for example the mixture of white and blue on the dress. The per pixel albedo estimation manages to recover that accurately.

Observation

Reconstruction

Geometry



Normals

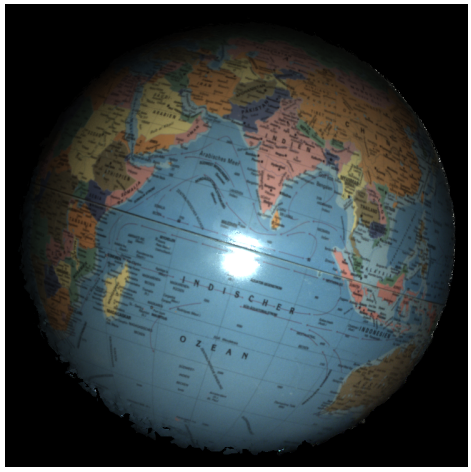
Albedo

Segmentation

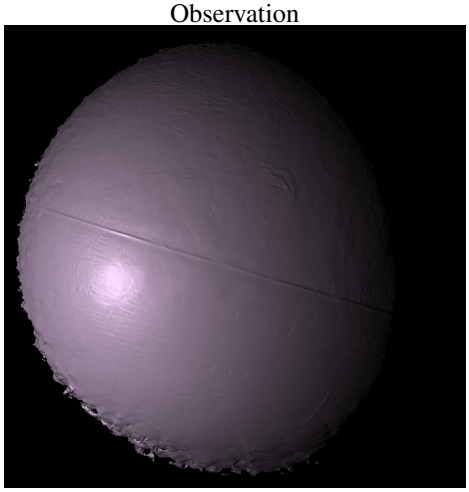
Girl: In terms of geometry, the girl statue shows many fine details like the strands of hair or wrinkles of the dress. Note that even the stitches on the cap are visible in both the estimated geometry and normal maps.

Globe: The globe is made of two half spheres that are connected around the equator. That causes significant changes in the geometry and an irregular specular lobe. Both aspects are recovered by the proposed method. Because of the highly detailed texture, it was much more obvious for this object than for others that optimization over camera poses is crucial – without it, the texture estimate is blurry as correspondences are offset.

