Supplementary Materials:
NEMTO: Neural Environment Matting for Novel View and Relighting Synthesis of Transparent Objects

Dongqing Wang   Tong Zhang   Sabine Süsstrunk
School of Computer and Communication Sciences, EPFL, Switzerland
{dongqing.wang, tong.zhang, sabine.susstrunk}@epfl.ch

A. Additional Implementation Details

A.1 Dataset details

On our synthetic datasets: kitty, bear, cow, and key mouse, we render images of resolution 512 × 512 with Mitsuba 3 [8], and we include details in the main paper, Sec. 4: Ln 499-509. We set the camera Field of View (FOV) to be 35°, and evenly sample 200 camera locations on the upper 120° sphere. We randomly select 100 for training and testing datasets respectively. An example of camera poses for the synthetic glass kitty dataset is shown in Fig. 1.

For our real-world datasets with rendered training data: dog, mouse, pig, and monkey which are released by TLG [9], each dataset contains 10 real-world captures for each object. Due to the lack of image data, we render 100 synthetic images with a resolution of 480 × 360 for training with the released ground truth mesh and environment illumination with the same camera setup as our synthetic datasets. We use the real-world captured images for testing and evaluations. Fig. 2 displays a set of example images and extracted masks for the synthetic and real-world datasets.

The object masks on each dataset are extracted with off-the-shelf background removal tool [6], and illumination estimation for each dataset is detailed in A.2.

In our real-world dataset (cat), we capture 136 images for training purposes, employing an iPhone 12 Pro Max. The camera poses are processed through the iOS application PolyCam [11] for further refinement. Furthermore, we capture the environmental illumination using the same iPhone and subsequently applied post-processing using the iOS application HDREye [4]. Notably, this experiment distinguishes our paper as the first in training on real-world natural light scenes for transparent objects.

A.2 Illumination Estimation For Synthetic Datasets

As acknowledged in Sec.3: Ln 251-265 of the main paper, we design an algorithm to pre-process input images to estimate the scene illumination as an environment map before training. The accuracy of our illumination estimation relies on the coverage of the environment from the sampling of camera poses and camera FOV, and the accuracy of estimated camera parameters.

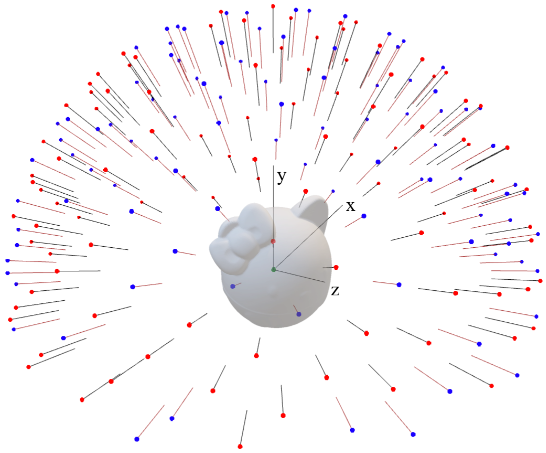
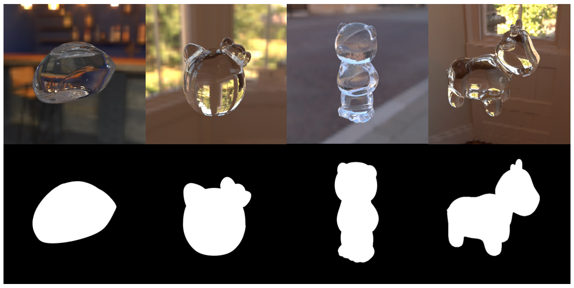
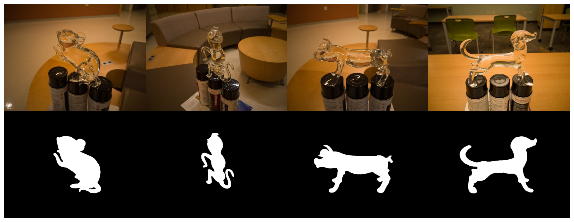


Figure 1. Camera poses for synthetic glass kitty dataset. Our sampled camera positions cover the upper 120° sphere. Blue points denote the positions of training cameras, and red points denote the positions of testing cameras. Line segments connected to the dots symbolize camera 'lookat' directions.

We assume the environment maps are of fixed resolution 250 × 500 for simplicity. Our estimation algorithm is shown as follows: assuming we have already obtained object masks for the current image set with [6], (1) for each pair of image $\mathbf{I}_i$ and mask $\mathbf{M}_i$, as well as the corresponding camera intrinsic matrix $\mathbf{K}$ and camera extrinsic matrix $\mathbf{P}_i^{\text{w2c}}$, we initialize per-pixel camera rays $\boldsymbol{\rho}(t) = \mathbf{o} + t\boldsymbol{\omega}_\mathbf{i}$ in world coordinates; (2) with each viewing ray direction, we use inverse texture mapping to obtain texture coordinates u and v from the environment map Γ for viewing direction $\boldsymbol{\omega}_\mathbf{i}$; (3) we mask out the object region of the image $\mathbf{I}_i[j,k] = 0$, if $\mathbf{M}_i[j,k] = 1$, leaving unrefracted background illumination, and map the image pixel color to cor-



(a) Synthetic Data



(b) Real world data

Figure 2. Examples of images and masks from synthetic and real-world datasets that NEMTO uses.

responding texture coordinates (u, v).

Our example reconstructed environment maps on the synthetic training datasets can be seen in Fig. 3. The algorithm cannot reconstruct the top region of each environment map, as our camera positions are concentrated on the upper hemisphere over the object. Therefore the combined viewing scope for all cameras in the dataset does not include the upper part of each environment map.

