

Supplementary material for Neural Reflectance for Shape Recovery with Shadow Handling

Junxuan Li ^{1,2}, Hongdong Li ¹
 Australian National University ¹ Data61, CSIRO ²
 {junxuan.li, hongdong.li}@anu.edu.au

1. Implementation Details

Network architecture In Fig. 1, we show the detail structure of our three MLPs : (a) specularity basis modeling S_Φ ; (b) surface modeling M_Θ , (c) depth modeling Z_Ψ . The design of these structures are inspired by a recent work NeRF [7].

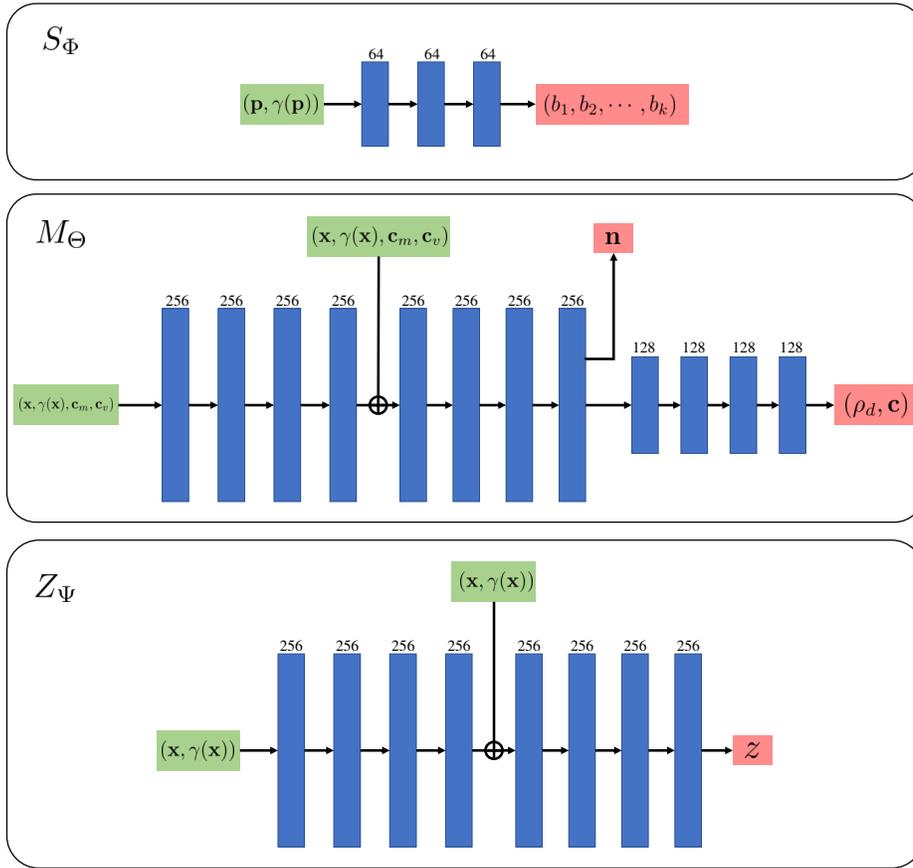


Figure 1. **Network architecture of our three MLPs:** S_Φ , M_Θ , Z_Ψ . In this figure, inputs of the network are shown in the green blocks; outputs are shown in the red blocks. The blue blocks represent the fully-connected layers with its size of the hidden channels stated on the top. All fully-connected layers are followed by a ReLU activation layer, except the output layers. The “ \oplus ” in the middle of the M_Θ , Z_Ψ network denotes the vector concatenation: we add a skip connection after the fourth layer of M_Θ , Z_Ψ , and concatenate its output features with the input.

Positional encoding of input of M_Θ and Z_Ψ The inputs of surface modeling M_Θ and depth modeling Z_Ψ are the pixel coordinates \mathbf{x} . We adopt the positional encoding strategy [7] to embed the input coordinates $\mathbf{x} \in \mathbb{R}^2$ into a higher space $\mathbf{x} \in \mathbb{R}^{4m}$:

$$\gamma(\xi) = (\sin(2^0\pi\xi), \cos(2^0\pi\xi), \dots, \sin(2^{m-1}\pi\xi), \cos(2^{m-1}\pi\xi)). \quad (1)$$

In practice, we first normalized the coordinates to range $(-1, 1)$, then apply the above encoding function with $m = 10$ to each of the two coordinate values in \mathbf{x} . Then, we concatenate the coordinate and its embeddings as $(\mathbf{x}, \gamma(\mathbf{x}))$ to be the input of the two MLPs.

Color mean and variation for M_Θ albedo estimation Since the albedo estimation of an object surface exists an ambiguities in its scales. We compute the mean value and variation value $\mathbf{c}_m, \mathbf{c}_v$ of the images and concatenate them to the input of M_Θ for albedo estimation.

Positional encoding of input of S_Φ Rusinkiewicz [9] reparameterized the BRDF as a function of the half-vector \mathbf{h} (*i.e.* the half-vector between lighting and viewing direction). This half-vector-based parameterization is further evaluated and discussed by [2, 8], where they found that a simplified isotropic BRDF can be modeled in two parameters (θ_h, θ_d) , where $\theta_h = \arccos(\mathbf{n}^T \mathbf{h}), \theta_d = \arccos(\mathbf{v}^T \mathbf{h})$. In our method, we take the cosine value of these two variables as the input:

$$\mathbf{p} = (\mathbf{n}^T \mathbf{h}, \mathbf{v}^T \mathbf{h}). \quad (2)$$

Then, the input \mathbf{p} is further encoded by $\gamma(\mathbf{p})$ with $m = 3$. Likewise, we concatenate \mathbf{p} and its embeddings as $(\mathbf{p}, \gamma(\mathbf{p}))$ to be the input of the specularity basis modeling S_Φ .