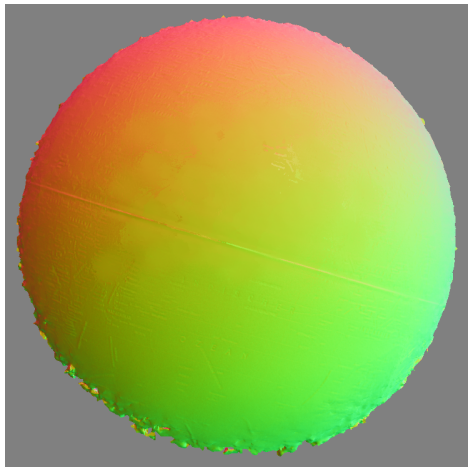
Observation



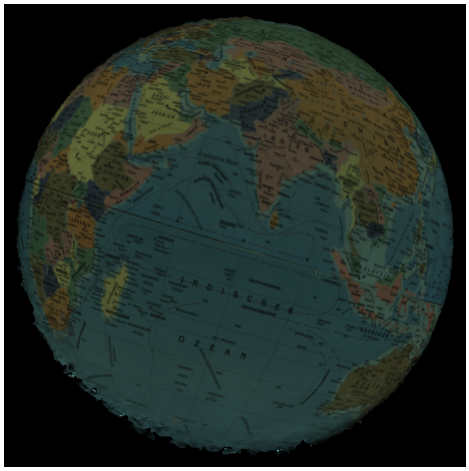
Reconstruction



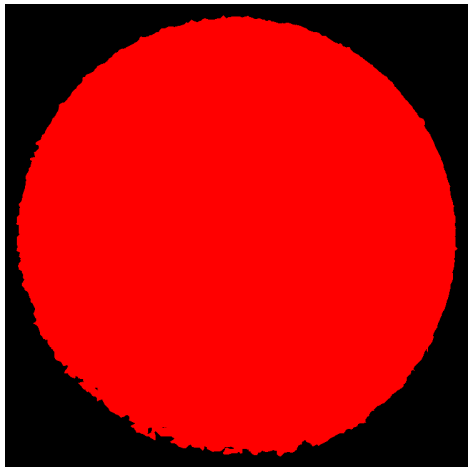
Geometry



Normals

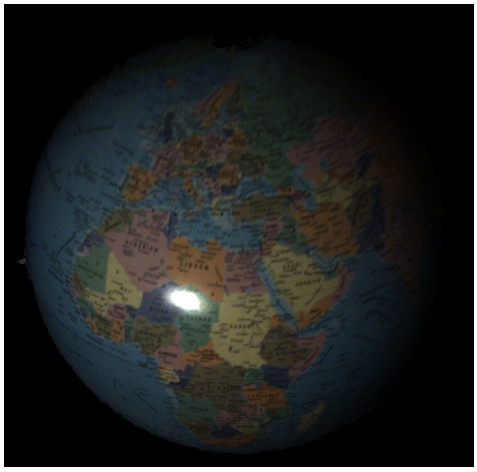


Albedo

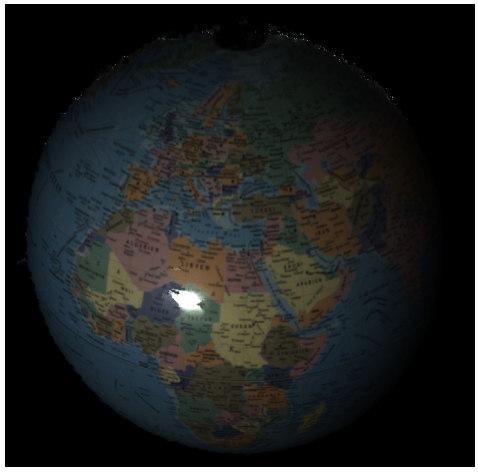


Segmentation

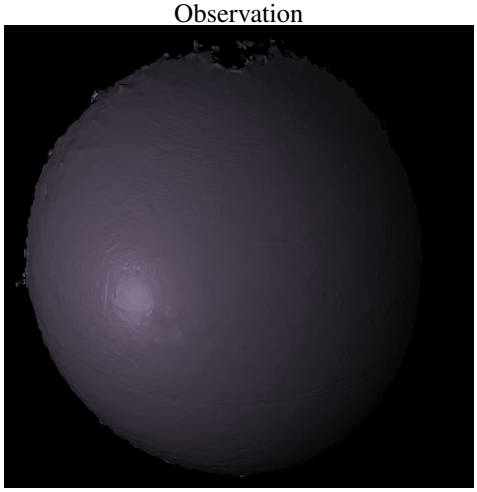
Globe: The globe sports strong texture on a very smooth and shiny surface, making the disentanglement of geometry and appearance very challenging. We see that the texture is well captured in the albedo map, yet some details are also present in the normal map. Even though the proposed approach is generally able to distinguish texture from geometry, there is still some ambiguity left; more input observations would help to disambiguate these further.



