Relightable Neural Actor with Intrinsic Decomposition and Pose Control —Supplementary Material—

Diogo Carbonera Luvizon[1,2], Vladislav Golyanik[1], Adam Kortylewski[1,3], Marc Habermann[1,2], and Christian Theobalt[1,2]

[1] Max Planck Institute for Informatics, Saarland Informatics Campus
[2] Saarbrücken Research Center for Visual Computing, Interaction and AI
[3] University of Freiburg
{dluvizon,golyanik,akortyle,mhaberma,theobalt}@mpi-inf.mpg.de
https://vcai.mpi-inf.mpg.de/projects/RNA

This document accompanies our paper "Relightable Neural Actor with Intrinsic Decomposition and Pose Control" and includes additional implementation details on network architectures (Appendix A), dataset collection (Appendix B), and additional results and comparisons of our method (Appendix C). Our dynamic results are provided in the supplementary video.

A Implementation Details and Network Architectures

A.1 Pose-driven Geometry Model

We represent the human geometry as an implicit field guided by the skeletal pose parameters $\mathbf{P}_t$. We follow the architecture from Neural Actor [7]. However, differently from the prior work, we sample image patches during training, allowing our method to process only the triangles that intersect the patch region and enabling the use of perceptual loss during training, as described in Section 3.3 of the main paper. For completeness, we illustrate the neural network architecture of the pose-driven geometry model in Fig. 9. The RGB color output $\mathbf{c}$ is activated with the *sigmoid* function, and the density output σ is activated with ReLU. The yellow rectangular blocks represent a single fully connected layer.

Since our goal at this stage is to obtain a drivable high-quality geometry, we only use the RGB color output $\mathbf{c}$ for supervising the geometry model. The loss for this part is defined as:

$$\mathcal{L}_{\text{geo}} = \lambda_{\text{L2}}\mathcal{L}_{\text{L2}} + \lambda_{\text{vgg}}\mathcal{L}_{\text{vgg}} + \lambda_{\sigma}\mathcal{L}_{\sigma}, \tag{1}$$

where $\mathcal{L}_{\text{L2}}$ and $\mathcal{L}_{\text{vgg}}$ are the L2 and perceptual losses [5] between the ground-truth and the predicted images, and $\mathcal{L}_{\sigma}$ pushes the density in the empty space to zero. Specifically, the density defined as $\sigma = \text{ReLU}(\sigma')$ suffers from zero-gradient on empty regions and the pre-activated σ' is pushed towards negative values with

$$\mathcal{L}_{\sigma} = (1 - \mathbf{M}(\mathbf{r}))\,\text{sigmoid}(\sigma'), \tag{2}$$

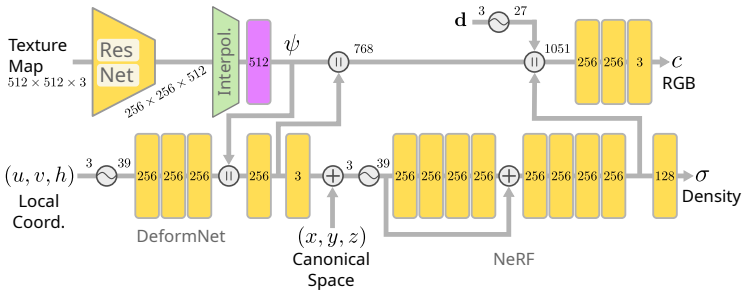

Fig. 9: Neural network architecture of the pose-driven geometry model adapted from Neural Actor [7]. The RGB color branch is only used during the training phase of the geometry model and is discarded during the relighting training and inference. The interpolation of texture features (obtained from the texture map) is performed in the UV coordinate (u, v) given by the projection of 3D points onto the human mesh. The obtained feature vector ψ (represented in purple) is also used in our UVDeltaNet. The symbol "$\sim$"refers to positional encoding as in NeRF [9] and "||" refers to feature-wise concatenation.

where $\mathbf{M}(\mathbf{r})$ is the binary mask at the pixel intersected by the ray $\mathbf{r}$. The coefficients $\lambda_{(.)}$ are defined empirically and set to $\lambda_{\text{L2}} = 100$ and λ_{vgg}=λ_{σ}=0.01 in all experiments. After training, we discard the RGB color output and only use the density σ to implicitly represent the geometry of the neural actor.

