

# Relightable Holoported Characters: Capturing and Relighting Dynamic Human Performance from Sparse Views

## –Supplementary Material–

Kunwar Maheep Singh<sup>1</sup> Jianchun Chen<sup>1,2</sup> Vladislav Golyanik<sup>1</sup> Stephan J. Garbin<sup>3</sup>  
 Thabo Beeler<sup>4</sup> Rishabh Dabral<sup>1,2</sup> Marc Habermann<sup>1,2</sup> Christian Theobalt<sup>1,2</sup>

<sup>1</sup> Max Planck Institute for Informatics, Saarland Informatics Campus <sup>2</sup> VIA Research Center <sup>3</sup> Google, London <sup>4</sup> Google, Zurich

### 1. Overview

In this supplemental document, we provide further details on the pre-scanned character model (Sec. 2). Next, we describe how the diffuse features are computed (Sec. 3). We then present a more in-depth explanation of our cross-attention mechanism (Sec. 4), Gaussian Projection (Sec. 5), and network architectures (Sec. 6). Following this, implementation details, training procedures, and loss functions are discussed (Sec. 7). Finally, we detail the data capture and hardware setup (Sec. 8) and the baselines (Sec. 9), provide additional information on our dynamic OLAT experiment (Sec. 10), give a runtime and memory breakdown along with comparisons to baselines (Sec. 11), and present further ablations (Sec. 12) as well as additional qualitative results (Sec. 13).

### 2. Character Animation Model

For each subject defined in Sec. 4.1 in the main paper, we estimate the skeletal motion using dense camera views with Captury [10]. In addition, we use a body scanner to obtain a template mesh per subject with 4890 vertices and 9700 faces, which is later simplified as an embedded graph with 489 nodes. Paired with the template mesh, a person-specific skeleton is produced with 173 joints, driven by the motion parameters  $\theta$  with 107 degrees of freedom. We generate the skinning weight  $W \in \mathbb{R}^{4890 \times 68}$  from Blender [3] autorig based on a subset of body and hand joints.

We train the Character Animation Networks  $\mathcal{G}_{\text{eg}}$  and  $\mathcal{G}_{\text{delta}}$  with multi-view images and NeuS2 [13] supervisions in the tracking frames. They produce 3D geometry interpolation in relit frames for training the neural relightable model, as well as the extrapolated geometry for novel motions during inference. Moreover, we annotate 20 part segmentation labels on the mesh vertices using [7] and assign rigidity weights, which facilitates the Laplacian and ARAP loss in training the *Character Animation Network*.

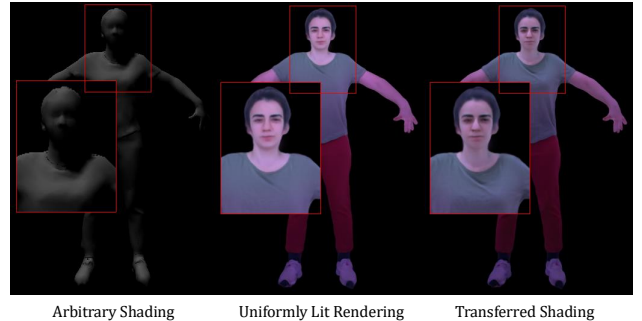


Figure 1. Our model enables the transfer of arbitrary shading to the rendered output. **Left.** diffuse shading computed under an arbitrary light source. **Middle.** uniformly lit rendering predicted by our model. **Right.** rendering obtained by providing the computed shading instead of the uniform diffuse shading. We note the correct transfer of shadows from our shading feature to the model rendering, showcasing the disentangled control our model provides.

### 3. Diffuse Shading

The diffuse shading feature  $d$  is computed via a CPU-based one-bounce ray tracer from Open3D [15]. We first map the light sources from the environment map to the light emitters in the lightstage with pre-calibrated 3D positions of LEDs. Then, we use this position to calculate the direction of incoming light for diffuse shading and occlusions. Jointly with the 3D position map  $p$ , the shading features encode signals about the light-to-body distance, which particularly helps to recreate near-field lighting effects.

We show the disentangled control this shading allows in the final rendering in Fig. 1. Here, we use the diffuse shading from an OLAT light condition to replace the one in a uniformly lit rendering, and notice the shadowing effects in the final rendering exactly mirror those from the diffuse shading.

### 4. Cross-attention Mechanism

The cross-attention layer processes the feature descriptor  $h_i$  corresponding to texel  $i$  in the higher-layer feature map of

*RelightNet* with the RGB intensity value  $e_j$  of pixel  $j$  of the environment map  $\mathbf{E}$ . Let  $\mathbf{q}$  denote the 2D coordinate of pixel  $j$ . The intensity value  $e_j$  is first concatenated with 64 dimensional positional encoding  $P(\mathbf{q})$  [11], then linearly mapped into  $\mathbf{K}_e, \mathbf{V}_e$ .

$$\mathbf{K}_e = \mathbf{W}_K[e_j, P(\mathbf{q})], \quad \mathbf{V}_e = \mathbf{W}_V[e_j, P(\mathbf{q})] \quad (1)$$

Similarly, the feature descriptor  $\mathbf{h}_i$  is linearly transformed into query vector  $\mathbf{Q}_f$ :

$$\mathbf{Q}_f = \mathbf{W}_Q \mathbf{h} \quad (2)$$

where  $\mathbf{W}_{\{Q,K,V\}}$  are the learnable weights. The integration of positional encoding with environment lights ensures that the spatial relationships within the environment map are preserved during the attention process.

