Neural Reflectance Fields for Appearance Acquisition ++

Liao Wang   Ziyu Wang   Fuqiang Zhao   Xinyi Ye

Shanghaitech University

393 Middle Huaxia Road, Pudong, Shanghai, China

{wangla,wangzy6,zhaofq,yexy2}@shanghaitech.edu.cn

Abstract

We reproduce neural reflectance fields which utilize a deep fully-connected neural network to enconde volume density, normal and reflectance properties of the scene.Neural reflectance fields are combined with a physically-based differentiable ray marching framework that enables a complete pipeline from high-quality and practical appearance acquisition to 3D scene composition and rendering. Since neural reflectance fields are demonstrated that they can be estimated from images captured with a simple collocated camera-light setup, and accurately model the appearance of real-world scenes with complex geometry and reflectance, we reproduce the neural reflectance fields and make some improvements. We introduce three contributions to enhance the performance of the neural reflectance fields, including a depth normal consistent regularization term, a solution to non-collocated light settings and a unqiue delta network. We demonstrate that our improvements can achieve a better performance and more accurately model the appearance of objects than original neural reflectance fields.

1. Introduction

Synthesizing and relighting novel views of a scene from captured images is a long-standing problem in computer graphics and vision. Classic techniques have addressed this using 3D reconstruction and inverse rendering methods to recover scene geometry and reflectance. Recently, learning-based methods have led to significant breakthrough towards accurate reconstructions, which firstly model a "neural" scene representation and then apply an appropriate volume rendering method to synthesize new views. The Neural Radiance Fields (NeRF) [18] approach has shown that it is possible to encode the geometry and appearance as volumetric density and color functions parameterized by 3D coordinates of query points in the scene.

Although NeRF produces outstanding results for view synthesis, it does not provide a solution for relighting. Thus,

a novel scene representation, Neural Reflectance Fields, is proposed, which accounts for both scene geometry and reflectance. Neural Reflectance Fields is an implicit MLP-based model that maps (4 + m)-dimensional vectors – comprising volume density (1-D), normal (3-D) and reflectance properties (m-D) – at any 3-D position in a scene. This representation can be combined with a differentiable ray marching framework – based on classical physically based volume rendering. Compared with NeRF, this method utilizes the transmittance not only along the camera ray but also along the light ray to model light effects like shadows for complex real scenes.

However, some potential weaknesses still exist in Neural Reflectance Fields. The first insufficiency is that the predicted normal has a deviation from the actual geometry. In the original Neural Reflectance Fields, it is usually inconsistent to fit the real geometry after the predicted normal being used into BRDF mode directly. Thus, we add a depth normal consistent loss to solve the ambiguity. The second deficiency is that the origin method cannot handle the situation when camera and light is not collocated. We put forward a shadow map which combines traditional graphics pipeline with Neural Reflectance Field to enable non-collocated light setting training. While rendering, the shadow map is used as a postprocessing to cast the shadow on the novel view. The last insufficiency is that captured image do not strictly obey the selected BRDF model. The solution we proposed is using a delta network to make up the gap between captured image and the BRDF model. The delta network takes the same input as the microfacet BRDF model from the Neural Reflectance Field MLP. Thus, this will generate a refined color comprised by the initial color and the delta color. It is worth noting that the loss will calculate with the initial color and refined color respectively to avoid the overfitting of the delta network.

In summary, we apply Neural Reflectance Fields to model both scene geometry and reflectance, render it under any view and lighting, and make some improvements to solve its insufficiencies. We demonstrate that our improved Neural Reflectance Fields method quantitatively and quali-

tatively outperforms the origin version.

2. Related Work

Neural scene representations. Recently, neural scene representations have attracted considerable attention. Previous work has applied deep neural networks to many 3D tasks with scene geometry modeled by various representations, such as volumes [11], point clouds [22], implicit functions [24], etc. Because neural representations are inherently 3D, they enable novel view synthesis. The proposed Neural Reflectance Fields models both geometry and reflectance in a real scene.

Many previous works aim to do view synthesis without any known geometry. However most of them only supports limited viewing range. Recent works leverage view-independent volumes, which are able to handle complex view-dependent effects [15]. The proposed Neural Reflectance Fields can be used for applications such as modeling complete scene appearance and relighting.

Recently, ray marching has been applied to train many neural scene representations for view systhesis without any ground-truth 3D representations. Lombardi et al. [15] apply ray marching in a discrete volume with a warping field for view synthesis. However, the fixed resolution of the discrete volume limits the appearance details in the rendering. To address this defect, the Neural Reflectance Fields applies a continuous functional neural representation and achieves much better results. Compared with the origin NeRF [18], the Neural Reflectance Fields leverage a novel reflectance-aware ray marching framework and learn to regress multiple decomposed shading components.

Geometry and reflectance capture. Classically, modeling and rendering a real scene requires full reconstruction of its geometry and reflectance. From captured images, scene geometry is usually reconstructed by structure-from-motion and multi-view stereo (MVS) [9, 7, 13, 23], which have recently been extended using deep learning techniques [3, 4, 29].

Reflectance acquisition traditionally requires sophisticated devices to sample the light-view space. [6, 12, 16, 20, 28] Recently, many works use a practical device – a modern cellphone that has a camera and a built-in flash light – and capture flash images to acquire spatially varying BRDFs [2, 10, 19]. More recently, deep learning methods have made BRDF acquisition with a single flash image possible. It is very difficult for traditional mesh-based methods to recover challenging thin structures and sharp specularities of complex real scenes using multi-view unstructured flash images. The Neural Reflectance Fields implicitly models the scene's geometry and reflectance, bypassing explicit mesh reconstruction.