2. Additional ablation study

In this section, we compare a variant of our surface modeling network M_Θ . The original model we use in the paper is denoted as the baseline. The equation for baseline surface modeling (*i.e.* the one used in main paper) is: $\mathbf{n}, \rho_d, \mathbf{c} = M_\Theta(\mathbf{x})$. **Surface Modeling Variant 1:** directly output depth, diffuse albedo and specular weights rather than outputting normals.

$$z, \rho_d, \mathbf{c} = M_\Theta^1(\mathbf{x}), \quad (3)$$

Table 1. Quantitative comparison on different variants of surface modeling. The metric here is MAE; lower is preferred. Below, we present the average MAE of ten objects in DiLiGenT.

Methods	Avg.
Baseline	6.50
Variant 1	7.56

Our experiments in Table 1 show that the baseline model (outputting normals, albedo and specular weights by a single network) has the best performance. Our analysis is that: Comparing to **Variant 1**: The surface normal is closely related to the photometric appearance of an object. By directly outputting normal, our baseline network can achieve a lower photometric appearance (reconstruction) loss. If we directly output depth, we need to apply an additional step to compute the finite difference to get normal. Hence, directly outputting normal will help the training of the network to minimize the reconstruction loss.

3. Evaluation on re-rendered images

Quantitatively evaluation We re-rendered the observed image with our estimated reflectances, and ground truth lights. We compare our re-rendered images with ACLS [1]. Note that ACLS’s BRDF fitting results are provided by Shi *et al.* [10], where the calibrated lightings, and the ground truth normal are used when fitting the BRDF. In comparison, our method only takes the calibrated lightings at the input. The results are shown in Tab. 2. Our method achieves better reconstruction results on average. Our method performs particularly well on shiny objects and objects with variously materials, such as “cow”, “goblet”, “harvest”, and “reading”. Visual comparisons are shown in Fig. 2.

Table 2. Quantitative comparison on the re-rendered images with our estimated BRDF. The metric here is peak signal-to-noise ratio (PSNR); higher is preferred.

	ball	bear	buddha	cat	cow	goblet	harvest	pot1	pot2	reading	Average
ACLS [1]	40.27	46.75	40.27	46.87	44.35	42.67	33.90	51.55	52.06	30.60	42.93
Ours	37.82	47.96	41.14	46.31	45.85	43.56	36.24	51.40	52.36	31.58	43.42

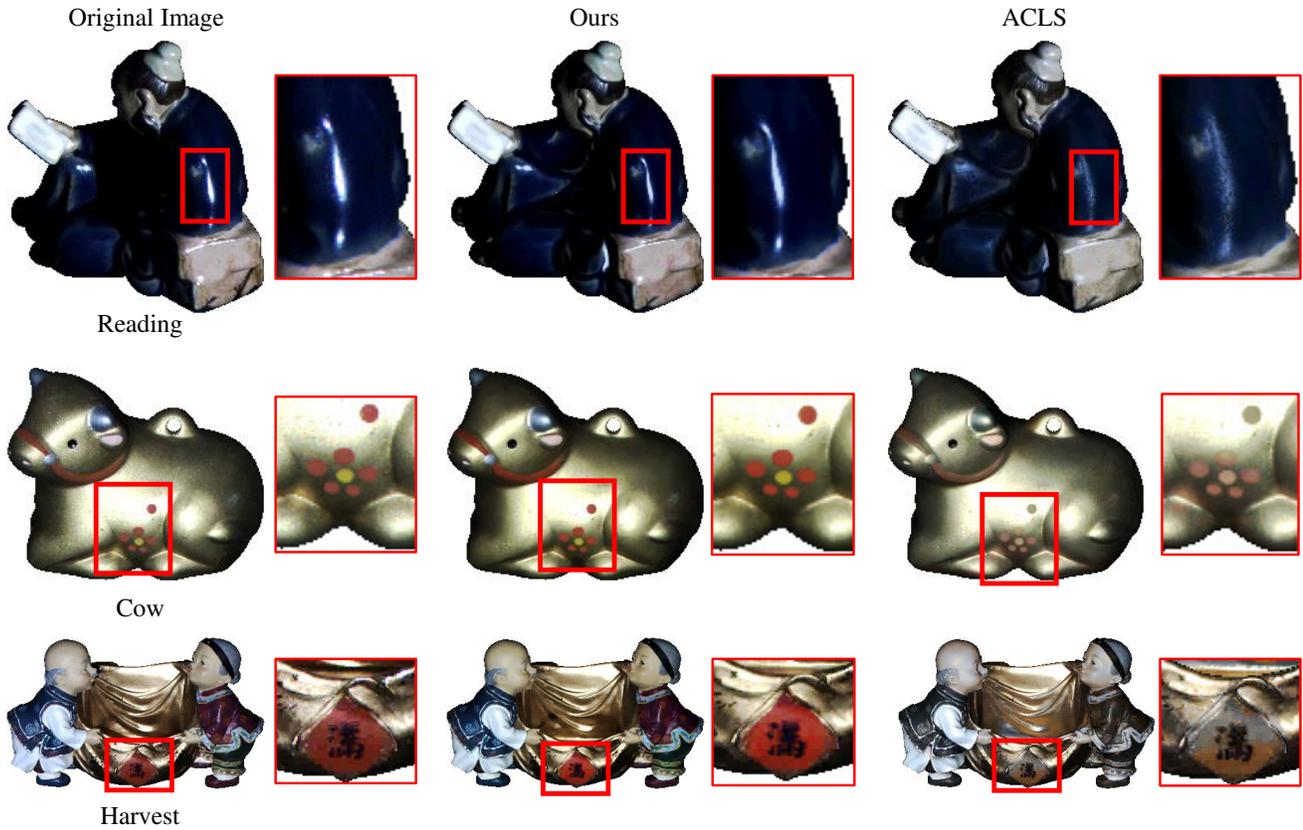


Figure 2. Re-rendered image by our method and ACLS [1]. From left to right, we showcase the original image, the re-rendered image using our estimated neural svBRDFs, and the re-rendered image by ACLS [1], respectively. Our method performs well on the specularities on “Reading”, while ACLS failed to recover these highlights. Our method is also able to recover the spatially-varying materials, see the “Cow” and “Harvest”, while ACLS failed to recover various materials on surfaces.