## 5. Gaussian Projection and Rendering

Each 3D Gaussian, predicted by our *RelightNet*, is defined by its center  $\mathbf{p}_i$ , scaling  $\mathbf{s}_i$ , rotation  $\mathbf{r}_i$ , opacity  $\alpha_i$ , and color  $\mathbf{c}_i$ . Following EWA splatting [16], we project each Gaussian into the image plane, obtaining a 2D center  $\mu_i$  and covariance matrix  $\Sigma_i$  as

$$\Sigma_i = \mathbf{J}_i \mathbf{W}_i \mathbf{R}_i \text{diag}(\mathbf{s}_i) \text{diag}(\mathbf{s}_i)^\top \mathbf{R}_i^\top \mathbf{W}_i^\top \mathbf{J}_i^\top, \quad (3)$$

where  $\mathbf{J}_i$  is the Jacobian of the viewing transformation,  $\mathbf{W}_i$  the camera-to-world matrix, and  $\mathbf{R}_i$  the rotation from quaternion  $\mathbf{r}_i$ . Given the 2D Gaussian  $G(\mathbf{x}; \mu_i, \Sigma_i)$ , the color of pixel  $p$  is computed as a weighted alpha-composited sum:

$$\mathbf{C}_p = \sum_{j \in \mathcal{N}} \mathbf{c}_j \alpha_j \prod_{k=1}^{j-1} (1 - \alpha_k), \quad (4)$$

where  $\alpha_j$  denotes the transparency of the  $j$ -th Gaussian, modulated by its contribution  $G(\mathbf{x}_p; \mu_j, \Sigma_j)$ . This formulation ensures smooth, view-consistent appearance reconstruction across multiple viewpoints.

## 6. RelightNet

*RelightNet* consists of interleaved convolutional, self-attention, and cross-attention layers for its encoder and decoder. Due to the high computational cost of self-attention and cross-attention at high resolution ( $\geq 64 \times 64$ ), we only use convolutional layers at these resolutions. For the higher layers of the *RelightNet* with low-resolution feature maps, we follow every convolutional layer with a self-attention layer and a cross-attention layer between the UV space features and the environment map. Here, self-attention is computed between flattened elements of the feature map. The number of heads in self-attention and cross-attention is set as 4. Table 1 illustrates the concrete architecture of *RelightNet*.

Table 1. **Illustration of the *RelightNet* architecture.** In the operation column, "C" denotes a convolution layer, "SA" denotes a self-attention layer, "CA" denotes a cross-attention layer, "DS" and "US" denote down-sampling and up-sampling layers with scale factors equal to 2.

	Number of Feature Channels	Operation
Input	24	N/A
Block 1	(32,32,32)	(C, C, DS)
Block 2	(48,48)	(C, DS)
Block 3	(64,64)	(C, DS)
Block 4	(64,64,64,64)	(C, SA, CA, DS)
Block 5	(128,128,128,128)	(C, SA, CA, DS)
Block 6	(256,256,256,256)	(C, SA, CA, DS)
Block 7	(256,256,256)	(C, SA, CA)
Block 8	(256,256,256,256)	(C, SA, CA, C)
Block 9	(256,256,256)	(C, SA, CA)
Block 10	(256,256,256,256)	(C, SA, CA, US)
Block 11	(256,256,256)	(C, SA, CA)
Block 12	(256,256,256, 256)	(C, SA, CA, US)
Block 13	(128,128,128)	(C, SA, CA)
Block 14	(128,128,128,128)	(C, SA, CA, US)
Block 15	(64,64,64)	(C, SA, CA)
Block 16	(64,64)	(US, C)
Block 17	(48,48)	(US, C)
Block 18	(32,32,14)	(US, C, C)
Output	14	N/A

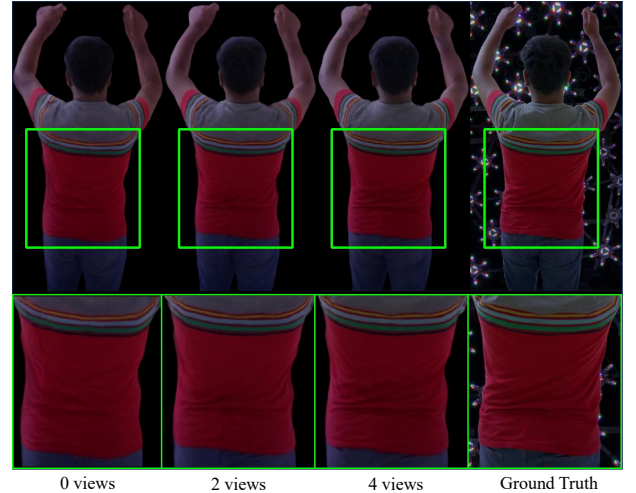


Figure 2. **Effect of the number of input views for our model.** As the number of input views decrease, we note the model begins to hallucinate details.