Relighting and view synthesis. Scene acquisition and ren-

dering can be also achieved using image-based techniques without explicit reconstruction [5, 14]. Recently, many learning based view synthesis methods have been presented [8, 18, 26, 27, 31]. In Neural Reflectance Fields, reserchers extend the ray marching in the view synthesis works to a more general reflectance-aware ray marching framework, which can also be used to do relighting. Learning-based relighting methods have also been presented, which are able to reproduce challenging appearance effects. Neural Reflectance Fields's ray marching considers light transmittance in ray integration, which recovers challenging hard shadows.

3. Method

We reproduce the Neural Reflectance Fields by ourselves. Neural Reflectance Fields allow to render high fidelity novel views in arbitrary point light illumination conditions. Also, it estimates the BRDF parameters of the object. So, another effect is that it can segment the material. To make up for the insufficient of the original Neural Reflectance Fields, we propose depth normal consistent loss, shadow map and delta network module to improve the result and extend Neural Reflectance Fields to non-collocated light settings.

3.1. Background: Neural Reflectance Fields

Neural Reflectance Fields (NeRF AA) utilizes classical reflectance models in ray marching framework which not only models lighting and also enables relighting and re-rendering applications. In this section, we will review underlying rendering formulation and ray marching framework of Neural Reflectance Fields.

Rendering equation. In the Neural Reflectance Fields, we compute the radiance $L_s(o, d)$ at a point o in spatial space in direction d based on volume rendering technique[21] :

$$L(o, d) = \int_0^\infty \tau_o(x)\sigma(x)L_s(x, d)dt, \quad (1)$$

where $\tau_o(x) = e^{-\int_0^t \sigma(o+ud)du}$, the integration factor t represents depth in direction d from the ray origination and $x(t) = o + td$ is the 3D location at t. o is the ray origination or camera position. σ is volume density and $\tau_o(x)$ represents opacity at the point x from camera location o alone the ray.

In the Eqn. 1, $L_s(x, d)$ represents the radiance arrives at position x in direction d, and the radiance at o is computed by integrating the modulated in-scattered light L_s along the ray:

$$L_s(x, d) = \int_S f_p(x, d, d_i)L_i(x, d_i)dd_i, \quad (2)$$

where S is a unit sphere, f_p is phase function that governs the light scattering.

$$\sum_j \tau_c(\pmb{x}_j)\,\tau_l(\pmb{x}_j)\,\left(1-\exp(-\sigma(\pmb{x}_j)\,\Delta t_j)\right) f_r(\pmb{x}_j)\,L_l(\pmb{x}_j)$$
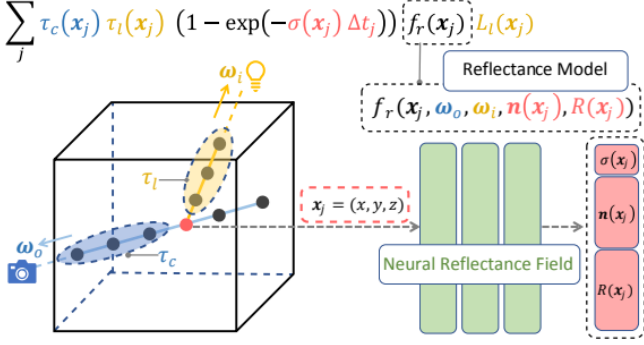

Figure 1. Overview of the neural reflectance field and ray marching operation. Blue rays are marched through each pixel from the camera and sample a sequence of shading points on the ray. Yellow rays are marched through each sample points on blue rays and point light source. In the right part, We propose to use a novel neural reflectance field, represented by an MLP, to regress the required rendering properties(σ, n, R) from the 3D location $x_j = (x, y, z)$ for each ray marching.

Considering single-bounce direct illumination under a single point light source to approximate L_s. We compute L_s in an explicit reflectance term which is an assumption of the phase function inspired by[17]:

$$L_s(x, d) = f_r(x, d, d_i, n(x), R(x))L_i(x, d_i), \quad (3)$$

where R is the parameters of the differentiable reflectance model f_r, n is the local surface shading normal. L_i is the incident radiance as in Eqn. 2. If in a point light source settings, L_i will be formulated as:

$$L_i(x, d_i) = \tau_l L_l(x), \quad (4)$$

where τ_l is the transmittance from the light to the shading point. L_l represents the light intensity with the consideration of distance attenuation. l denotes the light source.