4. Material editing

Our method opens up the possibilities to edit materials of objects. In the following figures, we select some surface points from the observed images and use the estimate reflectance (svBRDF) to re-render several new objects.

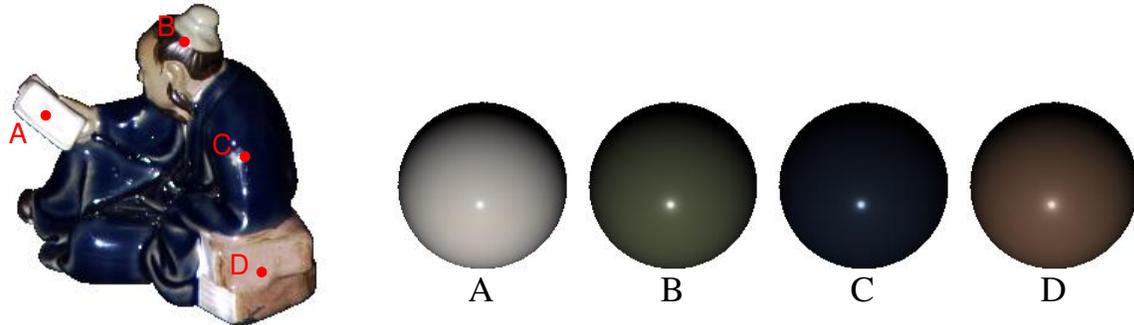


Figure 3. **Visualization on the estimated svBRDFs.** This figure shows the estimated svBRDFs of the surface points on the objects. On the left is the observed image of “Reading”. We select four different surface points on the object and showcase our re-rendered BRDF spheres of those points on the right side. For better visualization, we normalize the BRDF spheres to have the maximum intensity to be 1. The observed images are also scaled up for visualization.

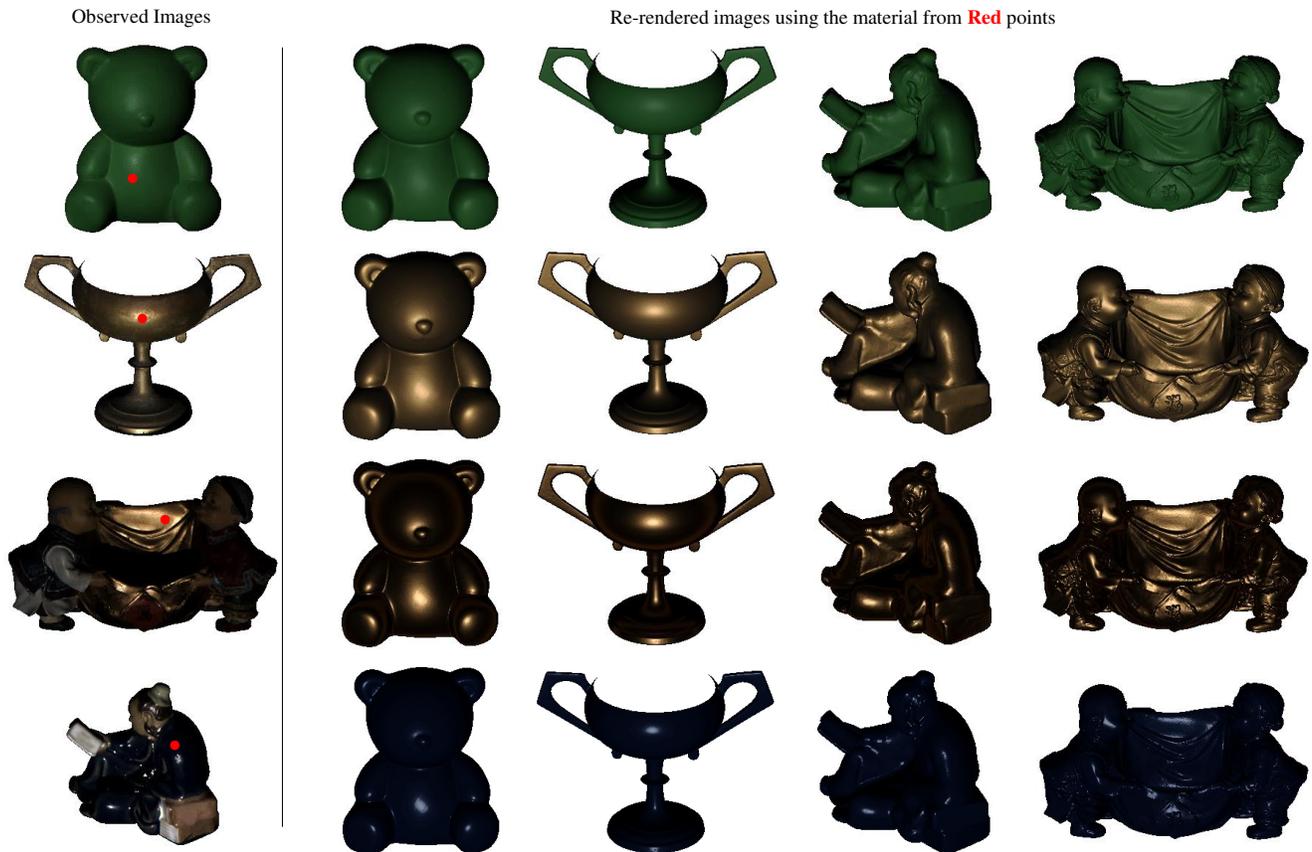


Figure 4. **Material editing.** For each row, the left-most images are the observed images from DiLiGenT dataset. We use the estimated reflectance from the red points denoted in the left-most images to re-render several objects and present them on the right.