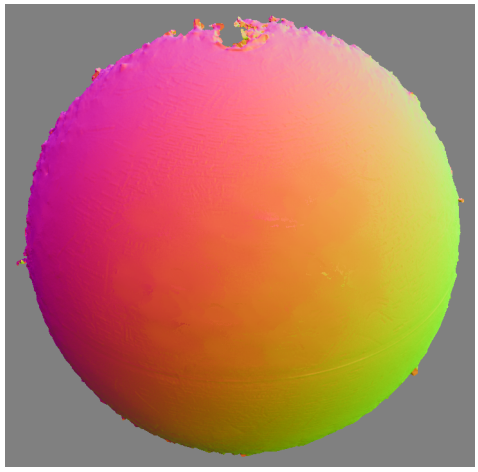
Observation



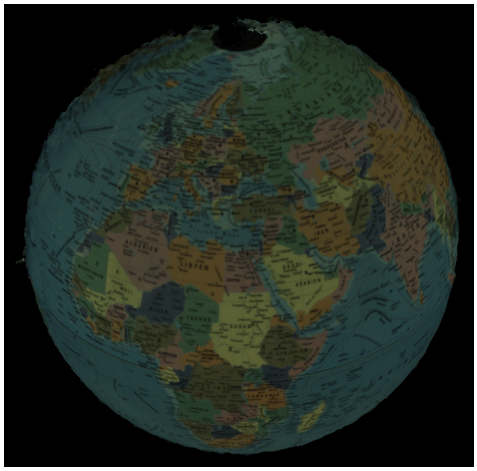
Reconstruction



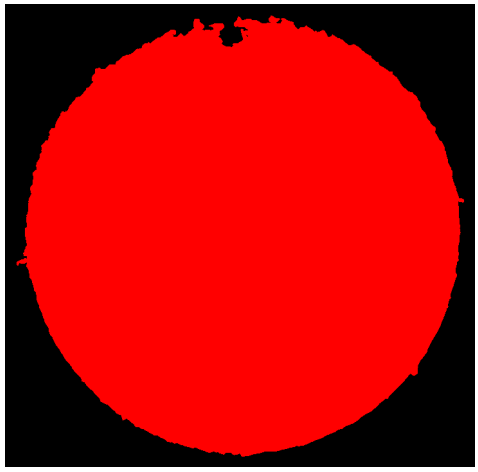
Geometry



Normals

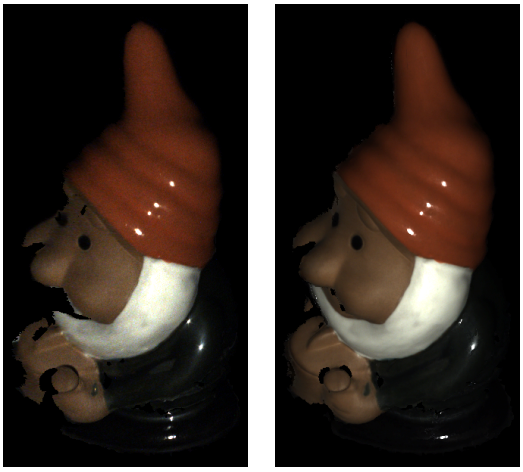


Albedo

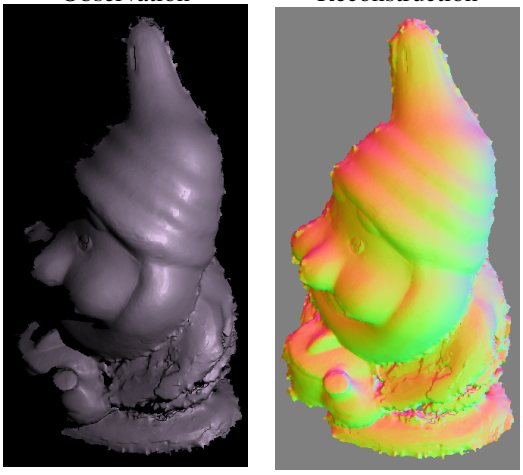


Segmentation

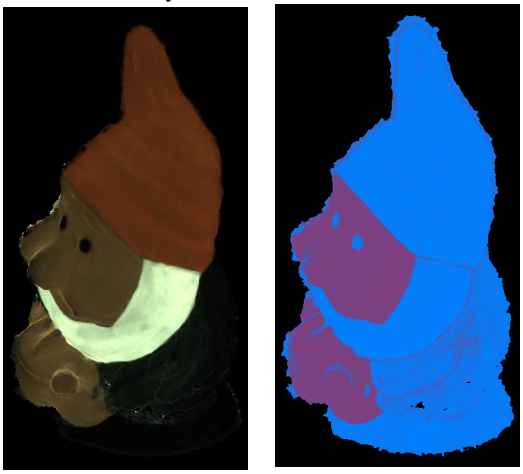
Gnome: The gnome is a very interesting objects since it combines materials with very different specular properties. This is well captured by the segmentation. Noticeable in the geometry and normal maps is the arm which is very specular and almost black. As the color is really dark and highlights are only observed very sparsely, the signal-to-noise ratio is very low. The reconstruction shows that there is not enough information in the data to recover the structure perfectly.



Observation Reconstruction



Geometry Normals

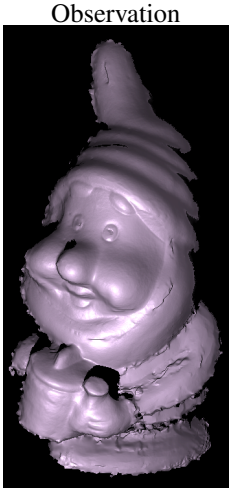


Albedo Segmentation

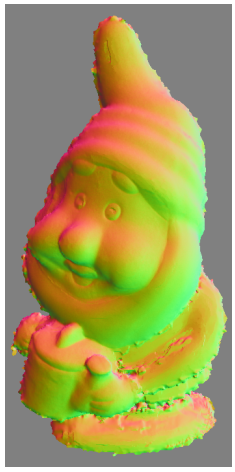
Gnome: For this reference view, we notice that the beard gets incorrectly assigned to the non-specular base material. Apparently, there are not enough photometric cues to assign it to the more specular base material. We expect that more observations, or a larger fraction of them showing specularities, would resolve this issue.