Therefore, combining the Eqn. 1,3, 4, the final volume rendering equation is:

$$L(o, d) = \sum_{j=0}^{N} \tau_o(x_j)\tau_l(x_j)T_i f_r(x_j)L_l(x_j),$$
$$where\ T_j = (1 - exp(-\sigma(x_j)\delta(t_j))) \quad (5)$$

Ray marching. In Eqn.5,$\tau_o(x_j)$ is an integral and can be numerically evaluated by:

$$\tau_o(x_j) = exp(-\sum_{k=0}^{j} \sigma(x_k)\delta(x_k)). \quad (6)$$

The transmittance $\tau_l(x_j)$ can be similarly evaluated, but it requires a second sampling operation to obtain another sequence of sampling points x'_p on an additional ray marched
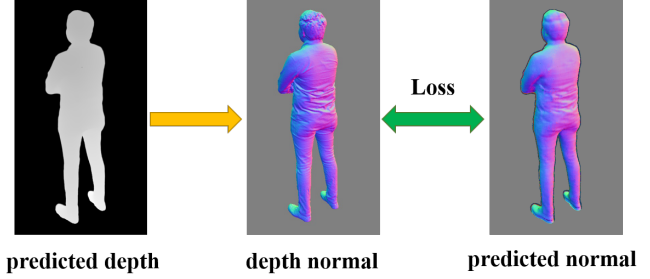

Figure 2. Visualization for the calculation of depth normal consistent loss.

from the point light source to the light source to the shading point x_j:

$$\tau_l(x_j) = exp(-\sum_{p} \sigma(x'_p)\delta(x'_p)). \quad (7)$$

Naively computing Eqn. 7 for Eqn. 5 would require marching a large number of light rays for all shading points on all camera rays which is an very ineffective operation. Instead, Neural Reflectance Fields leverage a collocated light source and camera setup (where the camera and light rays are the same) to avoid this during training.

3.2. Improvements of Neural Reflectance Fields

Next, we introduce 3 schemes to enhance the performance of the original Neural Reflectance Fields algorithm. The improved Neural Reflectance Fields can generate higher quality appearance and geometry in both collocated and non-collocated settings.

Depth normal consistent loss. In the original Neural Reflectance Fields, the normal of the object is predicted directly from the output of a 3 channel output of a MLP network. Then, the predicted normal is used directly in following BRDF model to calculate the view-dependent and light-dependent color. Although the direct output normal can be learned by back propagation, it is usually inconsistent to the actual geometry. Some reflection effects can be optimized by altering the normal instead of estimating the BRDF correctly. As a result, the predicted normal can be over smooth in some area or flickering in others.

As the Fig. 2 shows, we want to solve the ambiguity by coupling the surface normal with the actual shape. During the training, we randomly select a view on the sphere around the object and the its predicted depth map and normal map at this view. For the depth map, given the sample points along a pixel ray $\mathbf{r}(t) = \mathbf{o} + t\mathbf{d}$, we have the parameters $t_i, i \in \{1, ..., n\}$ for the n sampled points. Then, the

depth $D(r)$ is formulated as:

$$D(\mathbf{r}) = \sum_{i=1}^{n} W_i(1 - exp(-h(\sigma_i)\delta_i))t_i,$$

$$W_i = exp(-\sum_{j=1}^{i-1} h(\sigma_j)\delta_j), \tag{8}$$

where $\delta_i = t_{i+1} - t_i$ is the distance between adjacent samples. $h(\cdot)$ is a piece-wise function as follows:

$$h(x) = \begin{cases} 0 & x \le \tau \\ x & otherwise \end{cases} \tag{9}$$

where τ is a threshold. Note that $h(\cdot)$ filters those noises with low density in order to obtain more accurate depth values. Then, we can calculate the predicted intersection of surface with each ray using the depth map by $\mathbf{p} = \mathbf{o} + D(r)\mathbf{d}$. For each ray, we calculate its normal from cross product and finally normalize it. Its formulation is as:

$$\mathbf{N} = \frac{(\mathbf{p}_{i,j} - \mathbf{p}_{i,j-1}) \times (\mathbf{p}_{i,j} - \mathbf{p}_{i-1,j})}{\|(\mathbf{p}_{i,j} - \mathbf{p}_{i,j-1}) \times (\mathbf{p}_{i,j} - \mathbf{p}_{i-1,j})\|} \tag{10}$$

where $\mathbf{p}_{i,j}$ is the corresponding intersection point of the ray corresponding to pixel at index i, j of the depth map.

The reason why we use the depth to calculate instead of using the density field is that the depth map we calculated is smoother and has less noise. We use the MSE loss to calculate the difference between the normal from depth and the predicted normal:

$$L_{normal} = \sum_{q} \left\| N^q - \tilde{N}^q \right\|^2 \tag{11}$$

where q denotes a pixel ray, N is the normal from depth and $\tilde{N}$ is the predicted normal.

Non-collocated light settings. Origin Neural Reflectance Fields can only handle the collocated settings during the training. In Fig. 3(a), we need to additionally query the MLP for volume density at the sample points between the light location and every sample point along the camera ray in naive sampling. It is used to determine whether the light is occluded by objects. However, it is prohibitively expensive compared to only sampling along the camera ray during the training. Hence, we propose to use a shadow map that combines traditional graphics pipeline with Neural Reflectance Fields to enable non-collocated light setting training.

As illustrated in Fig. 3(b), for non-collocated settings, we avoid to sample the points between the light source and camera ray sampling points to save computational complexity. Instead, we use a shadow map to deal with shadow. During the training, we mask out the rays that belong to
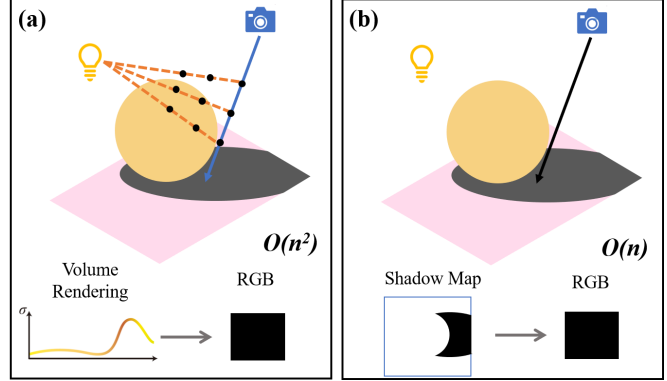

Figure 3. We visualize how we relax the computation burden in non-collocated light settings.

the shadows, which means we only use the rays that not in shadow for training. The light transmittance τ_l is set to 1. During the rendering, we will cast the shadow on the novel view using the shadow map as a postprocessing.

