DNF-Avatar: Distilling Neural Fields for Real-time Animatable Avatar Relighting

Supplementary Material

In this **supplementary document**, we provide additional materials to supplement our main submission. The **code** is available here for research purposes: github.com/jzr99/DNF-Avatar

6. Implementation Details

6.1. Final Objectives

In addition to the losses introduced in our manuscript, we also adapt the following loss during distillation. The final loss is a linear combination of the losses with the corresponding weights.

Material Smoothness Loss. We regularize the intrinsic properties $\{r, m, \mathbf{a}\}$ via a bilateral smoothness term[15], which prevents the material properties from changing drastically in areas with smooth colors:

$$\mathcal{L}_{\text{smooth}} = \|\nabla \mathbf{I}^s(\mathcal{R}, *)\| \exp\left(-\left\|\nabla \mathbf{I}^{gt}_{rgb}\right\|\right), \tag{22}$$

where $\mathbf{I}^s(\mathcal{R}, *)$ are rasterized material maps. $*$ denotes $\{r, m, \mathbf{a}\}$. $\mathbf{I}^{gt}_{rgb}$ represents ground truth images.

Anisotropy Regularization Loss. We adopt the loss from [74] for 2DGS:

$$\mathcal{L}_{\text{aniso}} = \frac{1}{N}\sum_{i=1}^{N} \max\left\{\max\left(\mathbf{s}_i^s\right) / \min\left(\mathbf{s}_i^s\right), r\right\} - r, \tag{23}$$

where $\mathbf{s}_i^s$ is the scaling of 2DGS. This loss constrains the ratio between the length of two axes of 2DGS that to not exceed predefined value r. We set $r = 3$ to prevent the Gaussian primitives from becoming threadlike, which alleviates the geometric artifacts under novel poses.

Normal Orientation Loss. Ideally, normals of visible 2D Gaussian primitives should always face toward the camera. To enforce this, we employ the normal orientation loss [67]:

$$L_{orient} = \frac{1}{|\mathcal{R}|}\sum_{\boldsymbol{r}\in\mathcal{R}} \left\|\max\left(-\boldsymbol{\omega}_{o,r}\cdot \mathbf{I}^s(\mathbf{r}, \mathbf{n}_o^s), 0\right)\right\|_1, \tag{24}$$

where $\boldsymbol{\omega}_{o,r}$ denotes the outgoing light direction (surface to camera) for ray $\mathbf{r}$. $\mathbf{I}^s(\mathbf{r}, \mathbf{n}_o^s)$ denotes the rasterized world-space normal for ray $\mathbf{r}$.

Environment Map Distillation Loss. In addition to the distillation loss between the two avatar representations, we also regularize the environment map of our student model with the one of our teacher model:

$$L_{distill}^{env} = \frac{1}{|\mathcal{S}^2|}\sum_{\boldsymbol{\omega}\in\mathcal{S}^2} \left\|L_e^t(\boldsymbol{\omega}) - L_e^s(\boldsymbol{\omega})\right\|_2, \tag{25}$$

where L_e^t denotes a spherical-gaussian-based environment map from our teacher model, and L_e^s represents a cubemap-based environment map from our student model. $\mathcal{S}^2$ is all possible lighting directions.

Depth Distortion and Normal Consistency. Following 2DGS[23], we apply the depth distortion loss and normal consistency loss to concentrate the weight distribution along the rays and make the 2D splats locally align with the actual surfaces:

$$L_{dist} = \frac{1}{|\mathcal{R}|}\sum_{\mathbf{r}\in\mathcal{R}}\sum_{i,j}^{N} w_i(\mathbf{r}) w_j(\mathbf{r}) \left\|z_i(\mathbf{r}) - z_j(\mathbf{r})\right\|_1, \tag{26}$$

$$L_{nc} = \frac{1}{|\mathcal{R}|}\sum_{\mathbf{r}\in\mathcal{R}}\sum_{i}^{N} w_i(\mathbf{r})(1 - \mathbf{n}_i^{\intercal}\mathbf{N}(\mathbf{r})), \tag{27}$$

where $w_i(\mathbf{r}) = o_i\hat{\mathcal{G}}_i(\mathbf{u}(\mathbf{r}))\prod_{j=1}^{i-1}\left(1 - o_j\hat{\mathcal{G}}_j(\mathbf{u}(\mathbf{r}))\right)$ is the blending weight for ith 2D splat along the ray $\mathbf{r}$, and z_i is the depth of the intersection point. $\mathbf{N}$ is the normal derived from the depth map.