5. BRDF evaluation on synthetic dataset.

In this section, we evaluate our method on a publicly available synthetic dataset¹ proposed by Chen et al [5]. The dataset was rendered using the physically-based raytracer Mitsuba with MERL [6] as the BRDFs. We showcase our results on “Armadillo” with “alum-bronze” as the material in Fig. 5. From the result, even if the shape is as complicated as “Armadilla”, our method can still recover the normals very well (with MAE 3.60°). We further plot a slice of our estimated BRDF curve and the ground truth BRDF curve in the right of Fig. 5. We can see that our estimated BRDF is very close to the ground truth BRDF, which demonstrates that our method is robust in material recovery.

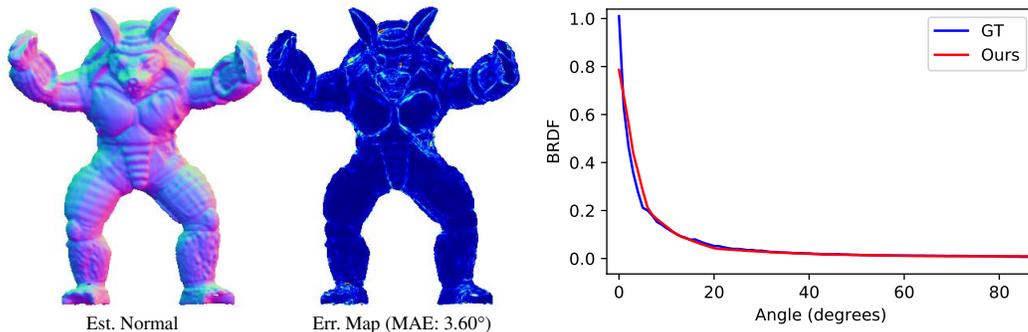


Figure 5. From left to right, we showcase the estimated normal, error map, and our recovered slice of BRDF curve of “Armadillo” with “alum-bronze” as the material.

6. Additional Results

Additional Results on Normal Estimation In Fig. 6, we present the normal estimation results on three specular objects: “Goblet”, “Reading” and “Harvest” from DiLiGenT [10]. These results demonstrate that our method is taking advantage of the information that the specularities provide. Hence, we can estimate the normal accurately on specular regions.

Visualization on each terms of the rendering equation Recall the rendering equation (Eq.(1) in the main paper) is defined by

$$I = s\rho(\mathbf{l}, \mathbf{v}, \mathbf{n}) \max(\mathbf{l}^T \mathbf{n}, 0), \quad \rho(\mathbf{l}, \mathbf{v}, \mathbf{n}) = \rho_d + \rho_s, \quad (4)$$

where ρ_d is the diffuse albedo; ρ_s is the specularities; s is the shadows; $\max(\mathbf{l}^T \mathbf{n}, 0)$ is the shading term. In Fig. 7, we present the visualization of our estimation on these terms.

Additional Results on Shadows Estimation In Fig. 8, we showcase the estimated shadows and specularities under different light directions.

Additional Results on Other Real-world Dataset We also test our method on two other challenging real-world datasets: Gourd&Apple dataset [1] and Light Stage Data Gallery [3], as shown in Fig. 9 and Fig. 10 separately. Both of these two datasets do not provide ground truth normal for evaluation. Hence, we provide the visualization of the estimated normal, diffuse albedo, and specular map on these datasets. Our method correctly recovers the shape and materials of different objects. It also demonstrates that our method is robust on different objects with different materials.

7. Ethics Statement

With the advancement of photometric stereo, anyone can easily capture the 3D shape of a person’s face. The inverse rendering technique allows the user to alter the shape and appearance of an individual’s face. The acquisition and alteration of such personal information, if without their consent, may lead to privacy and security breaching. Care must be taken to mitigate the potential risk of abusing this technique.

¹<https://github.com/guanyingc/UPS-GCNet>

References

- [1] Neil Alldrin, Todd Zickler, and David Kriegman. Photometric stereo with non-parametric and spatially-varying reflectance. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008. [3](#), [5](#), [10](#)
- [2] Brent Burley and Walt Disney Animation Studios. Physically-based shading at disney. In *ACM SIGGRAPH*, volume 2012, pages 1–7. vol. 2012, 2012. [2](#)
- [3] Charles-Félix Chabert, Per Einarsson, Andrew Jones, Bruce Lamond, Wan-Chun Ma, Sebastian Sylwan, Tim Hawkins, and Paul Debevec. Relighting human locomotion with flowed reflectance fields. In *ACM SIGGRAPH 2006 Sketches*, pages 76–es. 2006. [5](#), [11](#)
- [4] Guanying Chen, Kai Han, and Kwan-Yee K Wong. Ps-fcn: A flexible learning framework for photometric stereo. In *European Conference on Computer Vision*, pages 3–19. Springer, 2018. [7](#)
- [5] Guanying Chen, Michael Waechter, Boxin Shi, Kwan-Yee K Wong, and Yasuyuki Matsushita. What is learned in deep uncalibrated photometric stereo? In *European Conference on Computer Vision*, pages 745–762. Springer, 2020. [5](#)
- [6] Wojciech Matusik, Hanspeter Pfister, Matt Brand, and Leonard McMillan. A data-driven reflectance model. *ACM Transactions on Graphics*, 22(3):759–769, July 2003. [5](#)
- [7] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. Nerf: Representing scenes as neural radiance fields for view synthesis. In *European Conference on Computer Vision*, pages 405–421. Springer, 2020. [1](#), [2](#)
- [8] Romain Pacanowski, Oliver Salazar Celis, Christophe Schlick, Xavier Granier, Pierre Poulin, and Annie Cuyt. Rational brdf. *IEEE transactions on visualization and computer graphics*, 18(11):1824–1835, 2012. [2](#)
- [9] Szymon M Rusinkiewicz. A new change of variables for efficient brdf representation. In *Eurographics Workshop on Rendering Techniques*, pages 11–22. Springer, 1998. [2](#)
- [10] Boxin Shi, Zhipeng Mo, Zhe Wu, Dinglong Duan, Sai Kit Yeung, and Ping Tan. A benchmark dataset and evaluation for non-lambertian and uncalibrated photometric stereo. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2018. [3](#), [5](#)
- [11] Tatsunori Taniai and Takanori Maehara. Neural inverse rendering for general reflectance photometric stereo. In *International Conference on Machine Learning*, pages 4864–4873, 2018. [7](#)
- [12] Robert J Woodham. Photometric method for determining surface orientation from multiple images. *Optical engineering*, 19(1):191139, 1980. [7](#)