Delta network. Although the most versatile microfacet BRDF model is used, it is still an approximation of real world light model. The captured image do not strictly obey the selected BRDF model. Also, indirect lighting is not considered in our model, because for example as Fig. 4(b) [25] shows, the computation complexity of one-bounce indirect illumination during the naive raymarching process is $\mathcal{O}\left(n^3 d\ell\right)$ which is not affordable during the training. Then, the predicted normal or roughness may suffer from artifacts because even the microfacet BRDF model is lack of representation ability and the network will finally overfit the RGB loss by altering the normal or roughness. Therefore, we propose a delta network module to make up the gap between the captured images and the selected BRDF model.

The delta network module is as the grey area shown in Fig. 4(a). It takes the same input as the microfacet BRDF model: the normal, roughness, albedo color that estimated from the Neural Relfectance Field MLP and view direction, light direction for each sampling point. All the inputs are applied with the frequency-based positional encoding. The delta network is also a MLP network, and it outputs a 3 channel delta RGB color δC. The initial color for each sample point calculated from the microfacet BRDF model is added to the delta color to get the refined color C'. The refined color is used the same volume rendering procedure to calculate the final color.

It is worth noting that the loss will calculated with accumulated color both with and without delta network module. It is because we do not want that the delta network is overfitting. We still expect that the color achieved from microfacet BRDF is close to capture image, such that the estimated Neural Reflectance Field is regularized. In this way, the delta network module can do its real job.
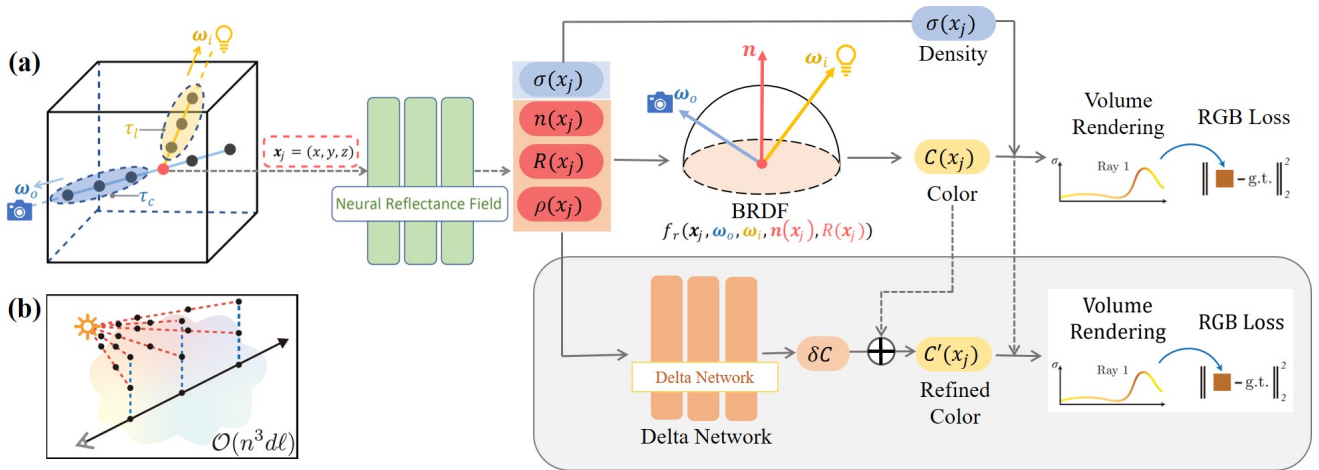
4

Figure 4. (a) The full pipeline after with delta network module. (b) Visualization of the one-bounce indirect lighting raymarching process (n is the number of samples along each ray, l is the number of light sources, and d is the number of sampled indirect illumination directions). Black dots represent evaluating a Neural Reflectance Fields MLP for volume density at a position.

3.3. Implementation Details

Data acquisition. We capture data by ourselves in both collocated settings and non collocated settings.

For collocated settings, we follow the origin Neural Reflectance Fields by capturing several frames using a handheld cellphone with flash. Instead of using a robotic arm holding the cellphone, we place the object on a rotating platform with marks. We capture around 120 images for objects and around 70 images for human portraits using a iPhone 7 Plus. The camera parameters are calibrated using structure from motion by software AGI [1].

For non-collocated settings, the data are provided by MARS lab which are captured in a light doom with calibrated camera positions and light positions. Around 600 images are used in training process.

Similar to the origin Neural Reflectance Field, the central regions around the objects are cropped in the captured images since the background scene is not rotating with the object. It can be done by training the origin NeRF in just a few epoches quickly (we use 4 epoches) to get a rough geometry using marching cube and project it to captured view to crop the object. Each network is trained in a scene dependent way, using the input images for that single scene.

Training parameters. We reproduce the Neural Reflectance field in the PyTorch framework. We randomly select 4000 rays as a batch during the training. We use Adam optimizer with an initial learning rate of 0.0001. For volumetric integration, we use 64 samples in coarse volume and 128 additional adaptive samples in fine volume.