Observation

Reconstruction

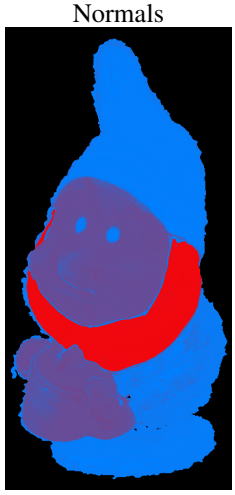


Geometry



Normals

Albedo

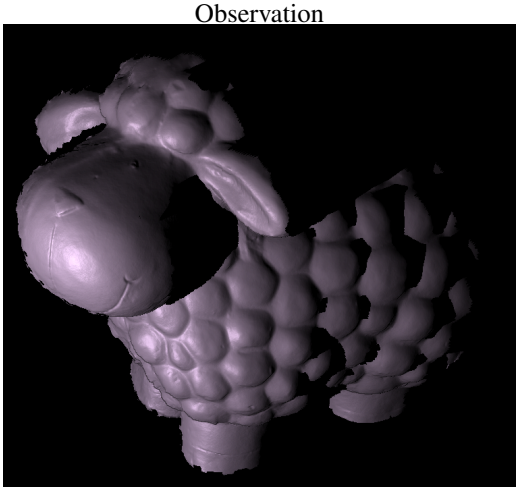


Segmentation

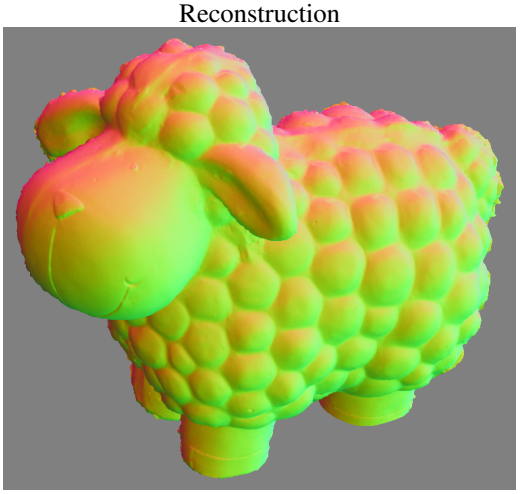
Sheep: The sheep is reconstructed very nicely. Details like the mouth and nose are part of the statue's geometry; these fine structures (on the order of 1mm) get recovered successfully by the proposed method.