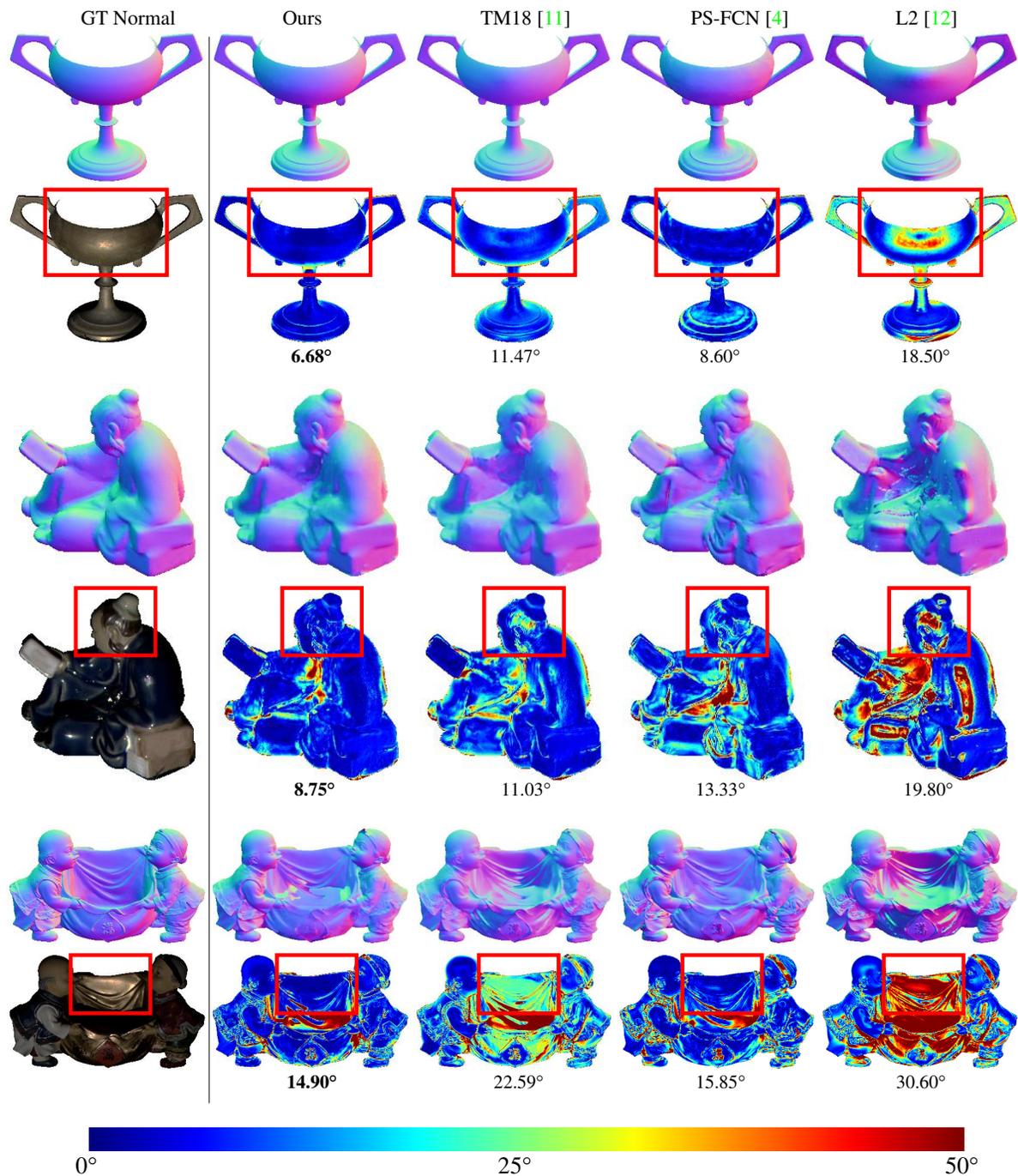


Figure 6. **Normal estimation on specular objects: “Goblet”, “Reading” and “Harvest”**. As shown in the observed image of these three objects, the “Goblet” is mostly made of metallic materials; “Reading” and “Harvest” present many specular effects over the clothes. Our method achieves the best performance in all these three objects, especially in those regions with high specularities. Please look at the red windows in the error map. “Reading” contains many specularities over its cloth and its head. While all the other methods suffer on these specularities, our method still performs well in these regions, especially on the head. The cloth of “Harvest” in the center also presents significant specular effects. While the other self-supervised method TM18 [11] failed on these regions, our method correctly recover the surface normal. These results demonstrate that our method is taking advantage of the information that the specularities provide. Hence, we can estimate the normal accurately on specular regions.