Our Neural Reflectance Field network structure is illustrated in Fig. 5. It has a main backbone and four output branch. The main backbone has 7 layers, 256 hidden neurons and ReLU activation with skip connection at the fifth
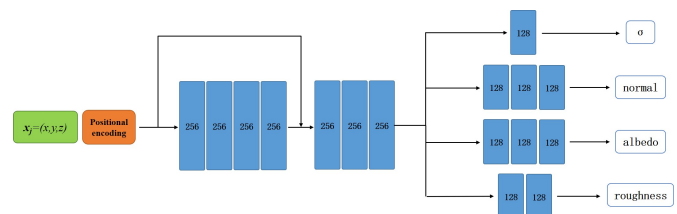

Figure 5. Visualization of our Neural Reflectance Fields network architecture.
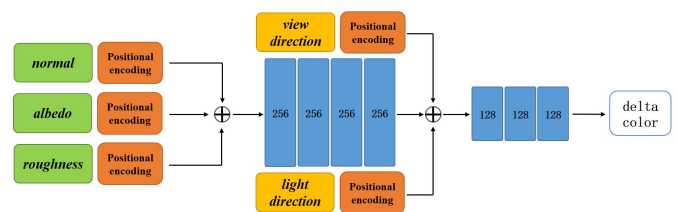

Figure 6. Visualization of our delta network architecture.

layer. Each output branch has 128 channels and different layer numbers (1 for density, 3 for normal and albedo, 2 for roughness).

Fig.6 shows the architecture for delta network module. The input of normal, albedo and roughness after positional encoding are concatenated together to pass a 4-layer MLP with 256 hidden neurons per layer and ReLU activation. It outputs a 128 feature vector. This feature vector is concatenated with the positional encoding of the input viewing direction and lighting direction, and is processed by 4 additional fully-connected ReLU layers, each with 128 channels. The final layer outputs the delta RGB color.

Loss function. Similar to original Neural Reflectance Field, we also force the ray transmittance of the fine network to be

close to 0 or 1. So, our total loss is given by:

$$\sum_q \left\| L^q_{\text{coarse}} - \tilde{L}^q \right\|^2 + \left\| L^q_{\text{fine}} - \tilde{L}^q \right\|^2 + b \left\| N^q - \tilde{N}^q \right\|^2$$
$$+ \left\| L^q_{\text{coarse\_delta}} - \tilde{L}^q \right\|^2 + \left\| L^q_{\text{fine\_delta}} - \tilde{L}^q \right\|^2$$
$$+ c \left[\log \left(\tau^q_c \right) + \log \left(1 - \tau^q_c \right) \right] \tag{12}$$

where q denotes a pixel ray, $b = 0.0001$, and $c = 0.0025$.

Shadow map. For synthetic data, we can directly achieve shadow map from the graphics engine. For real data, we use the origin NeRF to get a rough geometry which is also used to crop the object and use traditional graphics engine like OpenGL or Blender to render a shadow map for a given view.

Run time. We use a single NVIDIA RTX 2080Ti GPU to train network for about 4 days. At inference time, the network takes about 35 seconds to render a 800×600 image.

4. Experiment Results

In this section, We demonstrate two new applications brought by Neural Reflectance Fields, followed by the comparison to vanilla NeRF and the evaluation of our novel contributions.

4.1. Applications

Relighting. With the physical-based BRDF model and the novel volume rendering equation proposed in Neural Reflectance Fields, we can adjust the light source position to enable the relighting effect on rendered images. As illustrated in Figure(7), since all the training data is captured by a single collocated smart phone camera, the novel view image generated by the neural reflectance field also shows the collocated light effect. When we adjust the position of the light source from the middle to the right, the novel view image exhibits the relighting effect, which has striking contrast of the reflection and the shadow, and photo-realistic rendering quality.

Material Segmentation. Another application brought by Neural Reflectance Fields is material segmentation. Since our roughness metric is from 0 to 1, from the neural reflectance field for BRDF calculation, the roughness map itself can be used for material segmentation. As shown in Fig. 8, when generating the training images of synthetic data, we first manually set the roughness of the trousers to 0.2 (smooth), and 0.8 (rough) for the upper body. The rendered image by Neural Reflectance Fields illustrates a significant change in material, such as the specular highlight on the trousers. And the roughness map accumulated by ray sampling is close to the ground truth and can be used for material segmentation. As for the real data shown in
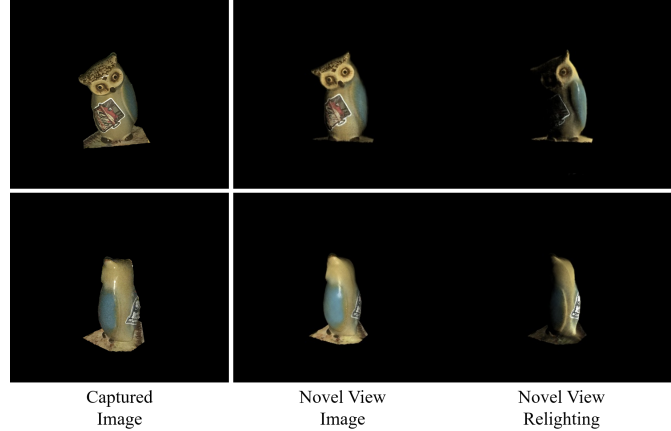