A.3 Training Details

In the early stage of training, we observe that joint optimization of ray refraction and geometry is difficult. Therefore, for the first 200 epochs, we set the weights for refraction guiding loss $\mathcal{L}_{\text{rg}}$, refraction smoothness loss $\mathcal{L}_{\text{rs}}$ and RGB pixel loss $\mathcal{L}_{\text{pix}}$ to zero, to initialize the geometry for the transparent object without entanglement with surface appearance. After a few epochs, the Geometry Network optimizes an SDF that is a rough estimate of the object geometry, we then set back $\lambda_{\text{rg}} = 100.0$, $\lambda_{\text{rs}} = 10.0$ and $\lambda_{\text{pix}} = 1.0$ as the rough shape is established. As mentioned in Sec. 3.5 Ln 180 - 190 of the main paper, we adopt a weight decaying strategy on $\mathcal{L}_{\text{rg}}$ to provide initial physically-guided supervision for Ray Bending Network on ray refractions. In the final stage of training, the weight λ_{rg} for refraction guiding loss is turned down, leaving pixel loss $\mathcal{L}_{\text{pix}}$ as the major loss term along with λ_{rs}.

The training time takes around 6 hours on a single RTX 3090 Ti with 2000 epochs for each scene.

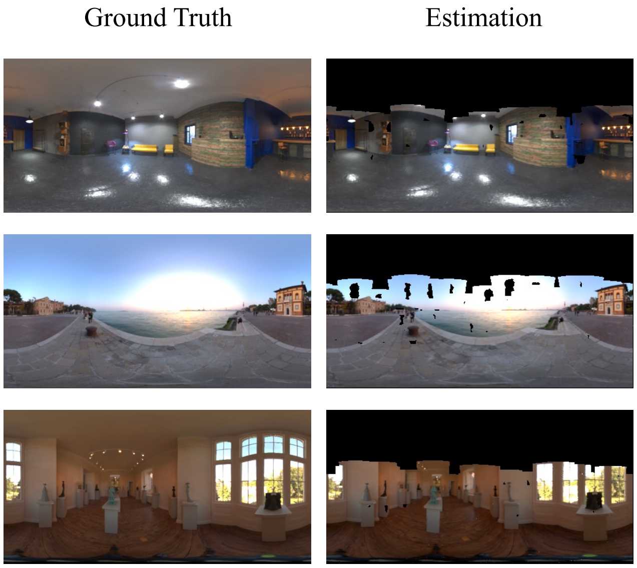


Figure 3. Examples of estimated environment illumination with our proposed data pre-processing. Our method highly depends on accurate camera estimation, and cannot extrapolate to unseen illuminations.

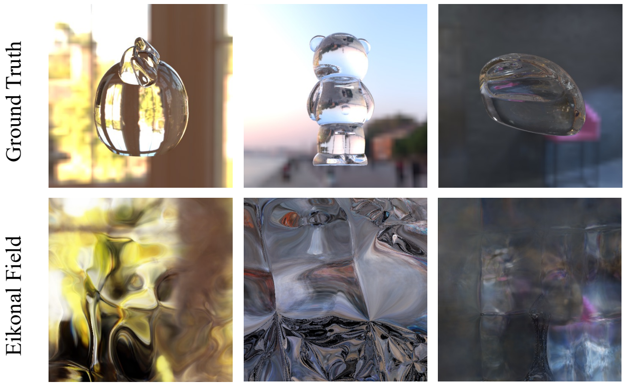


Figure 4. Examples of whole-image results of the Eikonal Field [3] on our synthetic datasets.

A.4 Baseline Evaluation Details

All of our baseline methods except for Eikonal Fields [3] are evaluated on ground truth images with their background masked out, as we want to emphasize the image synthesis quality on transparent objects.

Specifically, for the surface-based methods, i.e. IDR [14] and PhySG [15], we synthesize results with white backgrounds using optimized object masks during rendering. Therefore, we evaluate quantitative metrics by comparing model outputs with the masked ground truth. For volume-based methods with no specific object surface, we train and evaluate the model of NeRF [10] directly with masked ground truth for a fair comparison. However, Eikonal

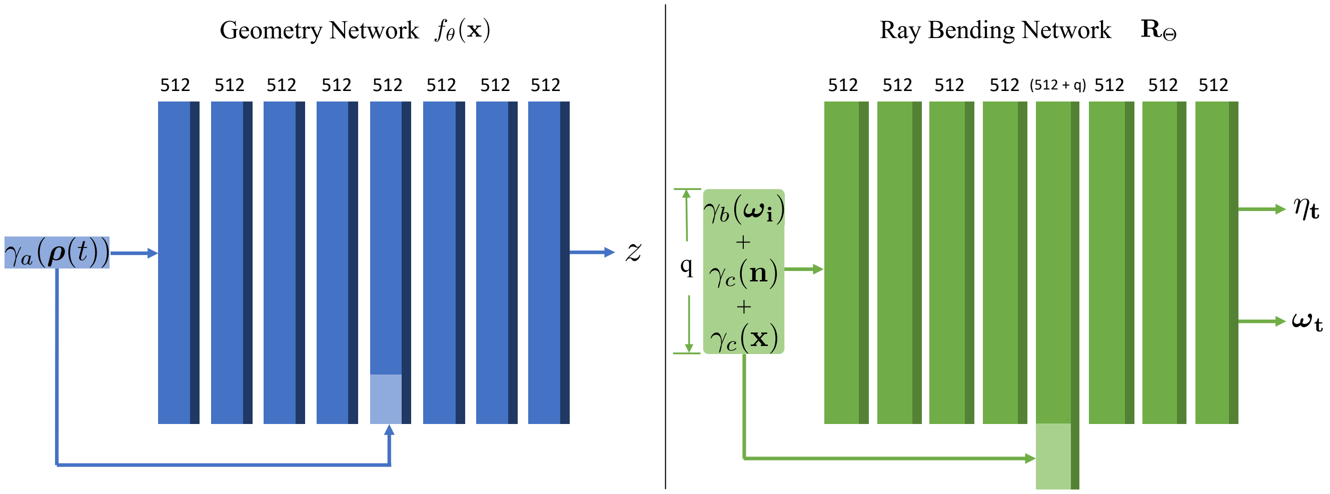

Geometry Network $f_\theta(\mathbf{x})$
512
512
512
512
512
512
512
512
$\gamma_a(\boldsymbol{\rho}(t))$
z
Ray Bending Network $\mathbf{R}_\Theta$
512
512
512
512
(512 + q)
512
512
512
$\gamma_b(\boldsymbol{\omega}_\mathbf{i})$
+
$\gamma_c(\mathbf{n})$
+
$\gamma_c(\mathbf{x})$
q
$\eta_\mathbf{t}$
$\boldsymbol{\omega}_\mathbf{t}$

Figure 5. **NEMTO Network Architecture.** We show the network layout of the two MLPs that we use to represent a transparent object. Our Geometry Network f_θ models object geometry through the zero-level set of an implicit signed distance function. Our Ray Bending Network $\mathbf{R}_\Theta$ models the refraction of each viewing ray through the transparent object.