A.2 NormalNet Model

The goal of NormalNet in our approach is to perform inpainting and refinement of the normal values sampled from the implicit neural field and aggregated in the UV map. Therefore, we leverage partial convolutions [6] as a building block and design a shallow convolutional neural network (CNN) that is illustrated in Fig. 10.

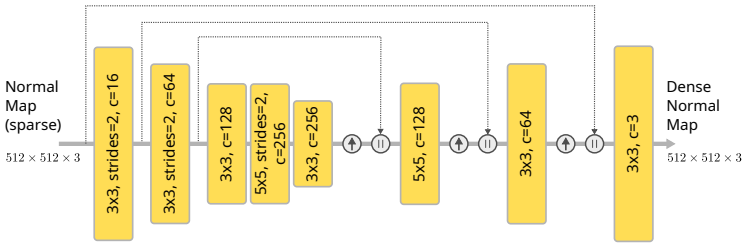

Fig. 10: Neural network architecture of our NormalNet model. Each rectangular block is a 2D partial convolution. The symbol "↑" refers to depth-to-space transformation, where the spatial resolution is increased by a factor of 2 in each dimension, and "||" refers to feature-wise concatenation.

A.3 VisibilityNet Model

We use a similar approach as in the NormalNet architecture for inpainting and refining the visibility information. However, the visibility is highly dependent on the normal values, since half of the light sources in the environment map are back lit as a function of the normal direction. Therefore, the VisibilityNet model also takes as input the sampled normal maps. The shallow CNN architecture of VisibilityNet is shown in Fig. 11.

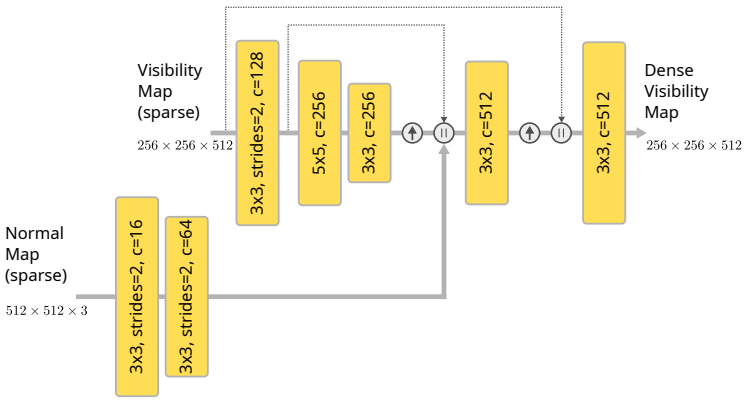

Fig. 11: Neural network architecture of our VisibilityNet model. Rectangular blocks represent 2D partial convolutions. "↑" and "||" refer to depth-to-space transformation and feature-wise concatenation, respectively.

A.4 UVDeltaNet Model

The architecture of UVDeltaNet is shown in Fig. 12. This shallow MLP has as objective predict a corrective term in the UV space, therefore it takes as input the local coordinates (u, v, h) and the texture features ψ (see Fig. 9), and predicts a corrective term (u', v').

B Dataset Collection

The only existing dataset for novel pose human relighting with ground-truth novel poses under new light conditions is proposed in RANA [4]. However, as can be seen in Fig. 13, the provided synthetic data presents severe image and geometry artifacts, despite being limited to a small number of frames from a monocular view of the person.

Therefore, we collected the new dataset "Relightable Dynamic Actors" with the goal of providing a real-world dataset with the same person recorded under different light conditions. Please refer to the main paper for samples from our

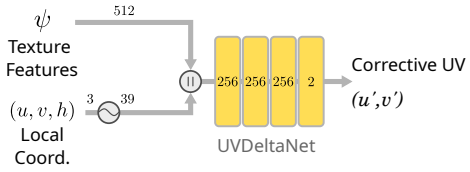

Fig. 12: Neural network architecture of our UVDeltaNet model. For predicting the corrective UV term, this model takes as input the local coordinates (u, v, h), corresponding to the 3D sampled point projected onto the human mesh surface, and the ResNet texture features ψ.

