Deep 3D Capture: Geometry and Reflectance from Sparse Multi-View Images Supplementary Materials

Sai Bi[1] Zexiang Xu[1] Kalyan Sunkavalli[2] David Kriegman[1] Ravi Ramamoorthi[1]
[1]UC San Diego [2]Adobe Research

1. BRDF Model

We use a simplified version of the Disney BRDF model [1] proposed by Karis et al. [2]. Let A, N, R, S be the diffuse albedo, normal, roughness and specular albedo respectively, L and V be the light and view direction, and $H = \frac{V+L}{2}$ be their half vector. Our BRDF model is defined as:

$$f(A, N, R, L, V) = \frac{A}{\pi} + \frac{D(H, R)F(V, H, S)G(L, V, H, R)}{4(N \cdot L)(N \cdot V)} \tag{1}$$

where $D(H, R)$, $F(V, H, S)$ and $G(L, V, H, R)$ are the *normal distribution*, *fresnel* and *geometric terms* respectively. These terms are defined as follows:

$$\begin{aligned}
D(H, R) &= \frac{\alpha^2}{\pi \left[(N \cdot H)^2(\alpha^2 - 1) + 1\right]^2} \\
\alpha &= R^2 \\
F(V, H, S) &= S + (1 - S)2^{-[5.55473(V \cdot H)+6.8316](V \cdot H)} \\
G(L, V, R) &= G_1(V, N)G_1(L, N) \\
G_1(V, N) &= \frac{N \cdot V}{(N \cdot V)(1 - k) + k} \\
G_1(L, N) &= \frac{N \cdot L}{(N \cdot L)(1 - k) + k} \\
k &= \frac{(R + 1)^2}{8}
\end{aligned}$$

2. Network Architecture

We have talked about the motivations, design and core components of our depth prediction network and SVBRDF prediction network in Sec. 3.1 and Sec. 3.2 in the paper. We now introduce the network architectures in detail as shown in Fig. 1.

Depth prediction network. As discussed in Sec. 3.1 in the paper, the depth prediction network consists of three parts: the feature extractor $\mathcal{F}$, the correspondence predictor $\mathcal{C}$ and the guidance map extractor $\mathcal{G}$. The feature extractor $\mathcal{F}$ and the correspondence predictor $\mathcal{C}$ are used to predict the initial depth map D_i'; the guidance map extractor is applied to refine D_i' using a guided filter [5] to obtain the final depth D_i. Figure 1 shows the details of these sub-networks in the first row.

We use the feature extractor and the correspondence predictor to regress the initial depth, similar to [7]. In particular, the feature extractor $\mathcal{F}$ is a 2D U-Net that consists of multiple downsampling and upsampling convolutional layers with skip links, group normalization (GN) [6] layers and ReLU activation layers; it extracts per-view image feature maps with 16 channels.

To predict the depth D_i at reference view i, we uniformly sample 128 frontal parallel depth planes at depth $d_1, d_2, \ldots, d_{128}$ in front of that view within a pre-defined depth range $[d_1, d_{128}]$ that covers the target object we want to capture. We project the feature maps from all views onto every depth plane at view i using homography-based warping to construct the plane sweep volume of view i. We then build a cost volume by calculating the variance of the warped feature maps over views at each plane. The correspondence predictor $\mathcal{C}$ is a 3D U-Net that processes this cost volume; it has multiple downsampling and upsampling 3D convolutional layers with skip links, GN layers and ReLU layers. The output of $\mathcal{C}$ is a 1-channel volume, and we apply softmax on this volume across the depth planes to obtain the per-plane depth probability maps $P_1, \ldots, P_{128}$ of the 128 depth planes; these maps indicate the probability of the depth of a pixel being the depth of each plane. A depth map is then regressed by linearly combining the per-plane depth values weighted by the per-plane depth probability maps:

$$D_i' = \sum_{q=1}^{128} P_q * d_q. \tag{2}$$

We apply the guidance map extractor $\mathcal{G}$ to refine the initial depth D_i'. $\mathcal{G}$ is a 2D U-Net that outputs a 1-channel feature map. We use the output feature map as a guidance map to filter the initial depth D_i' and obtain the final depth D_i.