6.2. Training Details

The teacher model is trained first and then frozen during distillation. We apply the marching cube algorithm to extract the mesh from the implicit teacher model and initialize the 2DGS with a sampled subset from the vertexes of the mesh. Similar to [82], during distillation, we periodically densify and prune the 2DGS with the initial sampled vertex to regularize the density of the 2DGS. Following IA [71], we employ a two-stage training strategy during distillation. We train a total of 30k iterations with distillation loss applied. We apply a color MLP [57] to estimate the radiance in the first 20k iterations, while we employ both color MLP and PBR rendering loss for the rest of the iterations. Note that the color MLP is only used during training, which helps regularize the geometry of the Gaussians. As for the pre-computation of occlusion probes, we separate the human avatar into 9 parts based on the skinning weights, and pre-compute the part-wise occlusion probes after the first 20k iterations.

During rendering, we adopt the standard gamma correction to the rendered image from linear RGB space to sRGB space and then clip it to [0, 1]. To stay consistent with R4D [9] and IA [71], we calibrate our albedo prediction to the range [0.03, 0.8], which prevents the model from predicting zero albedo for near-black clothes.

7. Additional Experimental Results

7.1. Metrics

For synthetic datasets, we assess several metrics:

Relighting PSNR/SSIM/LPIPS: We evaluate standard image quality metrics for images rendered under novel poses and illumination conditions.

FPS: We report the rendering frame rate per second for the 540×540 resolution images on a single NVIDIA RTX 4090 GPU.

Normal Error: This metric measures the error (in degrees) between the predicted normal images and the ground-truth normal images.

Albedo PSNR/SSIM/LPIPS: We use standard image quality metrics to evaluate albedos rendered from training views. Since there is inherent ambiguity between the estimated albedo and light intensity, we align the predicted albedo with the ground truth, following [85].

For real-world datasets, *i.e*. PeopleSnapshot, we provide qualitative results, showcasing novel views and pose synthesis under new lighting conditions.

7.2. Additional Qualitative Results

We show additional qualitative relighting results on the PeopleSnapshot dataset in Fig. 9. All of the subjects are rendered under novel poses and novel illuminations.

7.3. Additional Quantitative Results

The per-subject and average metrics of R4D, IA, Ours-D, and Ours-F are reported in Tab. 6. Note that the only difference between Ours-D and Ours-F is in the inference stage, so they share the same intrinsic properties.

7.4. Additional Ablation Study for Distillation

As shown in Tab. 4, we ablate the proposed distillation objectives on subject 01 of the RANA dataset. dist., i-dist., and p-dist. represent distillation, image-based distillation, and point-based distillation, respectively. When distillation is disabled, 2DGS itself cannot produce satisfying geometry, leading to poor relighting results. While image-based distillation successfully distills the knowledge from the training view, point-based distillation further improves the performance by distilling knowledge in both visible and occluded areas. We also note that the bias from the implicit teach model (smooth interpolation of density and color in regions not seen during training) helps reducing artifacts in our student model. We compare our model with a pure explicit 3DGS-based avatar model [57] and show that such explicit representation struggles to generalize to out-of-distribution joint angles, while our model achieves reasonable results, thanks to the smoothness bias distilled from the teacher model (Fig. 7).

Method	**Normal** ↓	**Relighting**		
		PSNR ↑	**SSIM** ↑	**LPIPS** ↓
R4D [9]	33.61 °	18.22	0.8425	0.1612
IA [71]	12.05 °	18.48	0.8859	**0.1219**
w/o dist.	16.49 °	18.99	0.8739	0.1488
w/o i-dist.	14.55 °	19.30	**0.8889**	0.1392
w/o p-dist.	11.56 °	19.42	0.8835	0.1374
w/o dist. avatar	<u>11.50</u> °	<u>19.47</u>	0.8878	<u>0.1332</u>
Ours	**11.41** °	**19.48**	<u>0.8884</u>	**0.1315**

Table 4. **Quantitative Ablation Studies on RANA.** Both objectives for distillation effectively contribute to the final relighting quality.