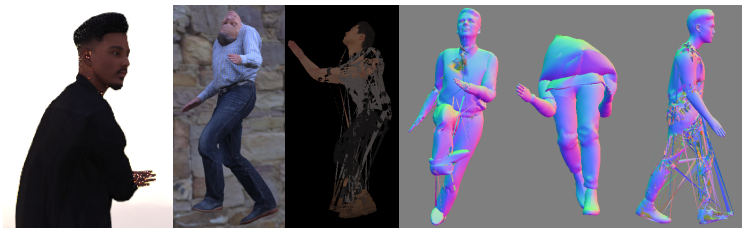

Fig. 13: Samples from the provided synthetic dataset in RANA [4]. Rendered images (left) present salt and pepper noise, as well as severe geometry artifacts (right).

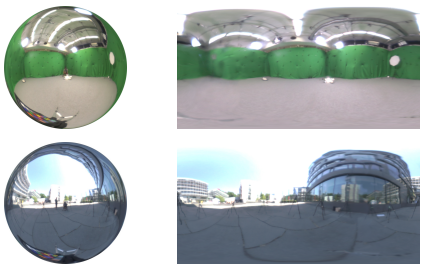

Fig. 14: Samples from our environment maps obtained from indoor and outdoor sequences. On the left is the light probe obtained with HDR pictures from a mirror sphere and on the right is the processed HDR images converted to a latitude-longitude format (EXR). The environment maps (right) are then resized to 32×16 pixels to be used in our method.

data. For each sequence in our dataset, the actors were instructed to casually follow a sequence of 10 activities defined as: *stretch arms*, *walk in a circle*, *jogging*, *stretch legs*, *talk on the phone*, *use a tablet*, *stretch legs up*, *stretch back*, *wave hands*, *freestyle*. We use a commercial markerless motion tracking system [10] for obtaining the human motion in 3D from the calibrated multi-view videos. To fit SMPL model [8] to our tracking, we first optimize the pose and shape parameters in the rest pose "T-pose" with a 3D body joint loss between SMPL and the tracked 3D pose based on EasyMocap [1], then track the SMPL model following the ground-truth motion.

We obtained the environment map for each sequence by taking multi-exposure pictures from a mirror sphere. The HDR image of the mirror sphere was obtained with the algorithm from Debevec et al. [3]. We combined two shots of the mirror sphere from different angles. The combined image was unwrapped to the EXR latitude-longitude format, resulting in the HDR environment map. Two sampled from our environment maps are shown in Fig. 14.

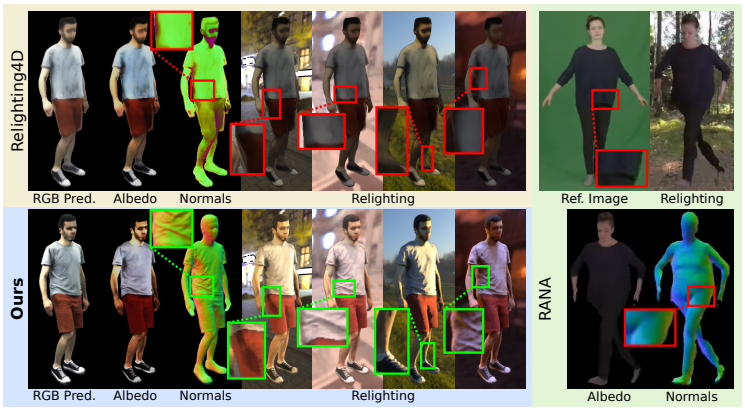

Fig. 15: Comparison between our method, Relighting4D [2], and RANA [4]. Note how Relighting4D and RANA fail to recover fine details on the surface and produce unrealistic results. Our results are much sharper and with realistic shading and colors under new challenging lights that produce strong cast shadows.

C Additional Results and Comparisons

Comparisons between the quality of our method and existing approaches are shown in Fig. 15. We further compare our method with Relighting4D [2], as shown in Tab. 1. Since Relighting4D can only replay the same sequence under new light conditions, we adapted it to render the virtual human under the new poses from our test sequences using the respective ground-truth environment maps. Relighting4D was trained using the same setup as our method, i.e., multi-view video sequence with all the training camera views.