SVBRDF prediction network. We have discussed the SVBRDF prediction network in Sec. 3.2, and shown the overall architecture, input and output in Fig. 2 and Fig. 3 of

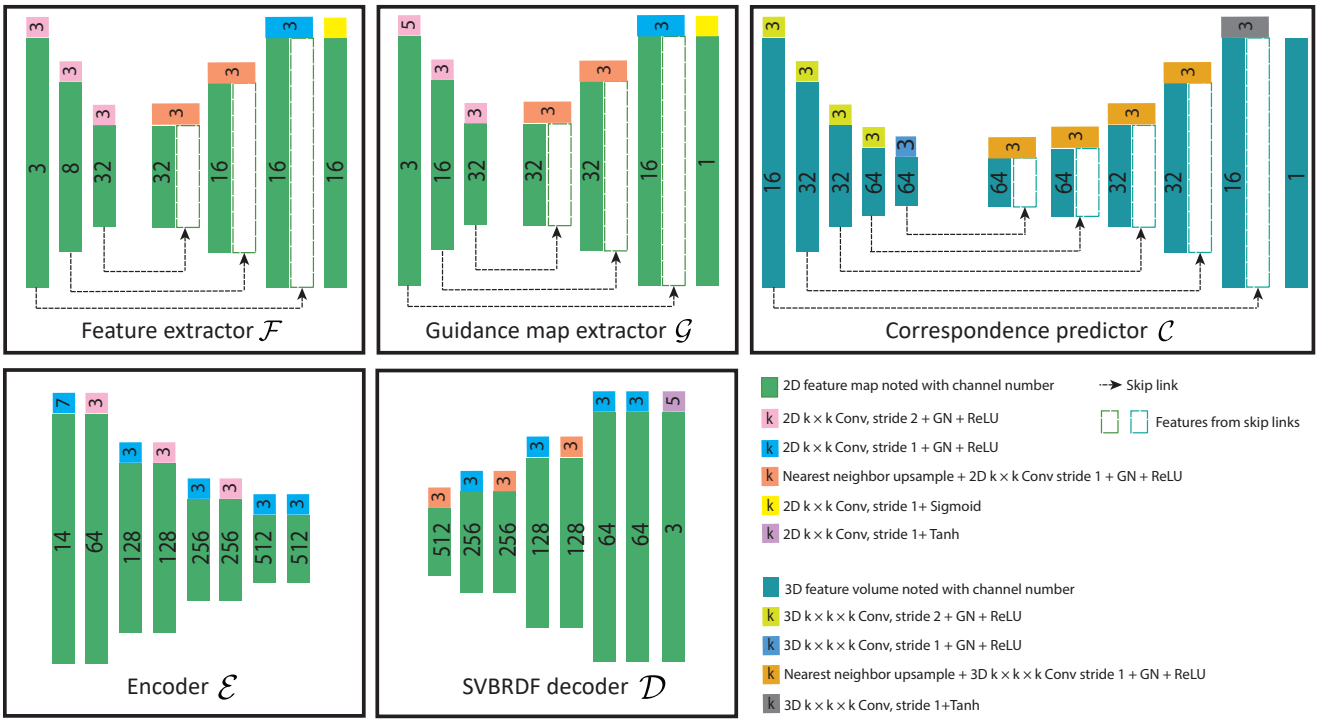

Figure 1: Our network architecture.

the paper. We now introduce the details of the encoder $\mathcal{E}$ and the SVBRDF decoder $\mathcal{D}$ in Fig. 1 (bottom row). Specifically, the encoder consists of a set of convolutional layers, followed by GN and ReLU layers; multiple convolutional layers with a stride of 2 are used to downsample the feature maps three times. The decoder upsamples the feature maps three times with nearest-neighbor upsampling, and applies convolutional layers, GN and ReLU layers to process the feature maps at each upsampling level. As discussed in Sec. 3.2 of the paper, we apply four decoders with the same architecture, which are connected with the same encoder, to regress three BRDF components and the normal map at each input view.