## 7. Training Details

**Character Animation Networks**  $\mathcal{G}$  consists of two graph neural networks  $\mathcal{G}_{\text{eg}}, \mathcal{G}_{\text{delta}}$  trained separately in uniformly lit frames. In particular,  $\mathcal{G}_{\text{eg}}$  is supervised with silhouette loss over multiview segmented images, and chamfer loss over NeuS2 [13] reconstructed point clouds. Since embedded graph deformation has a lower degree of freedom, we

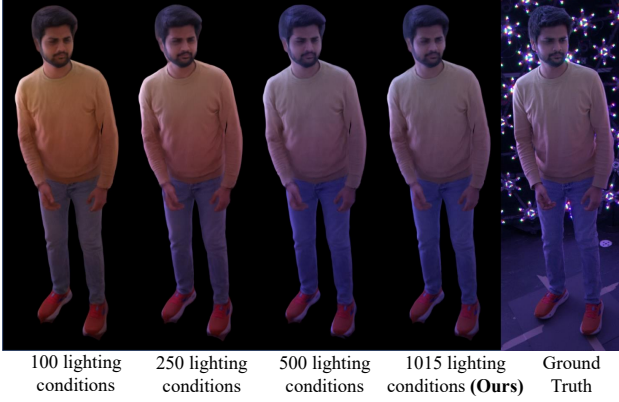


Figure 3. **Effect of the number of lighting conditions our model is exposed to at training time.** As the number of conditions decreases, the model overfits and performance drops, highlighting the importance of diverse lighting for end-to-end relighting.

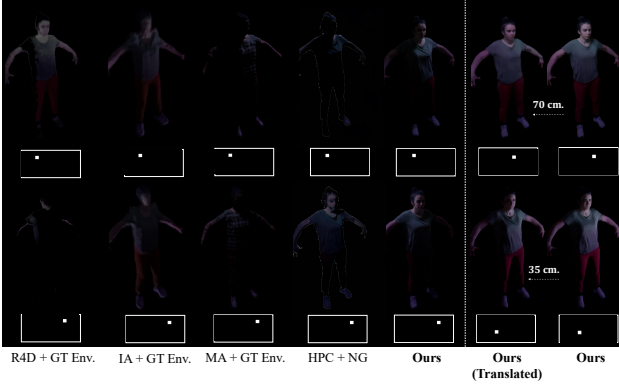


Figure 4. **OOD comparison.** Here, we compare our method on out-of-distribution lighting conditions, i.e. OLAT environment maps. Notably the model never saw OLAT environment maps during training. Nonetheless, it can generate plausible results while competing methods either produce blurry renderings or completely fail. Moreover, we illustrate that our method can reproduce near field lighting effects by translating the human by 35cm, i.e. modifying the positional map and diffuse shading, and we can observe a plausible change in illumination.

introduce an as-rigid-as-possible loss as a regularization.

$$L_{EG} = L_{Sil} + L_{Chamfer} + L_{ARAP} \quad (5)$$

The per-vertex offset is driven by additional photometric losses, with stronger regularization losses including Laplacian loss and Isometry loss.

$$L_{Delta} = L_{Sil} + L_{Photo} + L_{Chamfer} + L_{Lap} + L_{Iso} \quad (6)$$

*AlbedoNet*  $\mathcal{H}$  is trained with dense-view image capture in uniformly lit frames using photometric losses. Consequently, similar to the high-frequency normal estimation  $\hat{n}$ , the albedo feature  $\hat{p}$  incurs a one-frame delay when incorporated into the training process of our neural relightable

model for subsequent relit frames. *AlbedoNet* is supervised with  $L_1$  loss over multiview image captures in the uniformly lit frames. Unlike Holoported Character [9], we leverage a one-stage UNet architecture without additional super-resolution. The multiview images are resized into  $540 \times 1024$  as it is sufficient to represent high-quality albedo features in a  $512 \times 512$  texture map.

*RelightNet*  $\mathcal{F}$  is supervised with multi-view images. Unlike the Character Animation Networks  $\mathcal{G}$  and *AlbedoNet*  $\mathcal{H}$  that are trained with uniformly lit frames, *RelightNet* is trained with the interleaved relit frames using photometric losses and regularization terms reweighted by  $\lambda$ :  $L_{train} = \lambda_1 L_1 + \lambda_2 L_{reg} + \lambda_3 L_{vgg}$ . Here,  $L_1$  is the mean of the pixel-wise  $L_1$  distance between the rasterized image  $I_{pred}$  and the ground-truth image  $I_{gt}$ .  $L_{vgg}$  is the VGG perceptual loss [6]. The regularization term  $L_{reg}$  penalizes the deviations of scaling offsets  $s_i$  from 1 and position offsets  $\delta p_i$  from 0. However, the training of *RelightNet* can diverge in the early stage due to the undesired prediction of opacity and scaling parameters. To stabilize training, we further introduce a warm-up phase where we regularize the initial Gaussian prediction with Eq. 7.

$$L_{warmup} = \frac{1}{N_G} \sum_{i=1}^{N_G} (\lambda_s \|s_i - 1\|_2^2 + \lambda_t \|\delta p_i\|_2^2 + \lambda_o \|o_i - o_0\|_2^2 + \lambda_c \|c_i - c_i^{tem}\|_2^2), \quad (7)$$

where  $c_i^{tem}$  is the color of the texture map of the template mesh captured in a fully lit environment, and  $o_0 = 0.95$  is the default opacity. In this stage, the input environment map  $E$  is set as a uniformly lit environment map. This loss is linearly switched to  $L_{train}$  between 25,000 and 26,000 iterations.