Synthetic	↓Chamfer $L_1(10^{-3})$			
Method	Kitty	Bear	Key Mouse	Cow
SDF-A	10.24	8.73	7.28	6.89
NEMTO	**2.22**	**1.71**	**2.27**	**2.60**

Table 1. Quantitative evaluation on recovered meshes of synthetic datasets with SDF-A. We use Chamfer Distance as the metric for our evaluation[5].

RealWorld	↓Chamfer $L_1(10^{-3})$			
Method	Dog	Pig	Mouse	Monkey
TLG [9]	5.85	18.32	35.12	14.54
NEMTO	**2.65**	**3.14**	**6.18**	**9.22**

Table 2. Quantitative evaluation on recovered meshes of real-world datasets (trained with rendered data)with TLG [9]. We once again use Chamfer Distance as the metric for our evaluation[5].

Fields [3] fails to converge on our masked ground truth dataset and outputs all-zero values, as it cannot separate the opaque scene and the refractive part. Therefore, although Eikonal Field is also volume-based, we train and evaluate its model with the original unmasked images.

Fig. 4 shows a few examples of the full images of the Eikonal Fields model output, as Fig. 3 of our main paper only provides zoomed-in detail images for the model outputs. Again, NEMTO can work with real-world transparent objects and synthesize physically-plausible results.

A.5 Network Architecture

Our Geometry Network f_θ adopts the same MLP components as [14]. We use 8 layers with hidden width of 512, softplus activation: $x \mapsto \frac{1}{\beta}\ln(1+e^{\beta x})$ with $\beta = 100$, and a skip connection from input to the 4th layer. The network weights are initialized such that the initial SDF shape approximates a unit sphere as in [1].

The Ray Bending Network $\mathbf{R}_\Theta$ consists of 8 layers with layer width 512, followed by the activation function ELU. $\eta_\mathbf{t}$ is randomly initialized from [1.2, 3.0]. The input ray direction $\mathbf{r}$, surface normal $\mathbf{n}$ and coordinate $\mathbf{x}$ are transformed by Fourier output $\gamma(\cdot)$ separately with different numbers of bands for learning high-frequency information [12]. A skip connection is again used to concatenate input to the 4th layer. We set $\gamma_a = 3$, $\gamma_b = 8$, $\gamma_c = 4$, and we use the $\tanh$ function at the end for a valid output.

The network architecture of NEMTO is shown in Fig. 5.

B. Additional Results

B.1 Ablation on SDF-A

As mentioned in Sec. 4: Ln 697 - 701 and Ln 737 - 744 of the main paper, SDF-A is a naive version of NEMTO without using the Ray Bending Network. We demonstrate the advantage of our synthesis using Ray Bending Network over the analytical refraction of SDF-A by Fig. 6 and Tab. 1. We can see that SDF-A optimizes a slightly more inaccurate object geometry than ours, but the former's synthesized images due to analytical refraction are much less faithful to the ground truth than those of our Ray Bending Network.

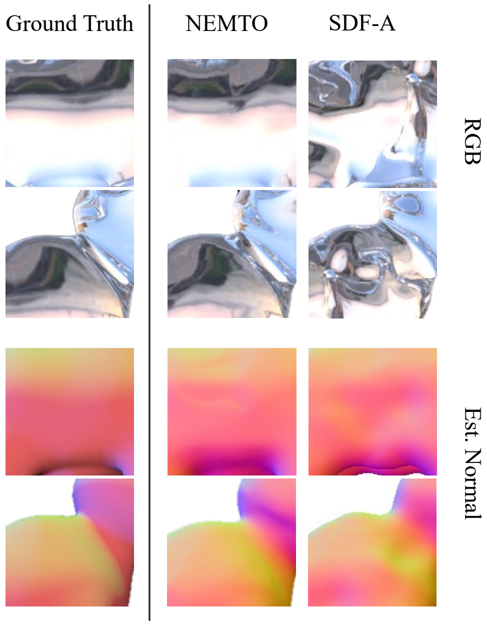


Figure 6. Qualitative comparison of the estimated surface normals and synthesized images between NEMTO and SDF-A on the synthetic glass cow dataset. Note the inaccurate refraction caused by analytical refraction in SDF-A.

	Novel View			Relighting		
Method	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
M-NeRF	23.59	0.86	0.16	-	-	-
NVDiffRec	20.45	0.79	0.21	14.92	0.69	0.24
NEMTO	**28.86**	**0.94**	**0.07**	**27.24**	**0.92**	**0.06**

Table 3. Quantitative comparisons with Multi-NeRF (containing Ref-NeRF [13] and Mip-NeRF360 [2]) and NVDiffRec [7]. Multi-NeRF is not designed for relighting, therefore, we omit the relighting evaluation to Multi-NeRF.

B.2 Synthetic Datasets Results

We show additional novel view synthesis results on all of our synthetic datasets in Fig. 8 and Fig. 9. The synthesized novel views are physically-plausible and faithful to the ground truth reference renderings. NEMTO is able to model complex light reflection and refraction, even in challenging cases such as the synthetic glass cow.