3. Comparison on SVBRDF Prediction

In Sec. 4.1 and Tab. 1 of the paper, we have shown quantitative comparisons on synthetic data between our network, the naïve U-Net and a single-image SVBRDF prediction network proposed by Li et al. [3]. We now demonstrate qualitative comparisons between these methods on both synthetic and real examples in Fig. 5, Fig. 6, Fig. 7 and Fig. 8. From these figures, we can see that the naïve U-Net produces noisy normals and the single-view method [3] produces normals with very few details, whereas our predicted normals are of much higher quality, especially in regions where there are serious occlusions (indicated by the *red* arrow). In contrast, as reflected by the comparison on synthetic data in Fig. 5 and Fig. 6, our predictions are more accurate and more consistent with the ground truth than the other methods. These results demonstrate that our novel network architecture (see Sec. 3.2 in the paper) allows for effective aggregation of multi-view information and leads to high-quality per-view SVBRDF estimation.

4. Comparison on Geometry Reconstruction

In Fig. 6 of the paper, we compare our optimized geometry against the optimized result from Nam et al. [4] that uses the same initial geometry as ours. We show additional comparisons on real data in Fig. 2. Similar to the comparison in the paper, our optimized geometry is of much higher quality than Nam et al. with more fine-grained details and fewer artifacts.

5. Additional Ablation Study

In this section, we demonstrate additional experiments to justify the design choices in our pipeline, including input variants of the SVBRDF estimation network, non-rigid warping and per-vertex refinement.

Network inputs. Our SVBRDF network considers the input image (I_i), the warped images ($I_{i\leftarrow j}$), the light/viewing (which are collocated) direction maps (L_i and $L_{i\leftarrow j}$), and the depth maps ($Z_{i\leftarrow j}$ and $Z^*_{i\leftarrow j}$) as inputs (please refer to

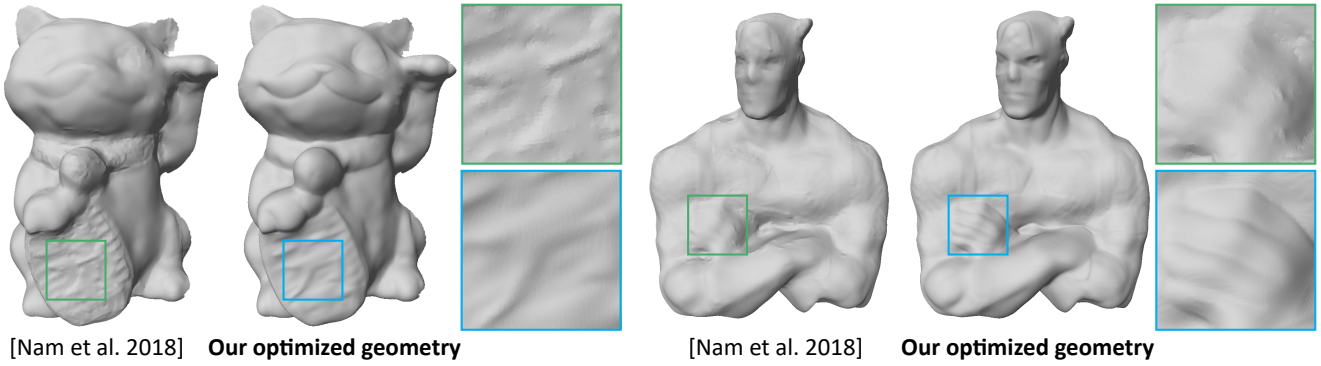

Figure 2: Comparison with Nam et al. [4] on geometry optimization. Our results have more fine-grained details and fewer artifacts.

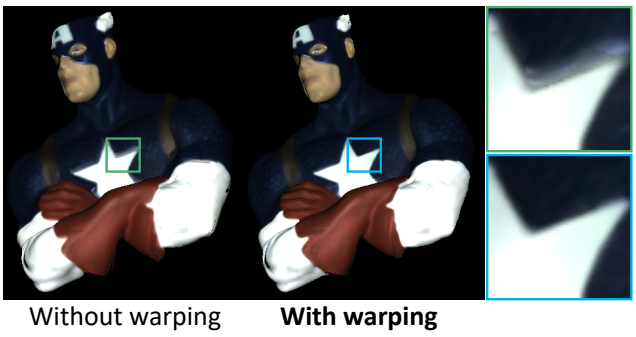

Figure 3: Comparison between optimizations with and without per-view warping. Our method with warping removes the ghosting artifacts around the edges.