We train *RelightNet* for 360,000 iterations on four H100s with a batch size of 4 and accumulate gradients every second iteration. We use the ADAM optimizer with a learning rate set to a constant  $1e-4$ . For hyperparameters, we use  $\lambda_1 = 2.7$ ,  $\lambda_2 = 1.2$ ,  $\lambda_3 = 75$ ,  $\lambda_{scale} = 1$ ,  $\lambda_{pos} = 1$ ,  $\lambda_s = 1$ ,  $\lambda_t = 0.1$ ,  $\lambda_o = 1$ ,  $\lambda_c = 1$ .

## 8. Additional Data Capture Details

The lightstage we use is equipped with 331 calibrated LED panels and 40 cameras, operating at 60 *fps* with 4K resolution. In total, we capture five sequences, each featuring a different identity and unique clothing. Each subject was captured twice: once for training and once for testing, ensuring that the test set contains novel poses. We sampled 1,015 environment maps from the Laval Indoor HDR dataset [4] for the training set, and held 8 environment maps out for the test set. These environment maps were downsampled to a resolution of  $256 \times 512$ , with each

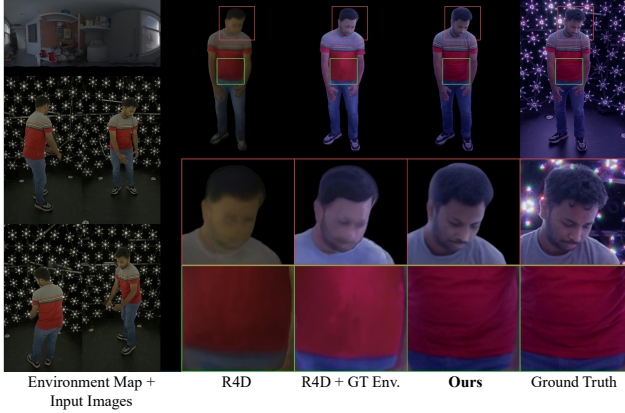


Figure 5. **Effect of using diverse illumination and ground truth environments for training.** On the left, we show test results of R4D [2] trained under uniform illumination only. Next, we show the same baseline trained with diverse illumination conditions using the ground truth environment maps (R4D + GT Env). We note significant generalization improvement for R4D with this training strategy.

pixel mapped to the nearest LED panel in 3D space. The intensities of pixels mapped to the same LED panel were averaged, and the resulting environment map was linearly normalized to match the maximum intensity that the LED panels could simulate. These processed environment maps are considered the ground-truth environment maps for all experiments.

## 9. Baseline Implementation Details

**Relighting4D.** Relighting4D [2] is originally designed for *replay* of human performance captured from a single-view camera, which optimizes a time-dependent latent vector that spans the training sequence. We extend Relighting4D as an animatable and relightable approach trained with multiview supervision. To enable the generalization ability to novel motion sequences, in each test frame, we provide the closest latent vector from the training sequence. In addition, we replace the original used SMPL [8] geometry with our high-quality mesh-based character model for a fair comparison. The Relighting4D is trained with uniformly lit frames, which contain a static illumination with a wide variety of motions. At test time, we scale our normalized environment maps using an optimized radiance scaling factor.

**Relighting4D + GT Env Map.** For this variant of Relighting4D, we use uniformly lit frames to train the geometry features of Relighting4D following the same training scheme as before. For the material features estimation, we utilize the supervision of the relit frames. Instead of jointly optimizing the full environment map with material parameters, this baseline takes the ground truth environment map as input, while optimizing one scaling factor for each scene, which rescales the normalized environment map to be com-

patible with the underlying BRDF model.

**IA + GT Env Map.** We train Intrinsic Avatar in a similar way to R4D + GT Env. Note that, unlike Relighting4D, it is non-trivial for us to upgrade the underlying mesh with our tracking result in IA, as IA is designed with strong entanglement with SMPL articulation. Therefore, we rely on the original method’s SMPL tracking. We also note that due to a bug in the code, the pose correction module was not working and had to be disabled during training.

**MA + GT Env Map.** We train Mesh Avatar similarly to IA + GT Env. For this baseline, we train directly with the relit frames, since it performs geometry optimization along with material parameter optimization.