We also provide quantitative comparisons with several more recent works [2, 7, 13] in Tab. 3. As explained in the main paper, these works do not consider light transmission for novel view synthesis. Those allowing relighting use Disney BRDF as the material decomposition model, which is not suitable for relighting transparent objects. Consequently, their results on novel view and relighting synthesis are expected to be similar, if not slightly better, than those shown in the main paper. We provide the comparative results as evidence supporting this point.

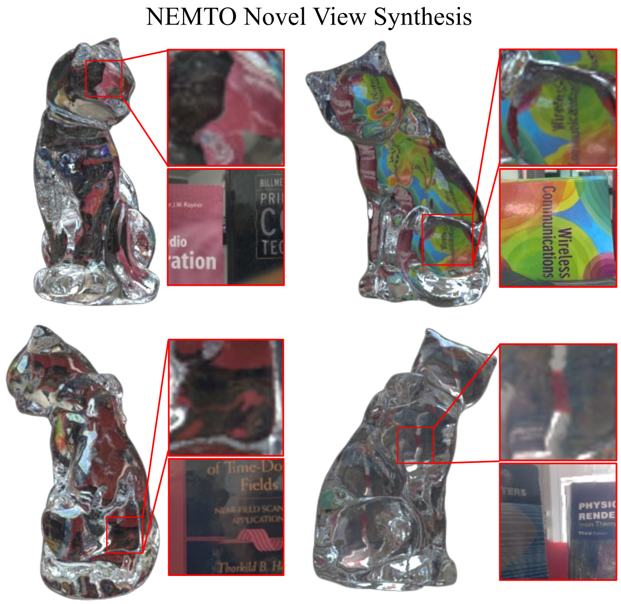


Figure 7. Quantitative results on novel view synthesis results from NEMTO trained on real-world captured data.

B.3 Real World Datasets with Captured Training Data

We show additional novel view synthesis results on our captured cat dataset in Fig. 7.

B.4 Real World Datasets with Rendered Training Data

Geometry estimation. Fig. 10 and Tab. 2 show the quality of estimated geometry for the real-world transparent objects from NEMTO and TLG [9]. NEMTO archives finer details on object surfaces, such as the shape of the dog's mouth and the pig's ears, as well as the eye socket section of the mouse. This is due to NEMTO's dense sampling of input viewpoints compared to TLG. However, as NEMTO relies heavily on silhouette loss for geometry regulation, it cannot accurately model certain detailed concave sections of the object, such as the left arm of the mouse. Still, NEMTO achieves an overall better estimation of geometry than TLG shown in Tab. 2, and we do not require a very large synthetic dataset with 1.5k HDR environment maps for network training as needed by TLG.

Novel view and relighting synthesis. In fig. 11, 12, and 13, we show a more comprehensive collection of novel views and relighting synthesis on our real-world datasets. The quality of our synthesis is dependent on estimated environment illumination, which can be inaccurate in the case of real-world datasets. Despite such inaccuracy, we generate high-quality novel views and relighting for real-world transparent objects. The relighting synthesis is produced with the environment map in the first row of Fig. 3.

References

[1] Matan Atzmon and Yaron Lipman. Sal: Sign agnostic learning of shapes from raw data. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2565–2574, 2020. 3

[2] Jonathan T. Barron, Ben Mildenhall, Dor Verbin, Pratul P. Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. *CVPR*, 2022. 4

[3] Mojtaba Bemana, Karol Myszkowski, Jeppe Revall Frisvad, Hans-Peter Seidel, and Tobias Ritschel. Eikonal fields for refractive novel-view synthesis. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference Proceedings*, pages 1–9, 2022. 2, 3

[4] The Young Astronauts Corp. Hdreye, May 2023. 1

[5] Christian Diller. Chamfer distance for pytorch, 2022. https://github.com/otaheri/chamfer_distance. 3

[6] Kaleido AI GmbH. Remove image background., 2022. https://www.remove.bg/. Accessed 2022-11. 1

[7] Jon Hasselgren, Nikolai Hofmann, and Jacob Munkberg. Shape, Light, and Material Decomposition from Images using Monte Carlo Rendering and Denoising. *arXiv:2206.03380*, 2022. 4

[8] Wenzel Jakob, Sébastien Speierer, Nicolas Roussel, Merlin Nimier-David, Delio Vicini, Tizian Zeltner, Baptiste Nicolet, Miguel Crespo, Vincent Leroy, and Ziyi Zhang. Mitsuba 3 renderer, 2022. https://mitsuba-renderer.org. 1

[9] Zhengqin Li, Yu-Ying Yeh, and Manmohan Chandraker. Through the looking glass: Neural 3d reconstruction of transparent shapes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1262–1271, 2020. 1, 3, 4, 8

[10] 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 *Computer Vision–ECCV 2020: 15th European Conference, 23-28 August 2020, Proceeding*, 2020. 2

[11] Polycam.Inc. YOLO by Ultralytics, May 2020. 1

[12] Matthew Tancik, Pratul Srinivasan, Ben Mildenhall, Sara Fridovich-Keil, Nithin Raghavan, Utkarsh Singhal, Ravi Ramamoorthi, Jonathan Barron, and Ren Ng. Fourier features let networks learn high frequency functions in low dimensional domains. *Advances in Neural Information Processing Systems*, 33:7537–7547, 2020. 3

[13] Dor Verbin, Peter Hedman, Ben Mildenhall, Todd Zickler, Jonathan T Barron, and Pratul P Srinivasan. Ref-nerf: Structured view-dependent appearance for neural radiance fields. In *2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5481–5490. IEEE, 2022. 4

[14] Lior Yariv, Yoni Kasten, Dror Moran, Meirav Galun, Matan Atzmon, Ronen Basri, and Yaron Lipman. Multiview neural surface reconstruction with implicit lighting and material. *Adv. Neural Inform. Process. Syst*, 3, 2020. 2, 3

[15] Kai Zhang, Fujun Luan, Qianqian Wang, Kavita Bala, and Noah Snavely. Physg: Inverse rendering with spherical gaussians for physics-based material editing and relighting. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5453–5462, 2021. 2