Network input	Diffuse	Normal	Roughness	Specular
$I_{i\leftarrow j}$	0.0081	0.0456	0.0379	0.0098
$I_i, I_{i\leftarrow j}$	0.0071	0.0363	0.0304	0.0109
$I_i, I_{i\leftarrow j}, Z_{i\leftarrow j}, Z^*_{i\leftarrow j}$	0.0063	0.0321	0.0306	0.0098
$I_i, I_{i\leftarrow j}, L_i, L_{i\leftarrow j}$	**0.0061**	**0.0304**	0.0299	0.0093
Ours full	**0.0061**	**0.0304**	**0.0275**	**0.0086**

Table 1: Quantitative comparisons between networks trained with different inputs on the synthetic test set.

Sec. 3.2 in the paper for details of these input components). We verify the effectiveness of using these inputs by training and comparing multiple networks with different subsets of the inputs. In particular, we compare our full model against a network that uses only the warped image $I_{i\leftarrow j}$, a network that considers both $I_{i\leftarrow j}$ and the reference image I_i, a network that uses the reference image, warped image and the depth, and a network that uses the reference image, warped image, and the viewing directions. Table. 1 shows the quantitative comparisons between these networks on the synthetic testing set. The network using a pair of images (I_i, $I_{i\leftarrow j}$) improves the accuracy for most of the terms over the one that uses only the warped image ($I_{i\leftarrow j}$), which reflects the benefit of involving multi-view cues in the encoder network. On top of the image inputs, the two networks that involve additional depth information ($Z_{i\leftarrow j}$, $Z^*_{i\leftarrow j}$) and the viewing directions (L_i, $L_{i\leftarrow j}$) both obtain better performance than the image-only versions, which leverage visibility cues and photometric cues from the inputs respectively. Our full model is able to leverage both cues from multi-view inputs and achieves the best performance.

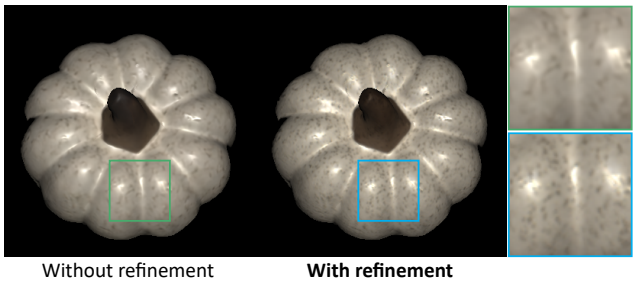

Figure 4: Comparison on results with and without per-vertex refinement. With the refinement, our method is able to recover high-frequency details such as the spots on the object.

Per-view warping. Due to potential inaccuracies in the geometry, the pixel colors of a vertex from different views may not be consistent. Directly minimizing the difference between the rendered color and the pixel color of each view will lead to ghosting artifacts, as shown in Fig. 3. To solve this problem, we propose to apply a non-rigid warping to each view. From Fig. 3 we can see that non-rigid warping can effectively tackle the misalignments and leads to sharper edges.

Per-vertex refinement. As shown in Fig. 4, the image rendered using estimated SVBRDF without per-vertex refine-

ment loses high-frequency details such as the tiny spots on the pumpkin, due to the existence of the bottleneck in our SVBRDF network. In contrast, the proposed per-vertex refinement can successfully recover these details and reproduces more faithful appearance of the object.

References

[1] Brent Burley. Physically-based shading at disney. In *ACM SIGGRAPH 2012 Courses*, SIGGRAPH '12, pages 10:1–10:7, 2012. 1

[2] Brian Karis and Epic Games. Real shading in unreal engine 4. *Proc. Physically Based Shading Theory Practice*, 2013. 1

[3] Zhengqin Li, Zexiang Xu, Ravi Ramamoorthi, Kalyan Sunkavalli, and Manmohan Chandraker. Learning to reconstruct shape and spatially-varying reflectance from a single image. In *SIGGRAPH Asia 2018*, page 269. ACM, 2018. 2, 5, 6, 7, 8

[4] Giljoo Nam, Joo Ho Lee, Diego Gutierrez, and Min H Kim. Practical SVBRDF acquisition of 3D objects with unstructured flash photography. In *SIGGRAPH Asia 2018*, page 267. ACM, 2018. 2, 3