**Holoported Characters.** Holoported Characters [9] produces human rendering under similar lighting conditions at training and test times. It is trained using the multiview images from uniformly lit frames for supervision. In the test time, we leverage geometry features and mesh from the relit frame and the partial texture of the prior uniformly lit frame, strictly following our testing setup.

**Holoported Character + Neural Gaffer.** Neural Gaffer [5] is a generative image-to-image relighting method. It is a pretrained model that takes an input image and an environment map to provide a relit image. We fine-tune Neural Gaffer for each identity separately and use it to translate the novel view rendering of Holoported Characters to obtain a relit output. Since the environment map in Neural Gaffer is considered object-centric, for each test camera, we transform the global space environment into camera space accordingly.

## 10. Dynamic OLAT Experiment

Since OLAT images are darker than environment-lit images, we add structured supervision to stabilize training, using coarse UV-space Microfacet rendering. We first optimize albedo and roughness maps on the DDC-tracked mesh using the Microfacet BRDF formulation of [2], supervised with ground-truth relit data. We compute per-textel normals, per-light visibility, and optimize a scaling parameter for the environment map. The resulting maps are used to render a coarse UV-space image, which serves as the input to the network in place of the diffuse shading feature. Instead of directly predicting relit colors, the network learns to predict additive color deltas over the coarse rendering. Evaluation follows our standard protocol with three novel views, novel poses, and eight held-out OLAT lighting conditions. At test time, we render the full OLAT basis and linearly combine them to reconstruct target environment maps.

Since the OLAT frames were extremely dark, we had to use a frame interpolation network [14] to generate intermediate uniformly lit frames on which we compute the foreground masks. Also, compared to 360,000 iterations of our model, this model took 1,200,000 iterations to converge.



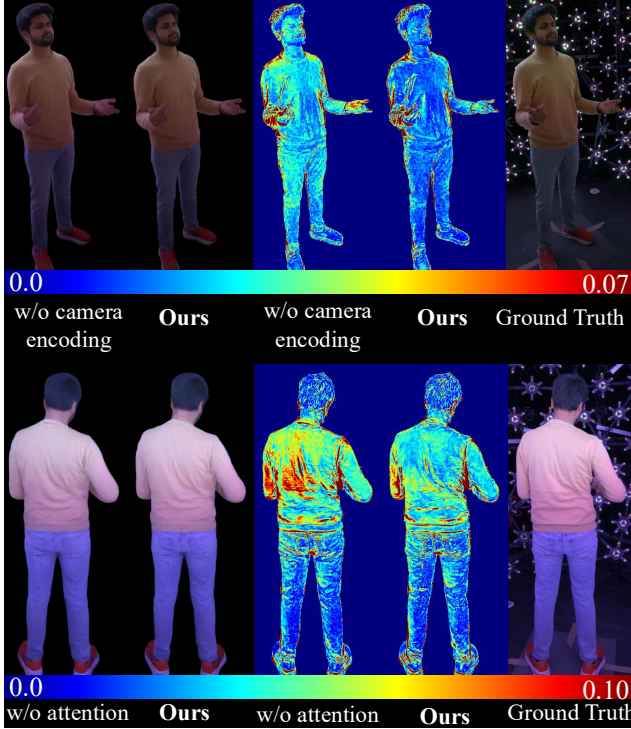


Figure 6. **Additional qualitative ablations.** Following Fig. 6 in the main paper, we demonstrate additional visual ablation results. By removing the camera encoding, our model is no longer able to learn view-dependent effects. Without the attention mechanism, the model is unable to learn the correct correlation with the environment.

## 11. Runtime and Memory

Tab. 2 reports a per-component runtime breakdown (H100; averaged over 100 iters; no data loading). The main bottleneck is Sapiens normal estimation (amenable to distillation or lighter model). Tab. 3 compares runtime and memory footprint against baselines. RelightNet has  $\approx 27$ M parameters. Overall, our full pipeline runs at  $\approx 2$  FPS, substantially faster than radiance-field-based R4D/IA (seconds per frame) and HPC+NG (multiple diffusion denoising steps), but slower than MA (7 FPS); a lightweight variant of our model without the Sapiens normal prior reaches up to 9 FPS.

## 12. Further Ablations

We perform additional ablations and present both quantitative (Tab. 4) and qualitative results (Fig. 8).

**Importance of Physics-informed Features.** We ablate all physics-informed features. Performance drops significantly when removing them (Tab. 4, Fig. 8), confirming that our gains arise from the combination of physics-informed features, network design, and capture strategy.