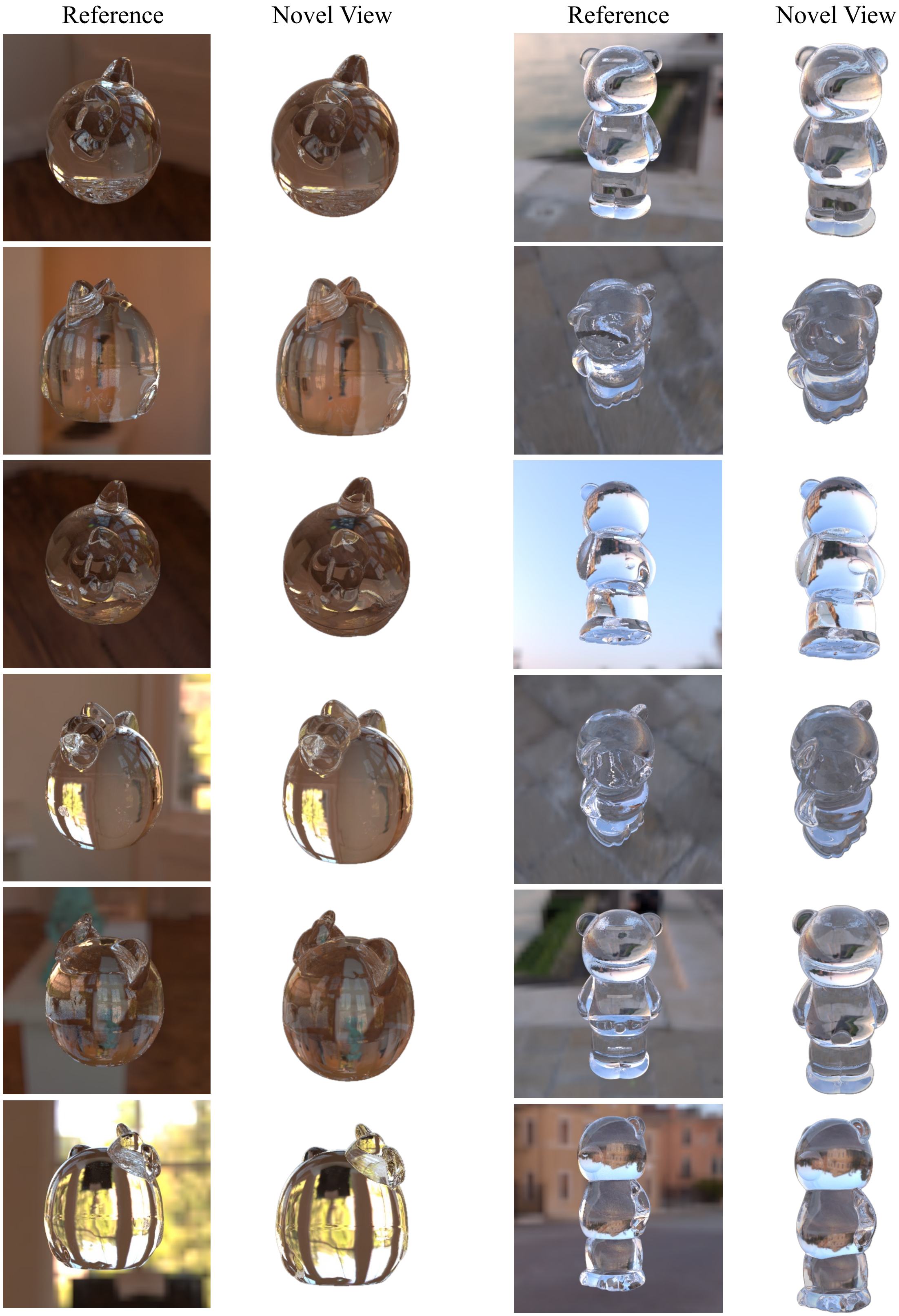


Figure 8. Novel view synthesis on the synthetic glass kitty and glass bear. Our synthesized images are physically-plausible and faithful to reference images.