[5] Huikai Wu, Shuai Zheng, Junge Zhang, and Kaiqi Huang. Fast end-to-end trainable guided filter. In *CVPR*, pages 1838–1847, 2018. 1

[6] Yuxin Wu and Kaiming He. Group normalization. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 3–19, 2018. 1

[7] Zexiang Xu, Sai Bi, Kalyan Sunkavalli, Sunil Hadap, Hao Su, and Ravi Ramamoorthi. Deep view synthesis from sparse photometric images. *ACM Transactions on Graphics*, 38(4):76, 2019. 1

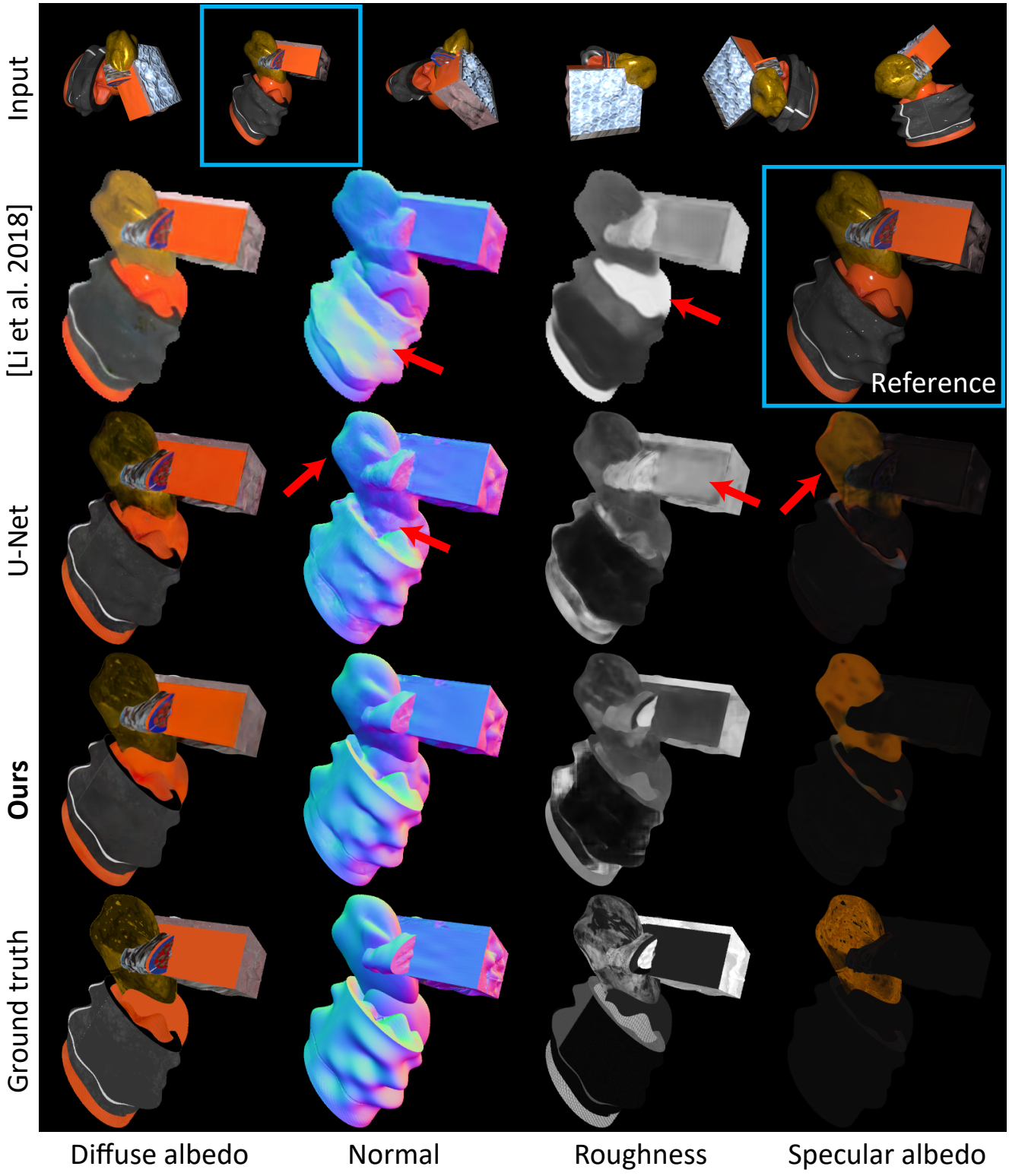

Figure 5: Qualitative comparison of SVBRDF estimation on synthetic data. Note that Li et al. [3] do not predict specular albedo.