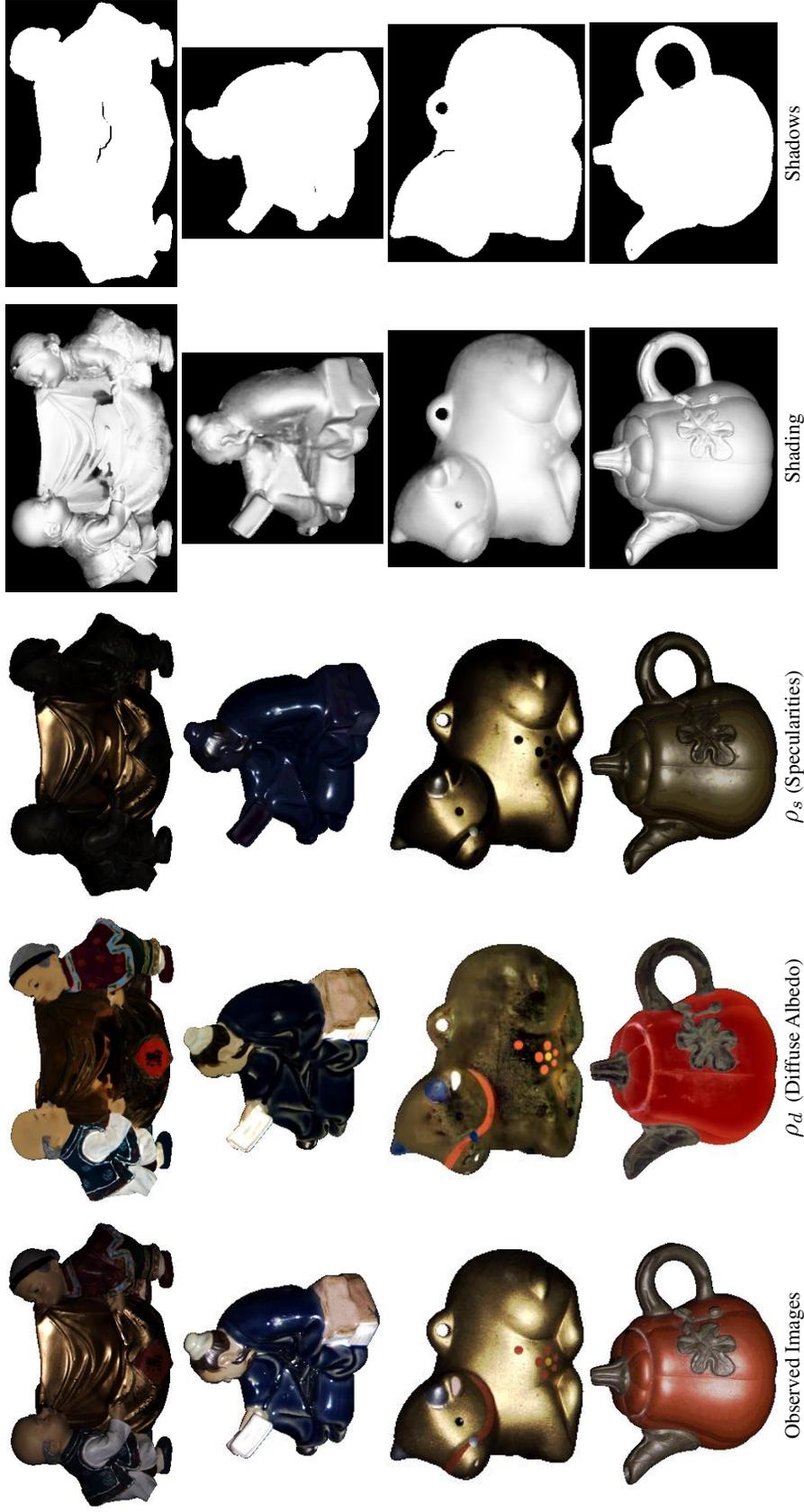


Figure 7. **Visualization on each terms of the rendering equation.** In the above images, the first column displays the observed images of the objects. The second and third column are the estimated diffuse albedo ρ_d , and specular components ρ_s . The fourth column is the shading map, which is computed by the dot product between light direction and surface normal ($l^T n$). The last column is the estimated shadows, corresponding to s in the equation. As seen from the diffuse albedo map in the cloth of the “Harvest” and the small patterns on the “Cow”, the estimated diffuse albedo map retains the objects’ fine details. These results demonstrate that our method can recover the fine details of the svBRDF map in the object. Note that, for better visualization, the images we selected here are all illuminated by a front light source. Hence, as can be seen in the observed images, there is little shadows.