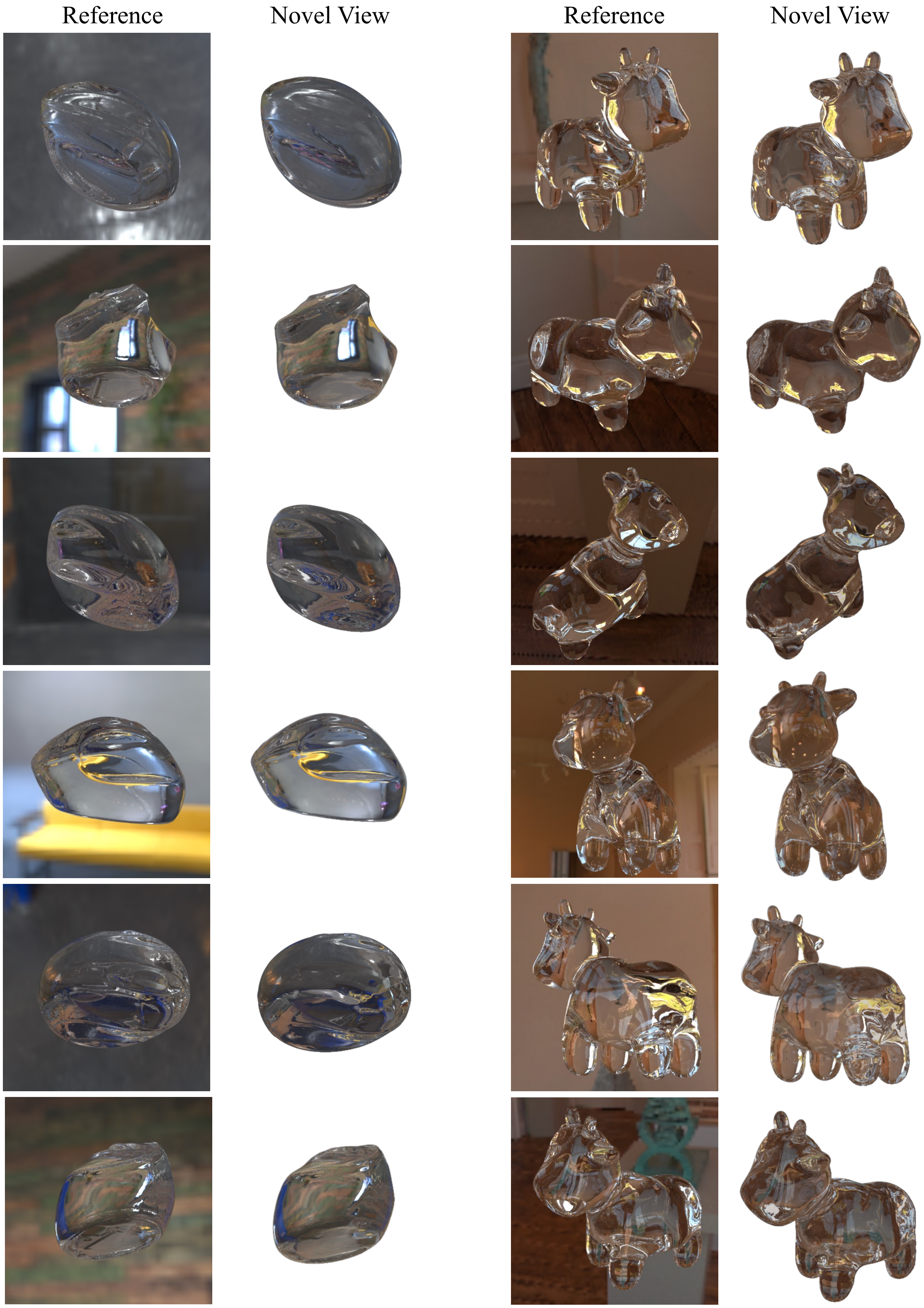


Figure 9. Novel view synthesis on the synthetic Key Mouse and Cow.

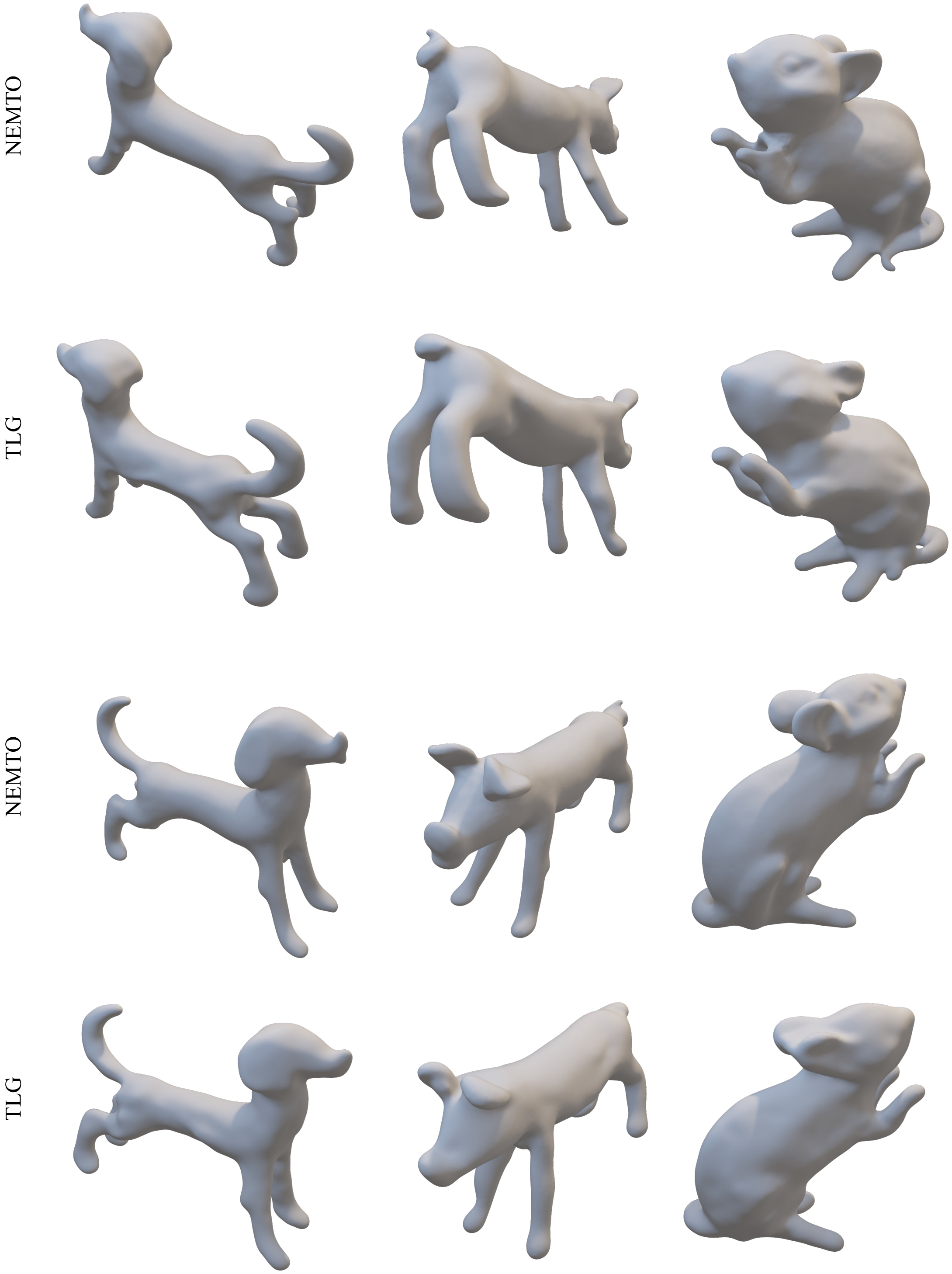


Figure 10. Estimated geometry comparison on real-world datasets between NEMTO and TLG [9]. NEMTO estimates geometry with more refined details such as the ear section of the pig model and the mouse section of the dog model.