**Sparse-View Tracking.** Using skeleton tracking

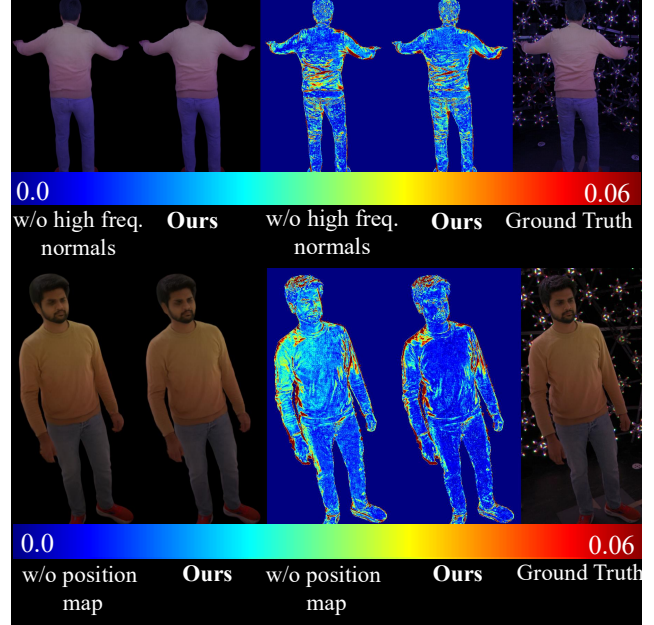


Figure 7. **Additional qualitative ablations.** Following Fig. 6 in the main paper, we demonstrate additional visual ablation results. Without the high-frequency normals, our method is unable to model wrinkle details correctly. Excluding the position map leads to incorrect relighting, especially when the subject approaches a light source.

Table 2. **Runtime Breakdown**

Component	Runtime (ms.)
Character Animation Module	9.41
Sapiens	244.10
AlbedoNet	15.31
Diffuse Shading (Open3D/Optix)	163.60 / 26.73
Texture Unprojections	46.78
RelightNet + Rasterization	30.50
<b>Total (Open3D/Optix)</b>	<b>509.70 / 372.83</b>

Table 3. **Runtime/Memory vs. Baselines**

Method	sec.	MB
R4D	8.49	8399
IA	32.72	9181
MA	0.15	23407
HPC	0.08	40784
HPC+NG	2.13	45831
Ours	0.51	49884

from four sparse views slightly degrades performance but still yields high-quality renderings (Supplementary Video); tracking error w.r.t. dense-view ground truth is MPJPE = 18.54 mm, mostly from hands. With two-view tracking integrated with our 2-view ablation (Tab. 2 of main paper), tracking errors increase substantially, but we still observe visually coherent renderings in many cases (Tab. 4, Fig. 8), suggesting feasibility of a setup with reduced input requirements, with improved tracking or learned pose priors.

Table 4. **Further Ablations.** We ablate the need for physics-informed features, evaluate performance when both, skeleton tracking and image inputs are limited to just two views, and assess RelightNet’s cross-identity generalization by measuring few-shot adaptation performance.

Method	PSNR $\uparrow$	LPIPS $\downarrow$	SSIM $\uparrow$
w/o any features	29.68	7.65	84.45
w/ 2-view tracking	27.82	10.25	81.35
w/ few-shot adaptation (FSA)	29.07	10.09	83.72
<b>Ours</b>	<b>32.07</b>	<b>5.55</b>	<b>89.34</b>

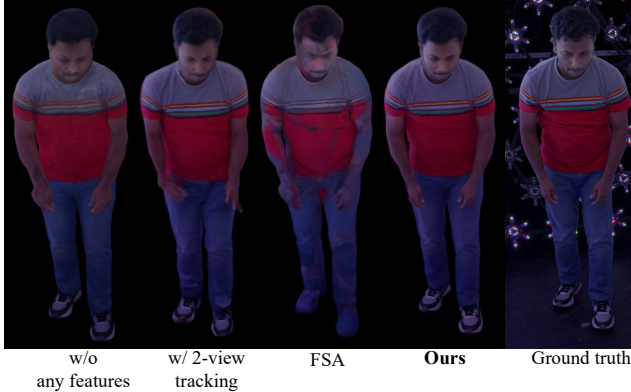


Figure 8. **Additional qualitative ablations.** We provide further visual ablation results. Removing all physics-informed features causes a significant drop in rendering quality. Limiting tracking and image observations to only two views primarily reduces performance due to inaccurate tracking. Finally, few-shot adaptation of RelightNet introduces artifacts arising from the material and UV bias of the original subject.

**Few-shot Adaptation (FSA).** We finetune a Subject-2 checkpoint on 12 Subject-5 samples (diverse poses/lighting/viewpoint), updating Input/Output blocks and Blocks 1–3, 16–18 (Tab. 1) for 1k iterations. The adapted model shows artifacts (Tab. 4, Fig. 8) due to Subject-2 material/UV bias. We acknowledge this as a main limitation and consider it for future work, since recent developments in generalizable priors show great promise.

### 13. Additional Qualitative Results

We conduct extensive experiments with qualitative results to demonstrate the superiority and robustness of our approach.

**Relighting4D Training Strategy.** In Fig. 5, we show the effect of training the Relighting4D [2] baseline with uniformly lit frames vs re-lit frames with ground truth environment maps. We observe more generalisable optimisation of material parameters by training with the re-lit frames and ground truth environment maps input.

**Impact of Lighting Diversity.** We analyze the impact of our data capture strategy for dynamic human avatar relighting. To assess lighting diversity, we progressively reduce the number of environment maps a subject is exposed

Table 5. **Impact of Lighting Diversity.** We ablate different choices of our data capture strategy.