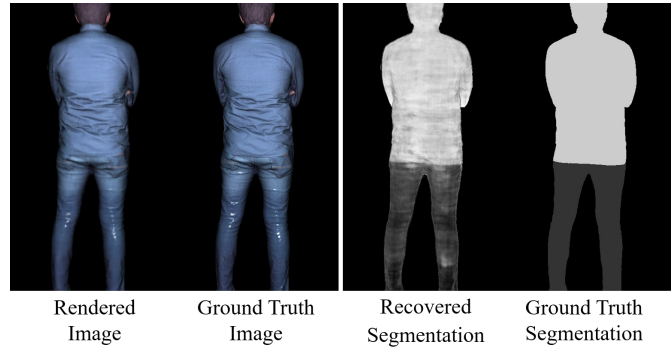Figure 7. Novel view relighting on real data.

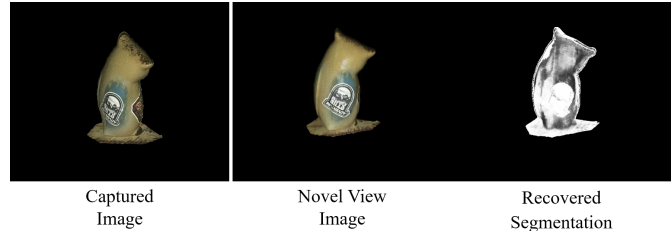Figure 8. Material segmentation on synthetic data.

Figure 9. Material segmentation on real data.

Fig. 9, we put a tag (rough) on the ceramics (smooth) before image capturing. The accumulated roughness map also distinguishes the material change on the object surface.

4.2. Comparison

We compare Neural Reflectance Fields with vanilla NeRF on two real scenes captured by a single collocated smart phone camera. The improvements of rendering quality by Neural Reflectance Fields are mostly in two aspects. At first, Neural Reflectance Fields avoid inconsistent light changing. As illustrated in Fig. 10, the novel view images rendered by NeRF appear salient artifacts due to the inconsistent light changing in training data. There are irreg-
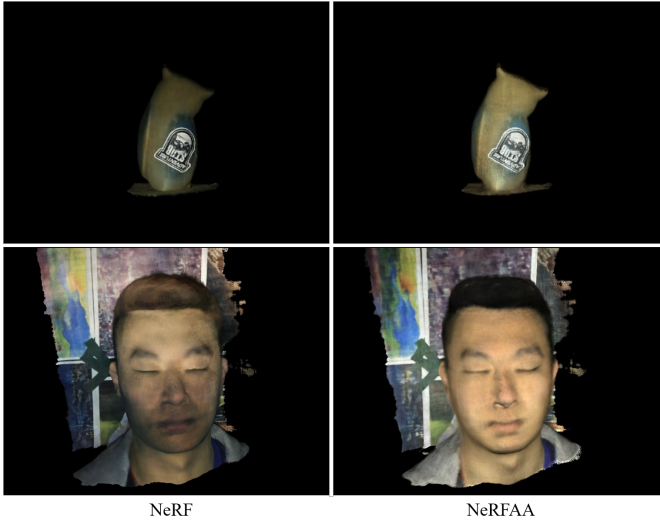
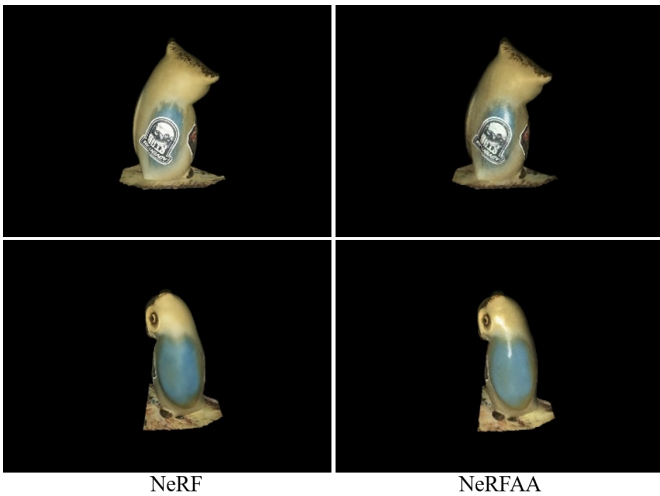Figure 10. Neural Reflectance Fields avoids inconsistent light changing.

Figure 11. Neural Reflectance Fields shows better appearance in specular area.

ular light spots on the human face which makes the whole image unrealistic. However, the images rendered by Neural Reflectance Fields still maintain photo-realistic rendering quality and have smooth shading changes. The second advantage of Neural Reflectance Fields is in rendering objects with reflective materials. As shown in Fig. 11, the whole object rendered by NeRF appears as diffuse reflection whereas Neural Reflectance Fields maintains the highlight on the specular area and the real material of the object.