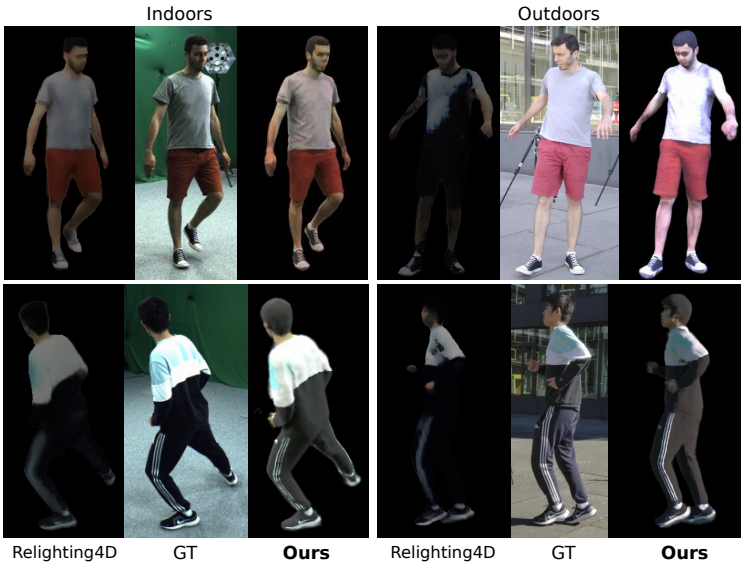

Fig. 16: Qualitative comparisons between Relighting4D [2] and our method. Our results capture more fine details such as clothe wrinkles and shadows, while Relighting4D produces blurry results with severe artifacts due to its inability to synthesize realistic renderings under novel poses.

Table 1: We compared our method with Relighting4D [2] on our real data, subject S1, considering indoor and outdoor scenes. Relighting4D was originally designed to replay the same training sequence. Therefore, we adapted it in this experiment to produce the new poses from our test sequences under new light conditions. Our approach consistently outperforms under novel poses, specially under outdoors illumination, which strongly differs from the training lights.

Method	T2			T3			T4			T5			T6 (outdoors)		
	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓	PSNR ↑	SSIM ↑	LPIPS ↓
Relighting4D [2]	16.108	0.743	0.183	15.728	0.737	0.171	16.969	0.748	0.172	16.759	0.739	0.166	8.492	0.600	0.296
Ours	**18.602**	**0.792**	**0.163**	**18.240**	**0.800**	**0.169**	**18.388**	**0.800**	**0.164**	**18.820**	**0.800**	**0.165**	**19.461**	**0.753**	**0.173**

In Fig. 16 we show qualitative comparisons between Relighting4D and our method. Note that Relighting4D is enable to produce realistic results under our settings, i.e., novel pose and novel light conditions, even when it is trained on the same setup. The superiority of our approach can be seem in its ability to model cloth wrinkles and shadows, resulting in more realist renderings.

References

1. Easymocap - make human motion capture easier. Github (2021), https://github.com/zju3dv/EasyMocap
2. Chen, Z., Liu, Z.: Relighting4d: Neural relightable human from videos. In: European Conference on Computer Vision (ECCV) (2022)
3. Debevec, P.E., Malik, J.: Recovering high dynamic range radiance maps from photographs. In: ACM SIGGRAPH 2008 classes, pp. 1–10 (2008)
4. Iqbal, U., Caliskan, A., Nagano, K., Khamis, S., Molchanov, P., Kautz, J.: Rana: Relightable articulated neural avatars. In: International Conference on Computer Vision (ICCV). pp. 23142–23153 (2023)
5. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: Computer Vision and Pattern Recognition (CVPR). pp. 4681–4690 (2017)
6. Liu, G., Reda, F.A., Shih, K.J., Wang, T.C., Tao, A., Catanzaro, B.: Image inpainting for irregular holes using partial convolutions. In: European Conference on Computer Vision (ECCV) (2018)
7. Liu, L., Habermann, M., Rudnev, V., Sarkar, K., Gu, J., Theobalt, C.: Neural actor: Neural free-view synthesis of human actors with pose control. ACM Transactions on Graphics (TOG) **40**(6), 1–16 (2021)
8. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: SMPL: A skinned multi-person linear model. ACM Transactions on Graphics (SIGGRAPH Asia) **34**(6), 248:1–248:16 (2015)
9. Mildenhall, B., Srinivasan, P.P., Tancik, M., Barron, J.T., Ramamoorthi, R., Ng, R.: Nerf: Representing scenes as neural radiance fields for view synthesis. In: European Conference on Computer Vision (ECCV) (2020)
10. TheCaptury: The Captury. http://www.thecaptury.com/ (2020)