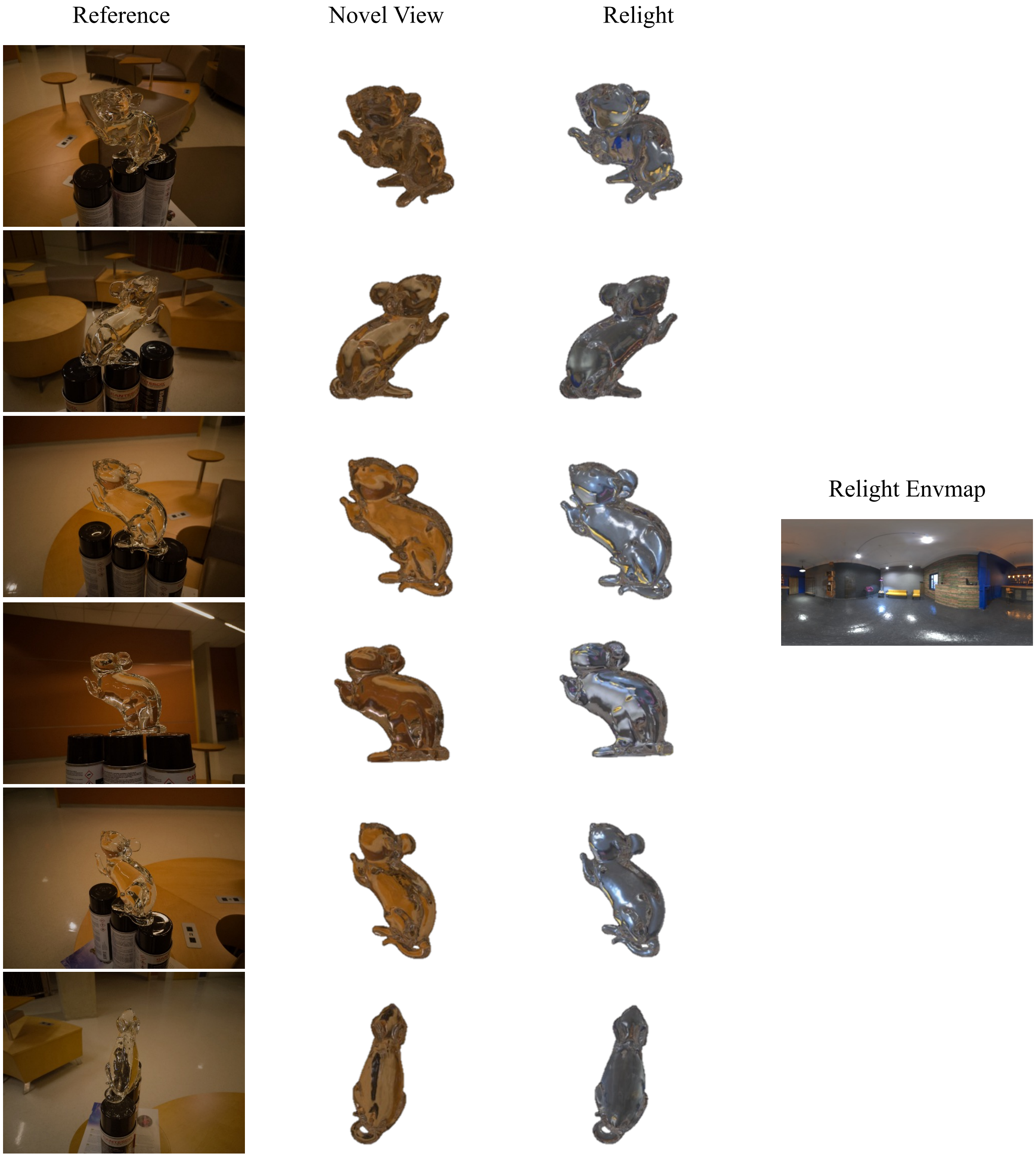


Figure 11. Novel view and relighting synthesis on the real-world mouse dataset. Results are best viewed on a tablet or computer screen. The ray bending network of NEMTO gives physically-plausible results to refract rays through the body of the object. Some inaccuracies persist, partially due to the error in estimating real-world environment illumination.