Observation

Reconstruction

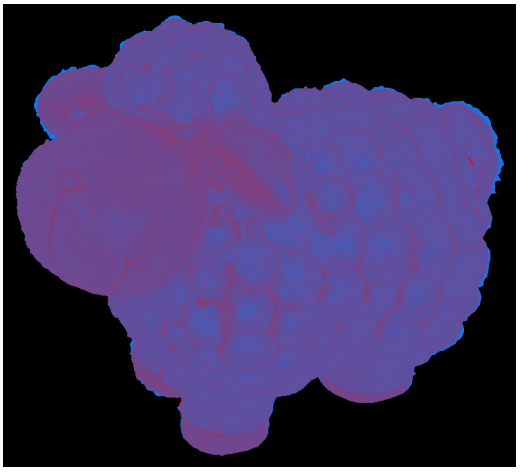


Geometry



Normals

Albedo

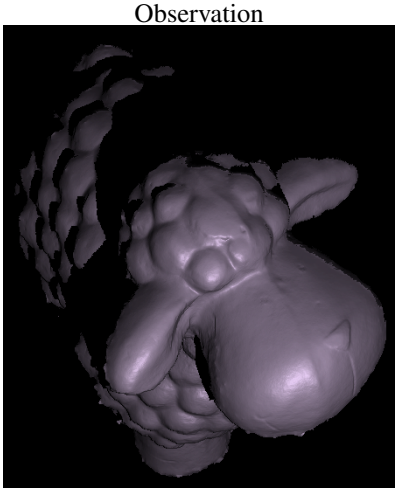


Segmentation

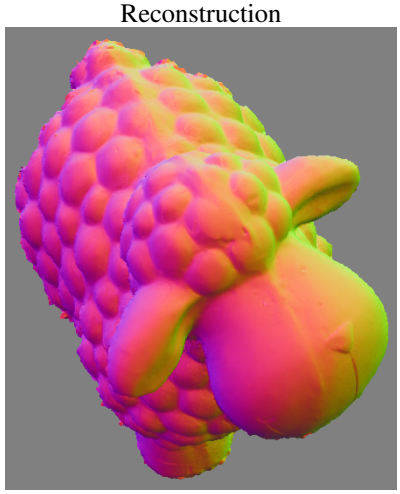
Sheep: The sheep is largely homogeneous in its specular behavior. It does not show very sharp specular highlights but the influence of the specular BRDF component is well visible on the white fur: While the albedo map does not show any distinct white spots, our reconstruction shows them clearly.

Observation

Reconstruction



Geometry



Normals

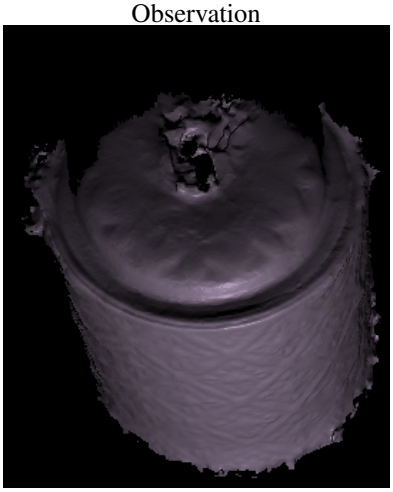
Albedo

Segmentation

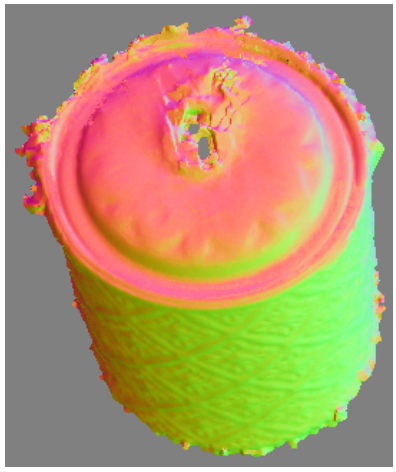
Vase: The vase has a small golden pineapple on top which is very specular, almost mirrorlike. It causes inter-reflections on the lid of the vase which we do not model for two reasons: 1) There is no initial depth from our structured light sensor, and 2) We only simulate a single light bounce in our differentiable rendering step. The specular highlights on the lid that *were* caused by a single bounce are correctly reconstructed.