Number of lighting conditions	PSNR $\uparrow$	LPIPS $\downarrow$	SSIM $\uparrow$
100	28.79	9.12	88.70
250	29.91	7.71	88.95
500	<u>30.91</u>	<b>6.97</b>	<u>89.54</u>
<b>1015 (Ours)</b>	<b>31.38</b>	<u>7.01</u>	<b>90.00</b>

to (Tab. 5, Fig. 3). Relighting quality clearly degrades with fewer lighting conditions, showing that diverse illumination is crucial for learning accurate relighting.

**Number of Views.** In Fig. 2, we show the effect of the number of views our model uses on the final rendering quality. As decreasing the number of views leads to more information being occluded, we observe lower reconstruction fidelity of high-frequency details as well as more hallucination.

**Generalization to OLAT and Near Field Relighting.** We test the robustness of our method in challenging lighting conditions, *i.e.* OLAT and near-field lighting, which is out of the distribution of our training environments (see Fig. 4). Without explicit modeling of light linearity, our learning-based *RelightNet* can still generate plausible shading for these illuminations. In contrast, NG fails to disentangle identity and shading in the OLAT setting, R4D + GT Env. based on empirical BRDF models can produce reasonable but imprecise OLAT effects. IA/MA + GT Env. lead to artifacts similar to R4D + GT Env. This demonstrates that our models learns the correct light-material disentanglement, even without being trained on OLAT data.

**Ablations.** In Fig. 6 and Fig. 7, we provide extra ablation results following Fig. 6 of the main paper to validate other design choices. We are able to see the visual impact of our design decisions, demonstrating that each component plays an important role in learning the correct light transport.

**Miscellaneous Results.** In Fig. 9, we show additional qualitative comparisons, where our method constantly achieves superior results with sharper clothing and facial details, with more accurate color shading. In Fig. 10, we test the robustness of our relightable avatar against novel illuminations in addition to OLAT/near-field lightings, including outdoor environments downloaded from Poly Haven<sup>1</sup>. Our proposed method renders consistent and sharp visual details with realistic shadings corresponding to input environment maps.