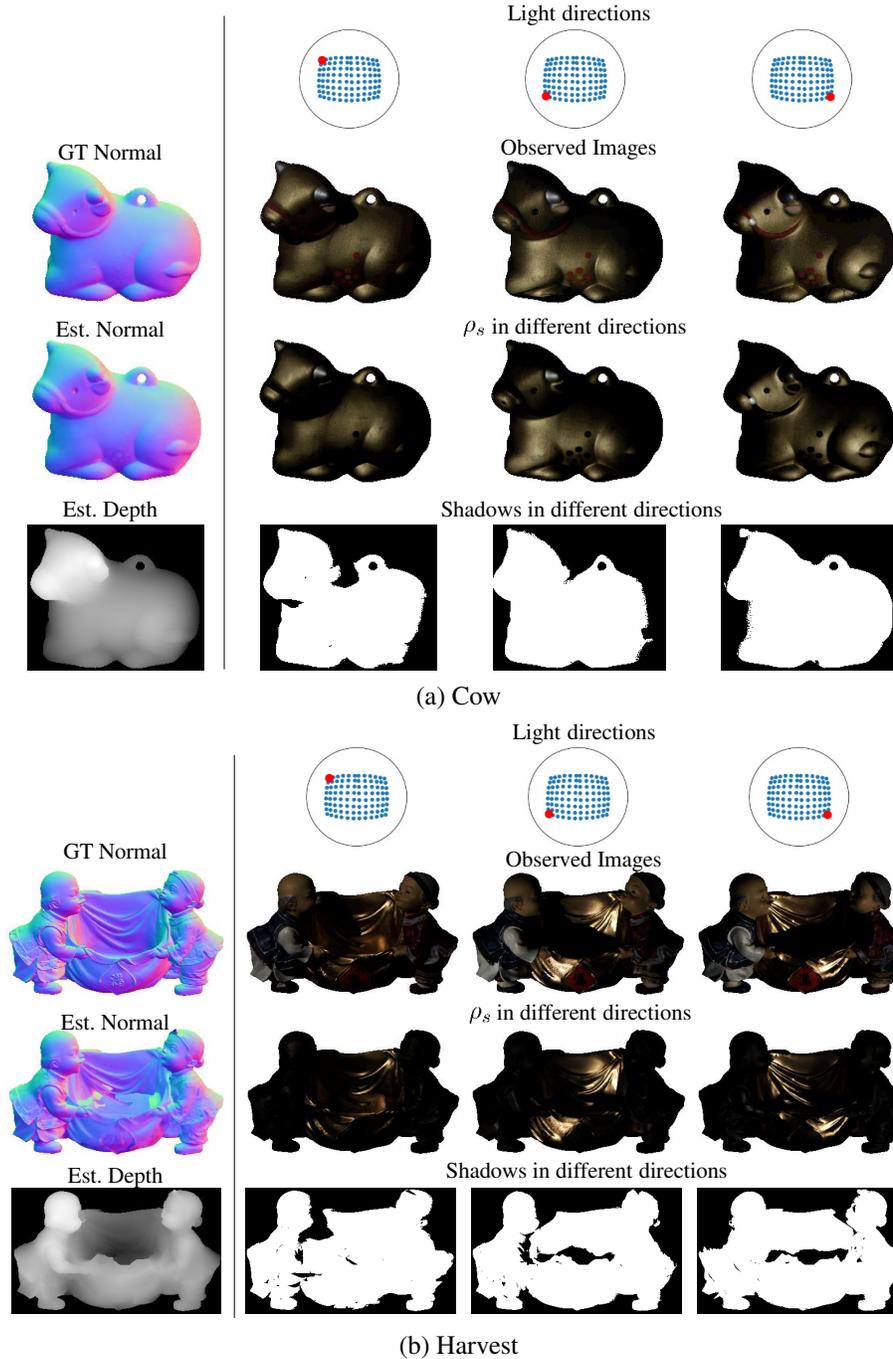


Figure 8. **Estimated specularities and shadows under different illuminations.** The leftmost column presents the ground truth normal and our estimated normal and depth as a reference of the object’s geometry. We show the estimated specular components ρ_s and estimated shadows under three different extreme lighting directions in the right-three columns. In “Cow”, the object is generally smooth, and our estimation of the shadows also visually match the observed images. “Harvest” has a complex geometry and consists of many depth discontinuities over the surface. As discussed in the Sec. 6 of the main paper, our method is influenced by these regions and will generate a “shallower” depth map. Hence, the estimated shadows are generally under-estimated.