4.3. Evaluation

Here we provide qualitative and quantitative evaluations of our contributions to original neural reflectance field. The synthetic data was generated in the non-collocated light set-
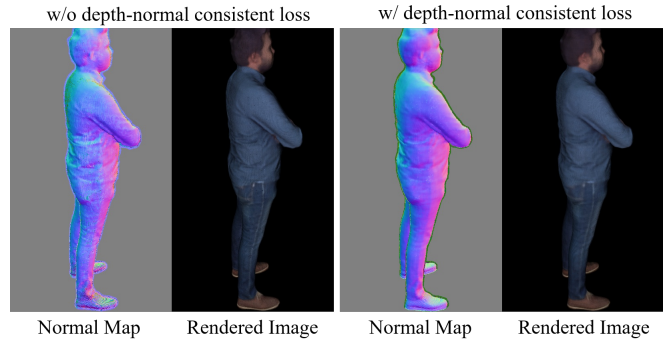

Figure 12. Qualitative evaluation for depth-normal consistent loss.
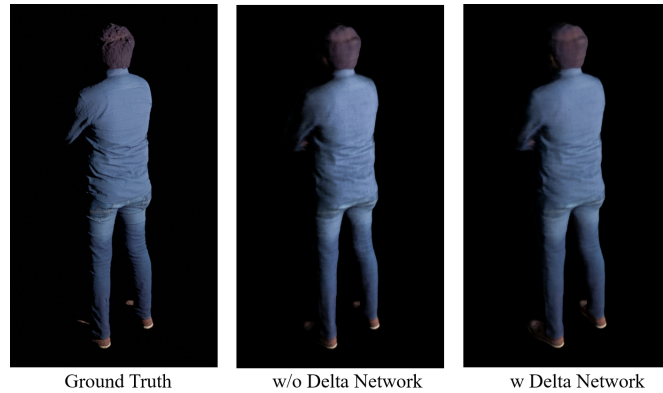

Figure 13. Qualitative evaluation for delta network

ting. As mentioned before, one of our contributions is using the shadow map to avoid tedious point sampling between the light source and the camera ray sampling points. All the results rendered below were generated directly from the network without relighting.

Qualitative Evaluation. We evaluate our two contributions against the original Neural Reflectance Fields baseline. As shown in the Fig. 12, the predicated normal by the original Neural Reflectance Fields have severe flickering in the most area, especially around the waist of the human. However, the model applied our depth-normal consistent loss predicts a smoother normal map without the stripe artifacts. Another contribution for the delta network was also compared to the baseline, as shown in Fig. 13, delta network alleviates the flickering around the back of the human due to the incomplete match between the BRDF model and the real world light condition, and the rendered images are closer to the ground truth.

Quantitative Evaluation. We further quantitative evaluate the rendering equality of our contributions on the novel view synthesis, and compare it against the original Neural Reflectance Fields. We generated a novel camera trajectory with 89 views for the human model and synthesized views for original Neural Reflectance Fields, ours without delta

Table 1. Quantitative evaluation on synthetic data.

	PSNR ↑	SSIM ↑	LPIPS ↓
origin Neural Reflectance Field	30.03	.7428	.1483
w/o Delta Network	30.92	.7440	.1450
w/o Depth-normal Consistent	31.46	.7456	.1404
Ours	**32.60**	**.7472**	**.1375**

network, ours without depth-normal consistent loss and our complete model. These views were unseen during training.

Table(1) summarizes the quantitative results. We use three metrics for the evaluation: Peak Signal-to-Noise Ratio (PSNR), Structual Similarity (SSIM) [32] and Learned Perceptual Image Patch Similarity (LPIPS) [30]. Our contributions improve upon the origianl Neural Reflectance Fields baseline in all metrics.

5. Conclusion

We have reproduced Neural Reflectance Fields for Appearance Acquisition to generate high fidelity relighting and view synthesis results which are significantly better than original NeRF at some challenging appearance effects such as specularities, shadows. Our improved Neural Reflectance Field enables not only cellphone flash images under collocated camera and light, but also non-collocated settings with known camera and light positions. Our self-supervised depth normal consistent loss enables implicit correctness of the predicted normal by building the relationship between surface normal and predicted depth. Our delta network scheme further improves the appearance performance of rendering and increase the expressive power of model. Given the aforementioned distinctiveness, we believe that our improved approach is a step to enable relightable and photo-realistic modeling at both collocated and non-collocated settings, with many potential applications in VR/AR like gaming, entertainment and immersive telepresence.

References

[1] Agisoft photoscan professional. https://www.agisoft.com/downloads/installer, 2019. 5

[2] Miika Aittala, Timo Aila, and Jaakko Lehtinen. Reflectance modeling by neural texture synthesis. *ACM Trans. Graph.*, 35(4), July 2016. 2

[3] Rui Chen, Songfang Han, Jing Xu, and Hao Su. Point-based multi-view stereo network. *CoRR*, abs/1908.04422, 2019. 2

[4] Shuo Cheng, Zexiang Xu, Shilin Zhu, Zhuwen Li, Li Erran Li, Ravi Ramamoorthi, and Hao Su. Deep stereo using adaptive thin volume representation with uncertainty awareness. *CoRR*, abs/1911.12012, 2019. 2

[5] Paul Debevec, Tim Hawkins, Chris Tchou, Haarm-Pieter Duiker, Westley Sarokin, and Mark Sagar. Acquiring the reflectance field of a human face. In *Proceedings of the 27th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '00, page 145–156, USA, 2000. ACM Press/Addison-Wesley Publishing Co. 2

[6] Sing Choong Foo. A gonioreflectometer for measuring the bidirectional reflectance of material for use in illumination computation, 1997. 2

[7] Yasutaka Furukawa and Jean Ponce. Accurate, dense, and robust multiview stereopsis. *IEEE Trans. Pattern Anal. Mach. Intell.*, 32(8):1362–1376, Aug. 2010. 2