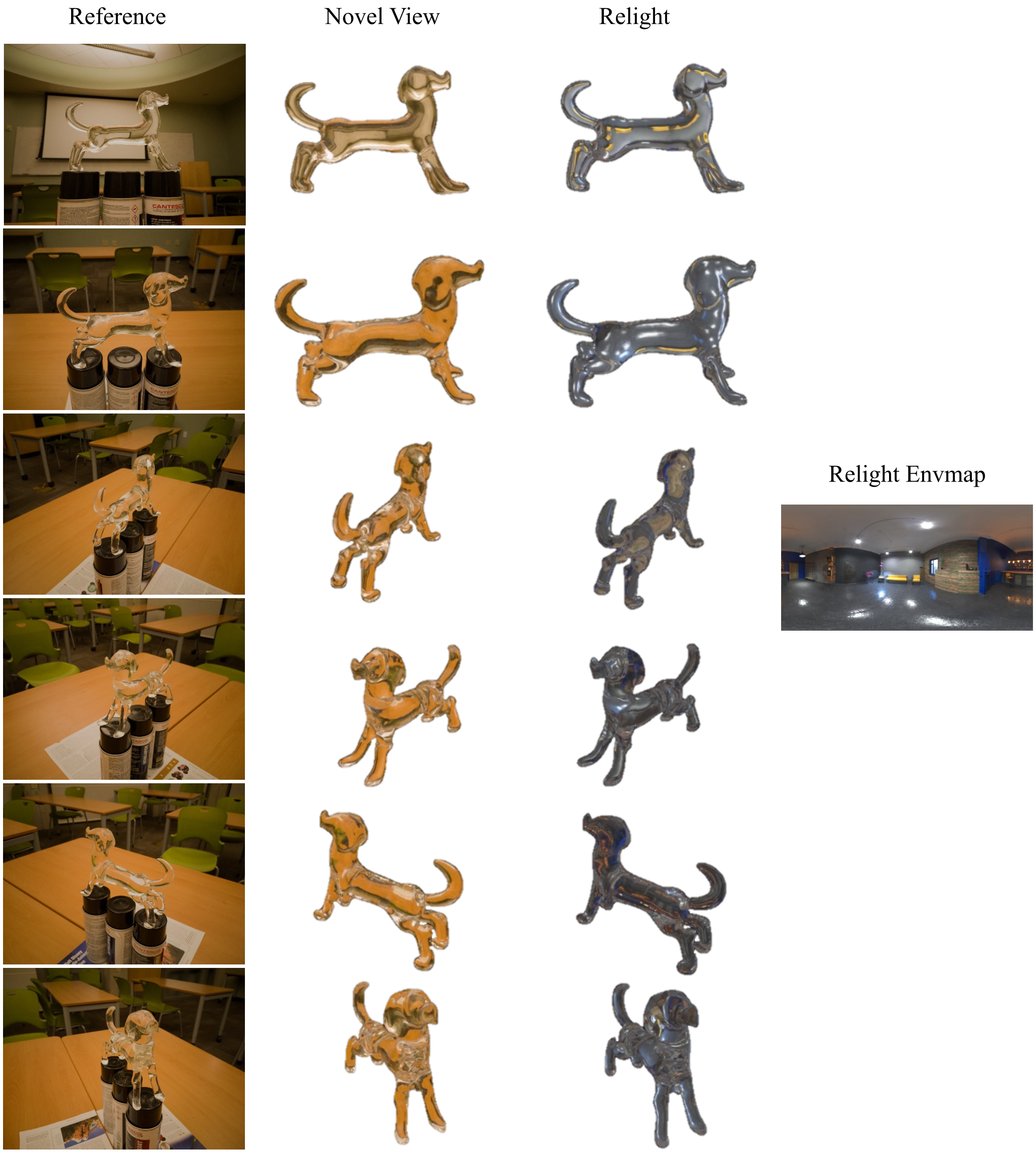


Figure 12. Novel view and relighting synthesis on the real-world dog dataset. Results are best viewed on a tablet or computer screen.

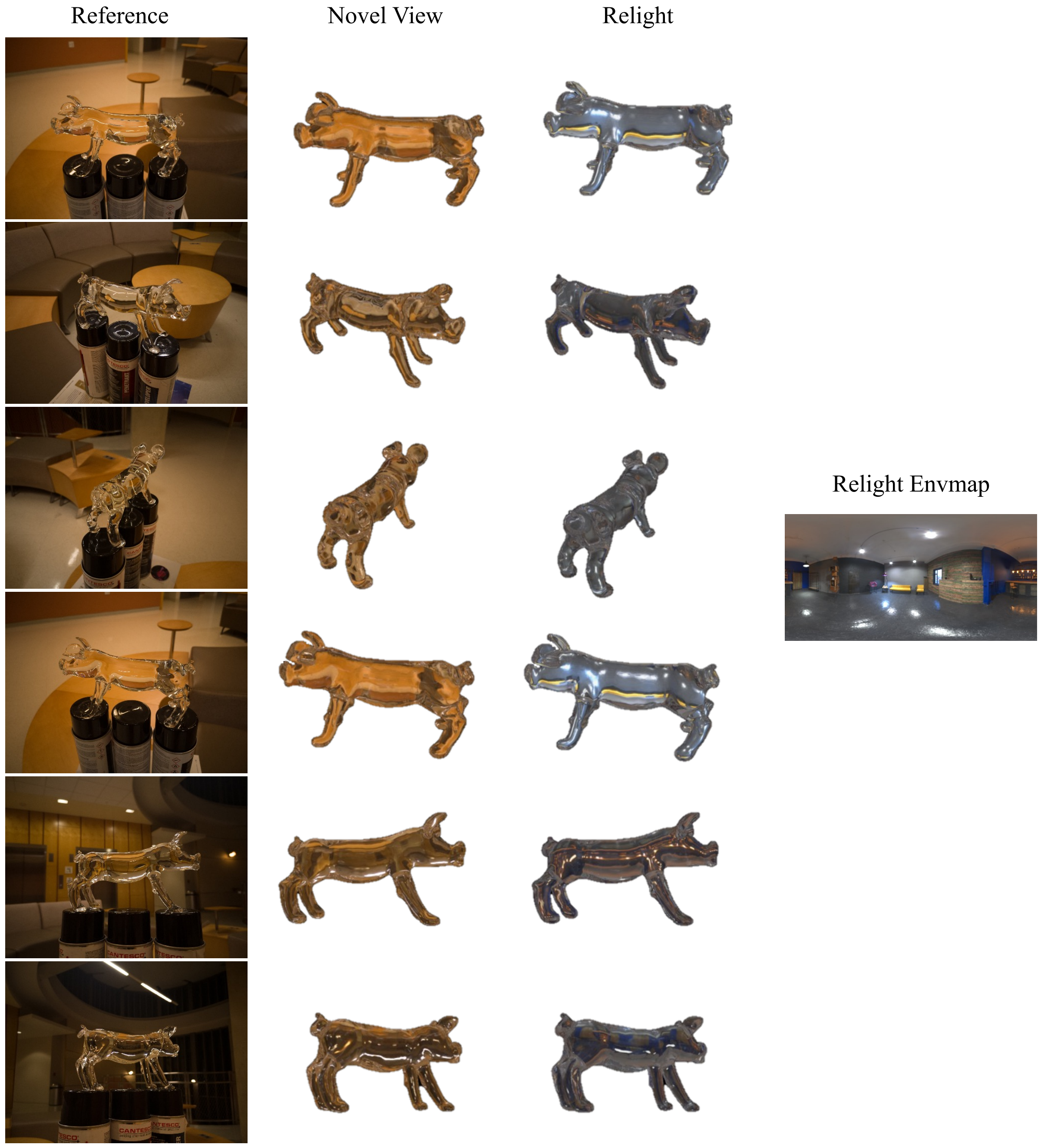


Figure 13. Novel view and relighting synthesis on the real-world pig dataset. Results are best viewed on a tablet or computer screen.