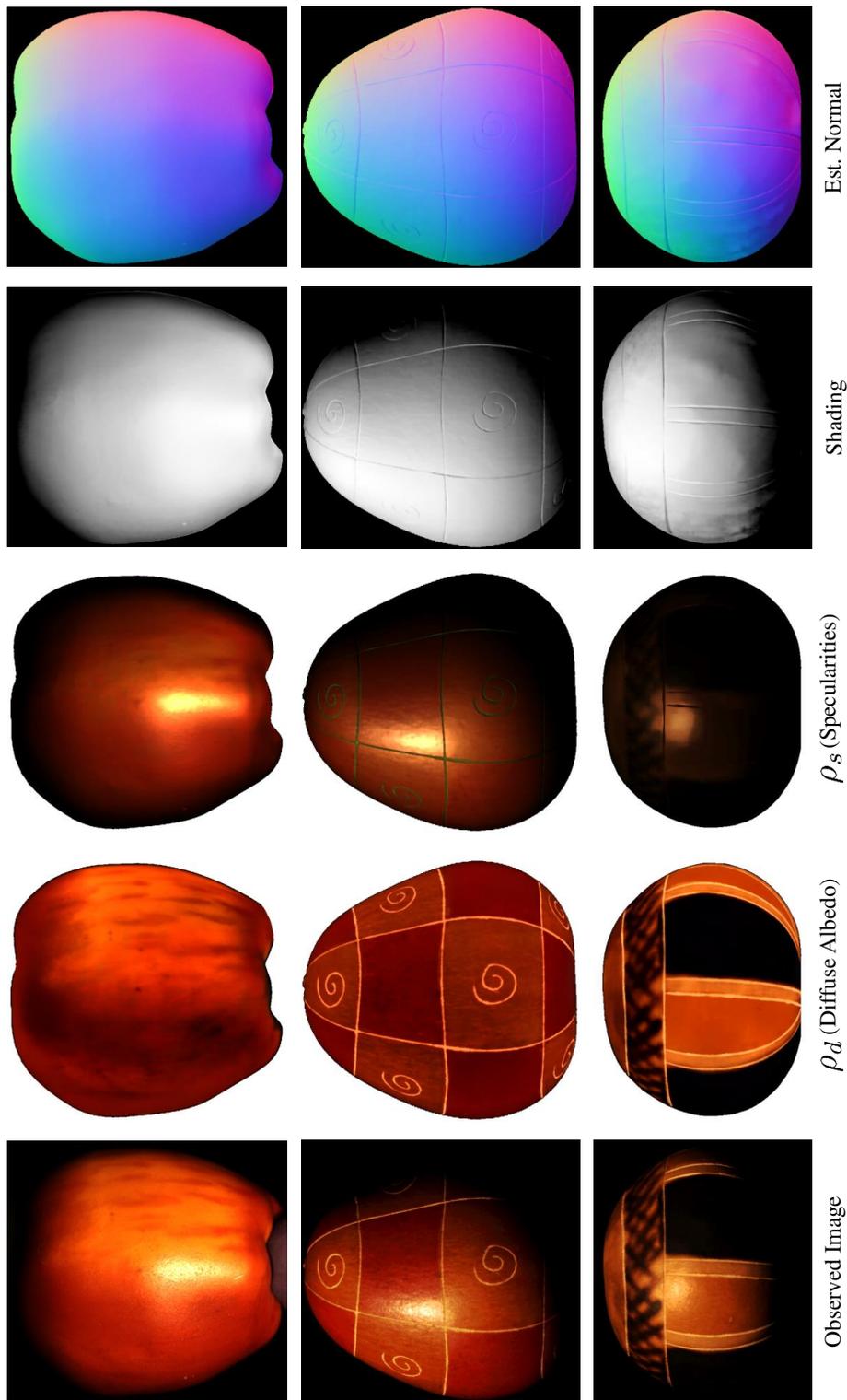


Figure 9. **Results on Gourd&Apple dataset [1].** The columns from left to right are the observed images, our estimated diffuse albedo, specularities, shading, and surface normal of the objects.

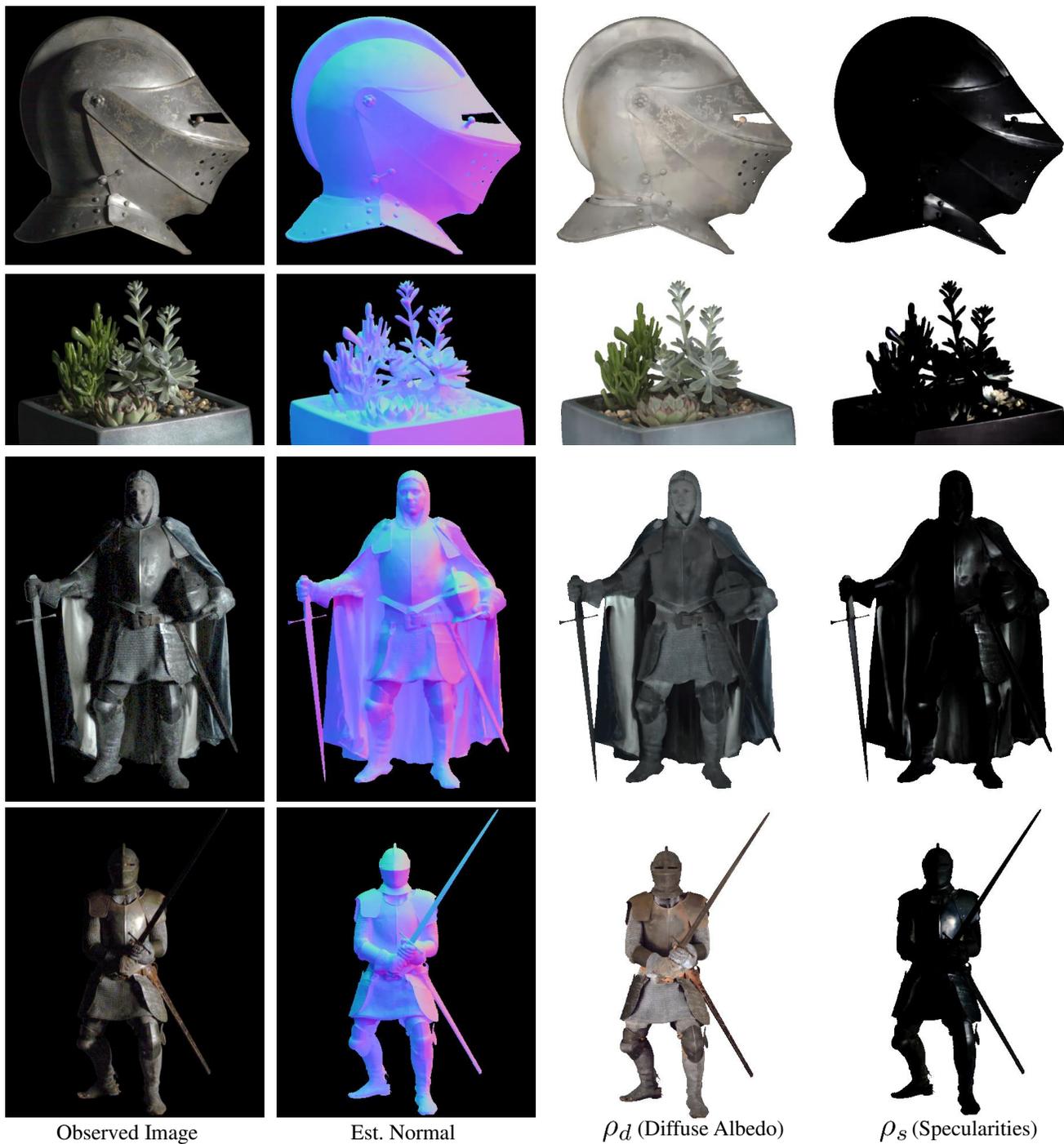


Figure 10. **Results on Light Stage Data Gallery** [3]. The columns from left to right are the observed images, our estimated normal, diffuse albedo, and specularities of the objects.