<sup>1</sup><https://polyhaven.com/>

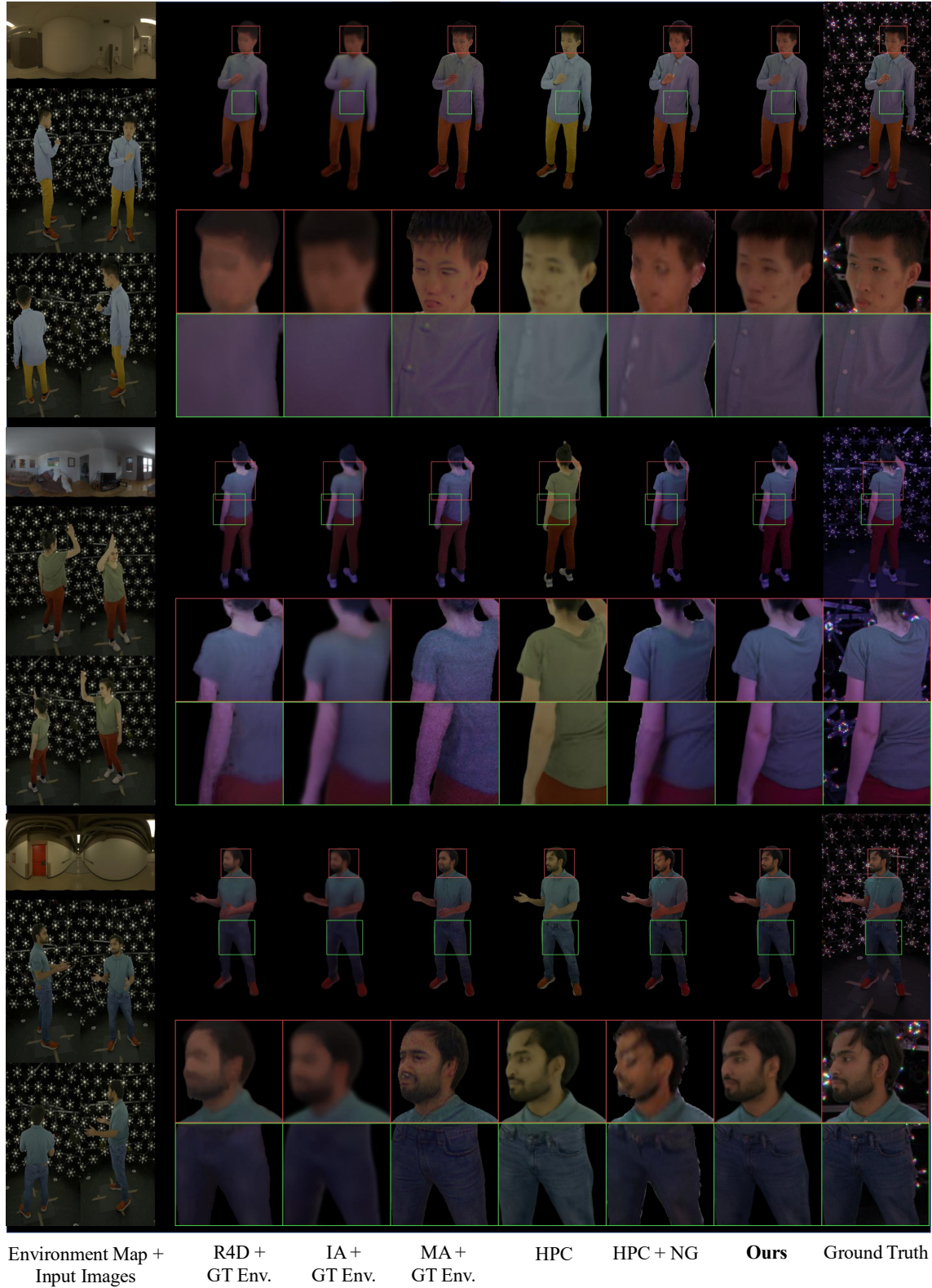


Figure 9. **Additional Qualitative Comparisons.** We show additional visual comparison between our proposed baseline methods, *i.e.* R4D [2], IA [12] and, MA [1] and, HPC [9] with/without NG [5] postprocessing on three other subjects. This figure showcases the significant improvement of our method in rendering and relighting quality among all subjects, which better supports our conclusion drawn in Fig. 5 of the main paper.





Figure 10. **Qualitative Results in Novel Illumination.** We present additional qualitative results under versatile unseen environments, including two outdoor environments downloaded online. Our model can generate realistic shading, including self-shadows, while preserving visual details under novel illuminations with different brightness, color tone, etc.



## References

- [1] Yushuo Chen, Zerong Zheng, Zhe Li, Chao Xu, and Yebin Liu. Meshavatar: Learning high-quality triangular human avatars from multi-view videos. In *European Conference on Computer Vision*, pages 250–269. Springer, 2024. [7](#)
- [2] Zhaoxi Chen and Ziwei Liu. Relighting4d: Neural relightable human from videos. In *European Conference on Computer Vision*, pages 606–623. Springer, 2022. [4](#), [6](#), [7](#)
- [3] Blender Online Community. *Blender - a 3D modelling and rendering package*. Blender Foundation, Stichting Blender Foundation, Amsterdam, 2018. [1](#)
- [4] Marc-André Gardner, Kalyan Sunkavalli, Ersin Yumer, Xiaohui Shen, Emiliano Gambaretto, Christian Gagné, and Jean-François Lalonde. Learning to predict indoor illumination from a single image. *ACM Transactions on Graphics (TOG)*, 36(6):1–14, 2017. [3](#)
- [5] Haian Jin, Yuan Li, Fujun Luan, Yuanbo Xiangli, Sai Bi, Kai Zhang, Zexiang Xu, Jin Sun, and Noah Snavely. Neural gaffer: Relighting any object via diffusion. *Advances in Neural Information Processing Systems*, 37:141129–141152, 2024. [4](#), [7](#)
- [6] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *Computer Vision–ECCV 2016: 14th European Conference, Amsterdam, The Netherlands, October 11–14, 2016, Proceedings, Part II 14*, pages 694–711. Springer, 2016. [3](#)
- [7] Peike Li, Yunqiu Xu, Yunchao Wei, and Yi Yang. Self-correction for human parsing. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 44(6):3260–3271, 2020. [1](#)
- [8] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 851–866. 2023. [4](#)
- [9] Ashwath Shetty, Marc Habermann, Guoxing Sun, Diogo Luvizon, Vladislav Golyanik, and Christian Theobalt. Holoported characters: Real-time free-viewpoint rendering of humans from sparse rgb cameras. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1206–1215, 2024. [3](#), [4](#), [7](#)
- [10] TheCapture. Capture motion capture redefined: Go markerless., 2020. [1](#)
- [11] A Vaswani. Attention is all you need. *Advances in Neural Information Processing Systems*, 2017. [2](#)
- [12] Shaofei Wang, Bozidar Antic, Andreas Geiger, and Siyu Tang. Intrinsicavatar: Physically based inverse rendering of dynamic humans from monocular videos via explicit ray tracing. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1877–1888, 2024. [7](#)
- [13] Yiming Wang, Qin Han, Marc Habermann, Kostas Daniilidis, Christian Theobalt, and Lingjie Liu. Neus2: Fast learning of neural implicit surfaces for multi-view reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3295–3306, 2023. [1](#), [2](#)
- [14] Guozhen Zhang, Chuxnu Liu, Yutao Cui, Xiaotong Zhao, Kai Ma, and Limin Wang. Vfimbamba: Video frame interpolation with state space models. *Advances in Neural Information Processing Systems*, 37:107225–107248, 2024. [4](#)
- [15] Qian-Yi Zhou, Jaesik Park, and Vladlen Koltun. Open3d: A modern library for 3d data processing. *arXiv preprint arXiv:1801.09847*, 2018. [1](#)
- [16] Matthias Zwicker, Hanspeter Pfister, Jeroen Van Baar, and Markus Gross. Ewa splatting. *IEEE Transactions on Visualization and Computer Graphics*, 8(3):223–238, 2002. [2](#)