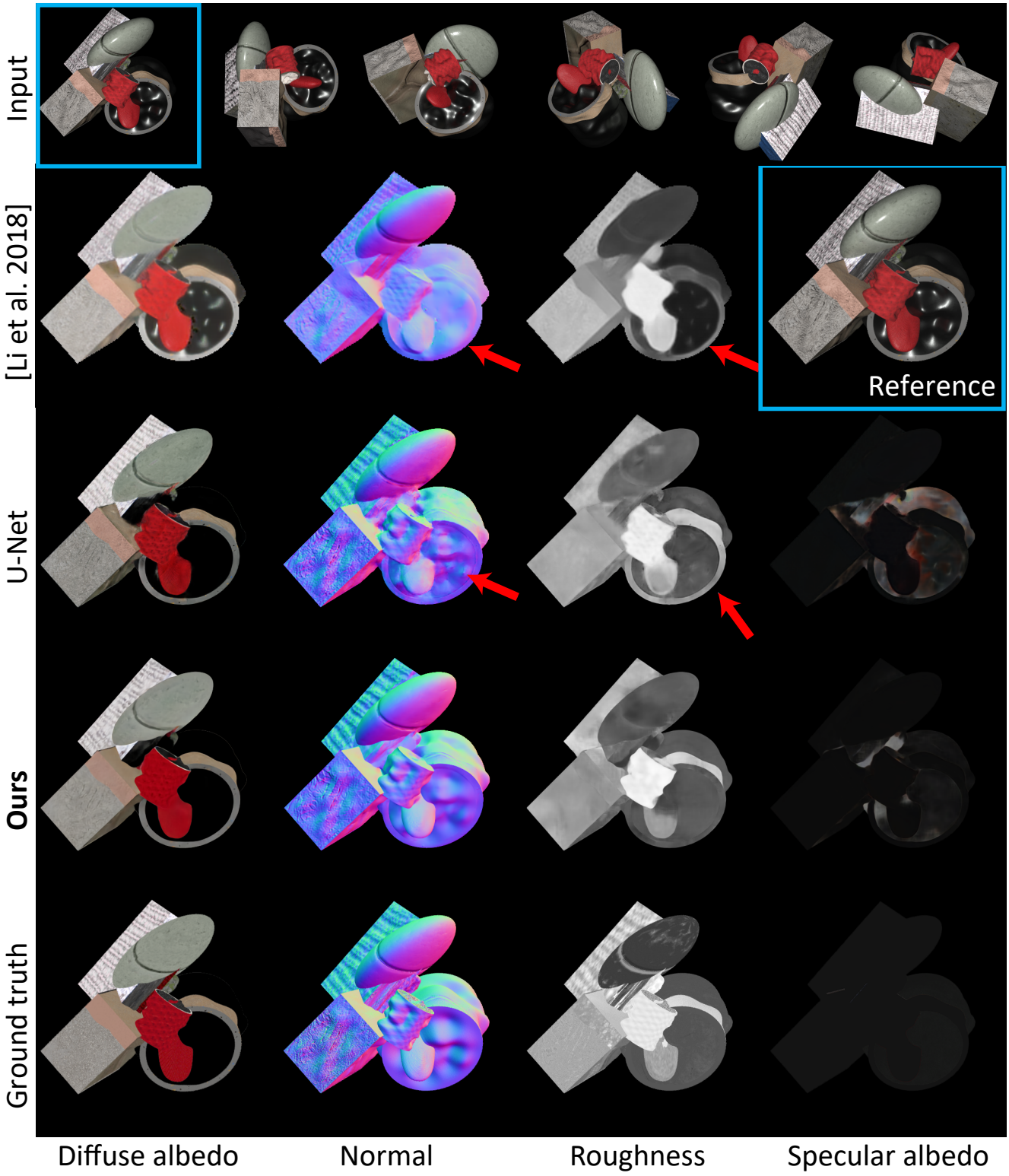

Figure 6: Qualitative comparison of SVBRDF estimation on synthetic data. Note that Li et al. [3] do not predict specular albedo.