[8] Peter Hedman, Julien Philip, True Price, Jan-Michael Frahm, George Drettakis, and Gabriel J Brostow. Deep Blending for Free-Viewpoint Image-Based Rendering. *ACM Transactions on Graphics*, 37(6), 2018. 2

[9] Carlos Hernández Esteban and Francis Schmitt. Silhouette and stereo fusion for 3d object modeling. *Computer Vision and Image Understanding*, 96(3):367 – 392, 2004. Special issue on model-based and image-based 3D scene representation for interactive visualization. 2

[10] Z. Hui, K. Sunkavalli, J. Lee, S. Hadap, J. Wang, and A. C. Sankaranarayanan. Reflectance capture using univariate sampling of brdfs. In *2017 IEEE International Conference on Computer Vision (ICCV)*, pages 5372–5380, 2017. 2

[11] Mengqi Ji, Juergen Gall, Haitian Zheng, Yebin Liu, and Lu Fang. Surfacenet: An end-to-end 3d neural network for multiview stereopsis. 10 2017. 2

[12] Kaizhang Kang, Cihui Xie, Chengan He, Mingqi Yi, Minyi Gu, Zimin Chen, Kun Zhou, and Hongzhi Wu. Learning efficient illumination multiplexing for joint capture of reflectance and shape. *ACM Trans. Graph.*, 38(6), Nov. 2019. 2

[13] Kiriakos N. Kutulakos and Steven M. Seitz. S.m.: A theory of shape by space carving. *International Journal of Computer Vision*, 2000. 2

[14] Marc Levoy and Pat Hanrahan. Light field rendering. In *Proceedings of the 23rd Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '96, page 31–42, New York, NY, USA, 1996. Association for Computing Machinery. 2

[15] Stephen Lombardi, Tomas Simon, Jason Saragih, Gabriel Schwartz, Andreas Lehrmann, and Yaser Sheikh. Neural volumes: Learning dynamic renderable volumes from images. *ACM Trans. Graph.*, 38(4), July 2019. 2

[16] Wojciech Matusik, Hanspeter Pfister, Matt Brand, and Leonard McMillan. A data-driven reflectance model. In *ACM SIGGRAPH 2003 Papers*, SIGGRAPH '03, page 759–769, New York, NY, USA, 2003. Association for Computing Machinery. 2

[17] N. Max. Optical models for direct volume rendering. *IEEE Transactions on Visualization and Computer Graphics*, 1(2):99–108, 1995. 3

[18] 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 *ECCV*, 2020. 1, 2

[19] Giljoo Nam, Joo Ho Lee, Diego Gutierrez, and Min H. Kim. Practical svbrdf acquisition of 3d objects with unstructured flash photography. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia 2018)*, 37(6):267:1–12, 2018. 2

[20] Jannik Boll Nielsen, Henrik Wann Jensen, and Ravi Ramamoorthi. On optimal, minimal brdf sampling for reflectance acquisition. *ACM Transactions on Graphics (TOG)*, 34(6):186:1–186:11, November 2015. 2

[21] Jan Novák, Iliyan Georgiev, Johannes Hanika, and Wojciech Jarosz. Monte carlo methods for volumetric light transport simulation. *Computer Graphics Forum*, 37(2):551–576, 2018. 2

[22] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. *arXiv preprint arXiv:1612.00593*, 2016. 2

[23] J. L. Schönberger and J. Frahm. Structure-from-motion revisited. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4104–4113, 2016. 2

[24] Vincent Sitzmann, Michael Zollhouml;fer, and Gordon Wetzstein. Scene representation networks: Continuous 3d-structure-aware neural scene representations. *ADVANCES IN NEURAL INFORMATION PROCESSING SYSTEMS 32 (NIPS 2019)*, pages 1119–1130, 2019. 2

[25] Pratul Srinivasan, Boyang Deng, Xiuming Zhang, Matthew Tancik, Ben Mildenhall, and Jonathan Barron. NeRV: Neural reflectance and visibility fields for relighting and view synthesis. *https://arxiv.org/abs/2012.03927*, 2020. 4

[26] Pratul Srinivasan, Tongzhou Wang, Ashwin Sreelal, Ravi Ramamoorthi, and Ren Ng. Learning to synthesize a 4d rgbd light field from a single image. pages 2262–2270, 10 2017. 2

[27] Zexiang Xu, Sai Bi, Kalyan Sunkavalli, Sunil Hadap, Hao Su, and Ravi Ramamoorthi. Deep view synthesis from sparse photometric images. *ACM Transactions on Graphics*, 38:1–13, 07 2019. 2

[28] Zexiang Xu, Jannik Boll Nielsen, Jiyang Yu, Henrik Wann Jensen, and Ravi Ramamoorthi. Minimal brdf sampling for two-shot near-field reflectance acquisition. *ACM Trans. Graph.*, 35(6), Nov. 2016. 2

[29] Yao Yao, Zixin Luo, Shiwei Li, Tianwei Shen, Tian Fang, and Long Quan. Recurrent mvsnet for high-resolution multi-view stereo depth inference. *CoRR*, abs/1902.10556, 2019. 2

[30] R. Zhang, P. Isola, A. A. Efros, E. Shechtman, and O. Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018. 8

[31] Tinghui Zhou, Richard Tucker, John Flynn, Graham Fyffe, and Noah Snavely. Stereo magnification: Learning view synthesis using multiplane images. *CoRR*, abs/1805.09817, 2018. 2

[32] Zhou Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, 2004. 8