Observation

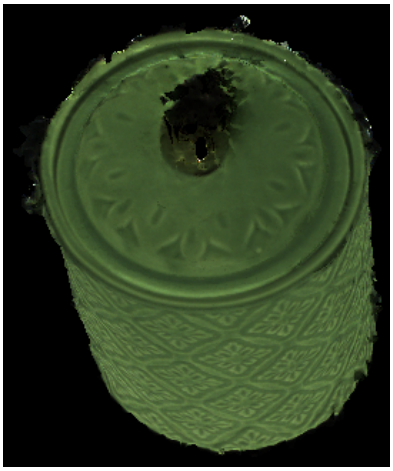
Reconstruction



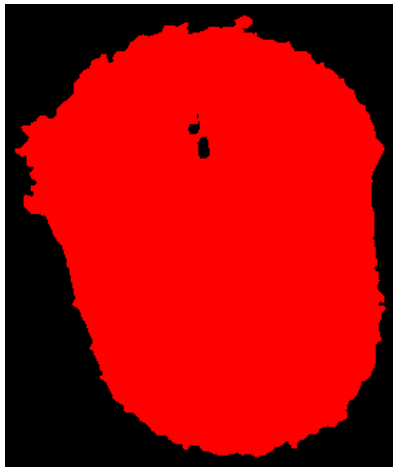
Geometry



Normals



Albedo



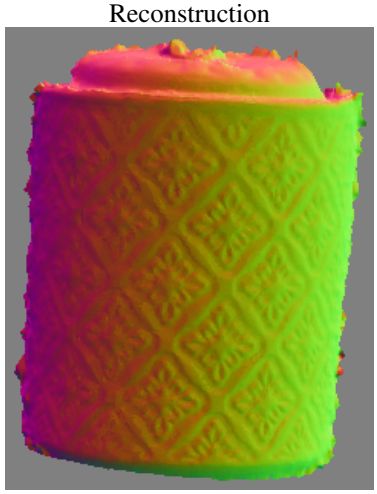
Segmentation

Vase: The detailed reconstruction of the geometry and normals enables the proposed method to recover most of the specular highlights on the body of the vase.

Observation

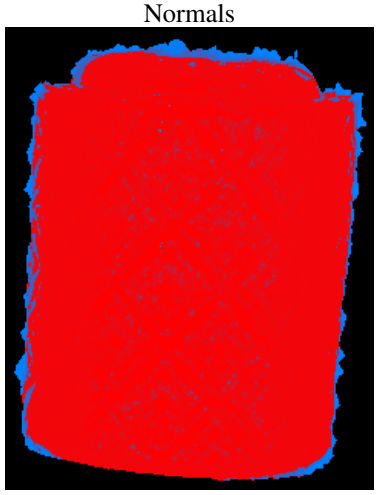
Reconstruction

Geometry



Normals

Albedo

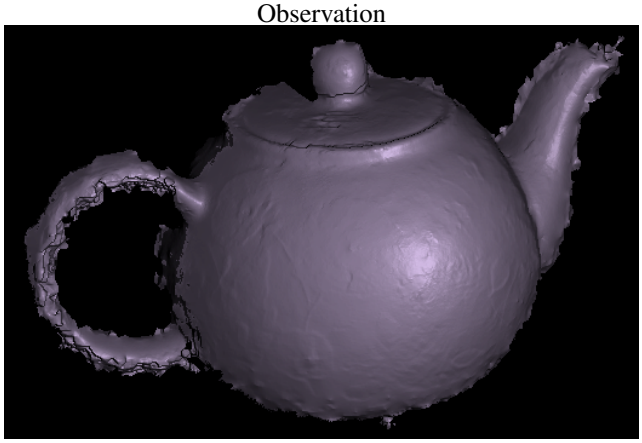


Segmentation

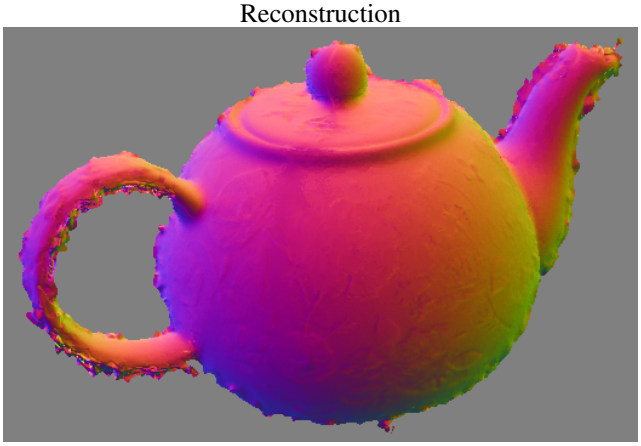
Teapot: The teapot is made of fine bone china with delicate and very thin flowers painted on top of it, which introduces very high-frequency normal and texture behavior. That additional paint layer is visible as very fine structures in both the normal and the albedo map.