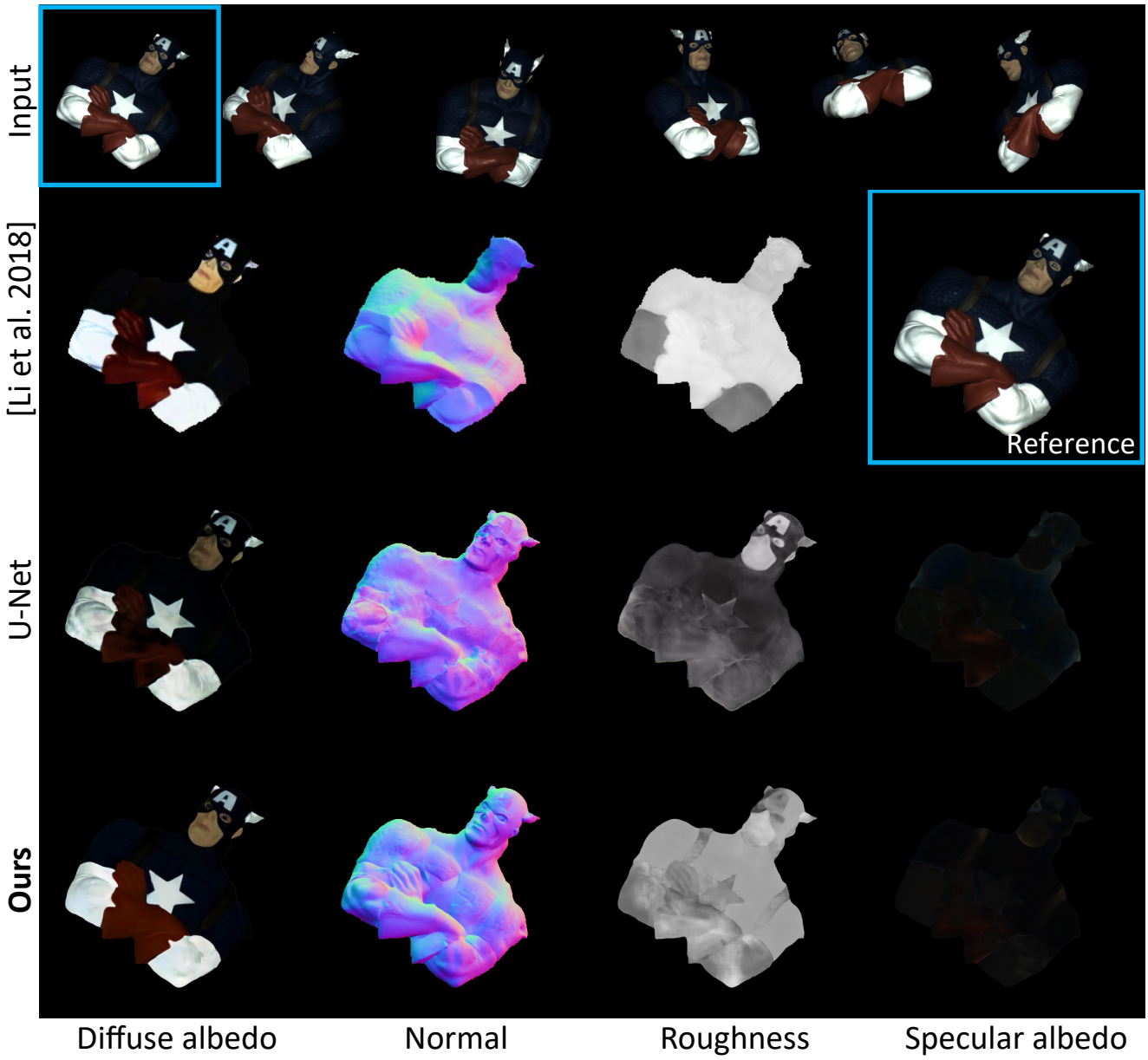

Figure 7: Qualitative comparison of SVBRDF estimation on real data. Note that Li et al. [3] do not predict specular albedo.

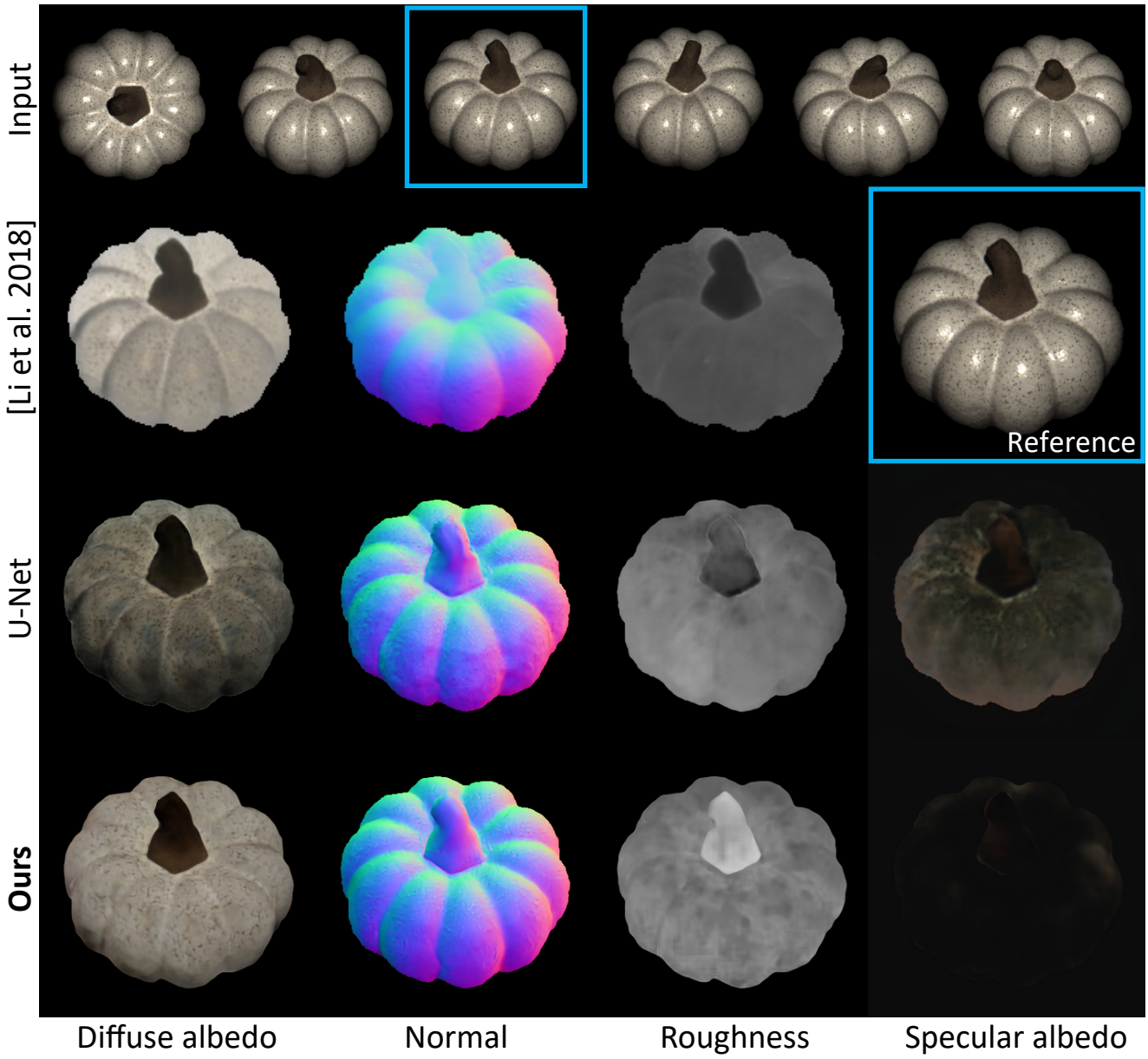

Figure 8: Qualitative comparison of SVBRDF estimation on real data. Note that Li et al. [3] do not predict specular albedo.