Observation

Reconstruction



Geometry

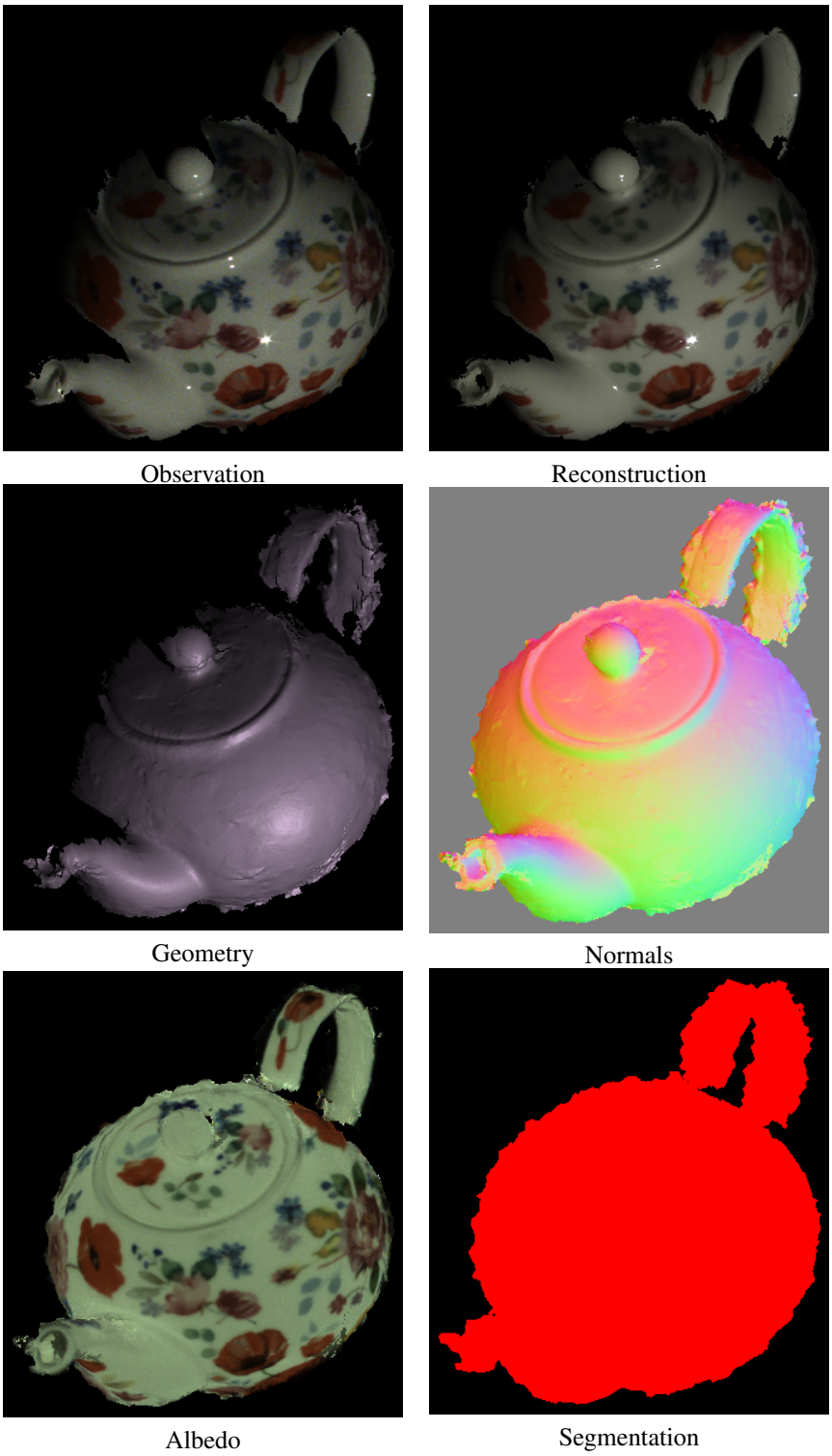


Normals

Albedo

Segmentation

Teapot: A lacquer finish lies on top of the artwork of the teapot. Therefore, a single specular base material is sufficient to model its specular behavior.



References

[1] Tomoaki Higo, Yasuyuki Matsushita, Neel Joshi, and Katsushi Ikeuchi. A hand-held photometric stereo camera for 3-d modeling. In *Proc. of the IEEE International Conf. on Computer Vision (ICCV)*, 2009. 4, 5, 6, 7

[2] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *Proc. of the International Conf. on Learning Representations (ICLR)*, 2015. 2

[3] Adam Paszke, Sam Gross, Soumith Chintala, Gregory Chanan, Edward Yang, Zachary DeVito, Zeming Lin, Alban Desmaison, Luca Antiga, and Adam Lerer. Automatic differentiation in PyTorch. In *Advances in Neural Information Processing Systems (NIPS) Workshops*, 2017. 2

[4] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2016. 3

[5] Andy Zeng, Shuran Song, Matthias Nießner, Matthew Fisher, Jianxiong Xiao, and Thomas Funkhouser. 3dmatch: Learning local geometric descriptors from rgb-d reconstructions. In *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1, 4, 5, 6, 7