7.5. Rendering Speed

Method	**LBS**	**Occ.**	**Shading**	**Rast.**	**Total**
Ours-D	3.3ms	7.7ms	12.1ms	6.9ms	30.0ms
Ours-F	3.3ms	7.7ms	0.9ms	2.9ms	14.8ms

Table 5. **Time cost for each part of our model.**

As shown in Tab. 5, we test the performance for each component of our PBR pipeline. The test is done with a 540×540 resolution using around 70000 Gaussian primitives. The deferred shading version is bounded by the shading time, which scales linearly with the number of pixels. In comparison, for forward shading, the shading module itself is very fast, while querying part-wise occlusion probes becomes the bottleneck of performance. The bottleneck of part-wise occlusion probes is governed by the number of Gaussian primitives. In addition, we assume the environment map remains unchanged for a single animation sequence so that the precomputation time (around 10ms per environment map) for the Equ. (10) and Equ. (11) is ignored. However, our forward shading pipeline can still achieve around 40 FPS, even if we take this precomputation into account.

8. Additional Discussion

8.1. Clarification for Part-wise Occlusion

Ambient occlusion is calculated on the fly during animation. The idea is that each body part is rigid, and thus we can pre-compute its occlusion probes in canonical space of each part. The pre-convolution (Eq. (9)) ensures that a single query is sufficient to obtain ambient occlusion at test time. For a single Gaussian in observation space, we transform it to the canonical space of N_p body parts according to per-part rigid transforms. By querying partwise ambient occlusion probes in canonical space, we obtain N_p ambient occlusion values, and the product of these values is the final ambient occlusion (Eq. (16)). We also present Fig. 8, where intra-part occlusion (calculated when Gaussian was transformed to the canonical space of its own part) captures

Figure 7. **Implicit bias helps pose generalization.** Under limited training pose variation, the bias imposed by our implicit teacher model helps our student model to achieve reasonable rendering on out-of-distribution poses (left). In comparison, the state-of-the-art 3DGS-based avatar model [57] tends to fail on out-of-distribution poses, especially around joints (right).

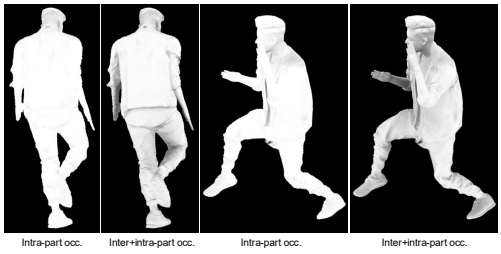

Figure 8. Visualization of intra/inter-part occlusion.

the shadow of the wrinkle and local geometry, while inter-part occlusion (calculated when Gaussian was transformed to the canonical space of other parts) models *e.g.*, shadow cast by the body onto the inner side of the arm. Posed body geometry can be used with Monte Carlo (MC) ray-tracing to compute shadows but it's not real-time. Our pre-computed part-wise occlusion probes avoid MC ray-tracing, enabling real-time rendering.

8.2. Worse Albedo but Better Relighting Results

The ground-truth dataset and the teacher model both employ MC ray tracing. If we use a large loss to enforce the albedo from the teacher to the student, the final relighting results will be suboptimal since split-sum is an approximation to ray-tracing. Hence, we use a small weight for albedo distillation, which serves more like a regularization to make sure the student does not produce unreasonable albedos. Our albedos are thus optimized for split-sum and are not consistent with the ground-truth, which employs ray-tracing. On the other hand, our learned normals are less noisy than the teacher, while split-sum does not suffer from MC noise. These factors combined give us a better relighting result.

9. Limitations and Societal Impact Discussion

The final quality of our approach largely depends on the stability of the teacher model. Currently, the teacher model [71] requires accurate body pose estimation and foreground segmentation, which may not be the case for in-the-wild captures. Combining existing state-of-the-art in-the-wild avatar models [17, 29, 43] with our efficient relightable model is an interesting direction for future work.

Furthermore, the ambient occlusion assumption in our method may not hold in the presence of strong point lights. In such cases, the shading model may not be able to capture the correct shadowing effects. Also, similar to other state-of-the-art models [42, 71, 76], our model can only handle direct illumination at inference time. Modeling global illumination effects while still achieving real-time performance is an active area of research in both computer graphics and computer vision.

Moreover, our current per-scene optimization-based pipeline remains slow during training. Similar to other state-of-the-art feed-forward dynamic reconstruction methods [30, 35, 73], a promising future direction is to learn a data-driven prior for intrinsic property decomposition, enabling a feed-forward approach for animatable and relightable avatar reconstruction.

Regarding the societal impact, our work can be used to create realistic avatars for virtual reality, gaming, and social media. However, it is important to consider the ethical implications of using such technology. For example, our method can be used to create deepfakes, which can be used to spread misinformation. It is important to develop methods to detect deepfakes and educate the public about the existence of such technology.

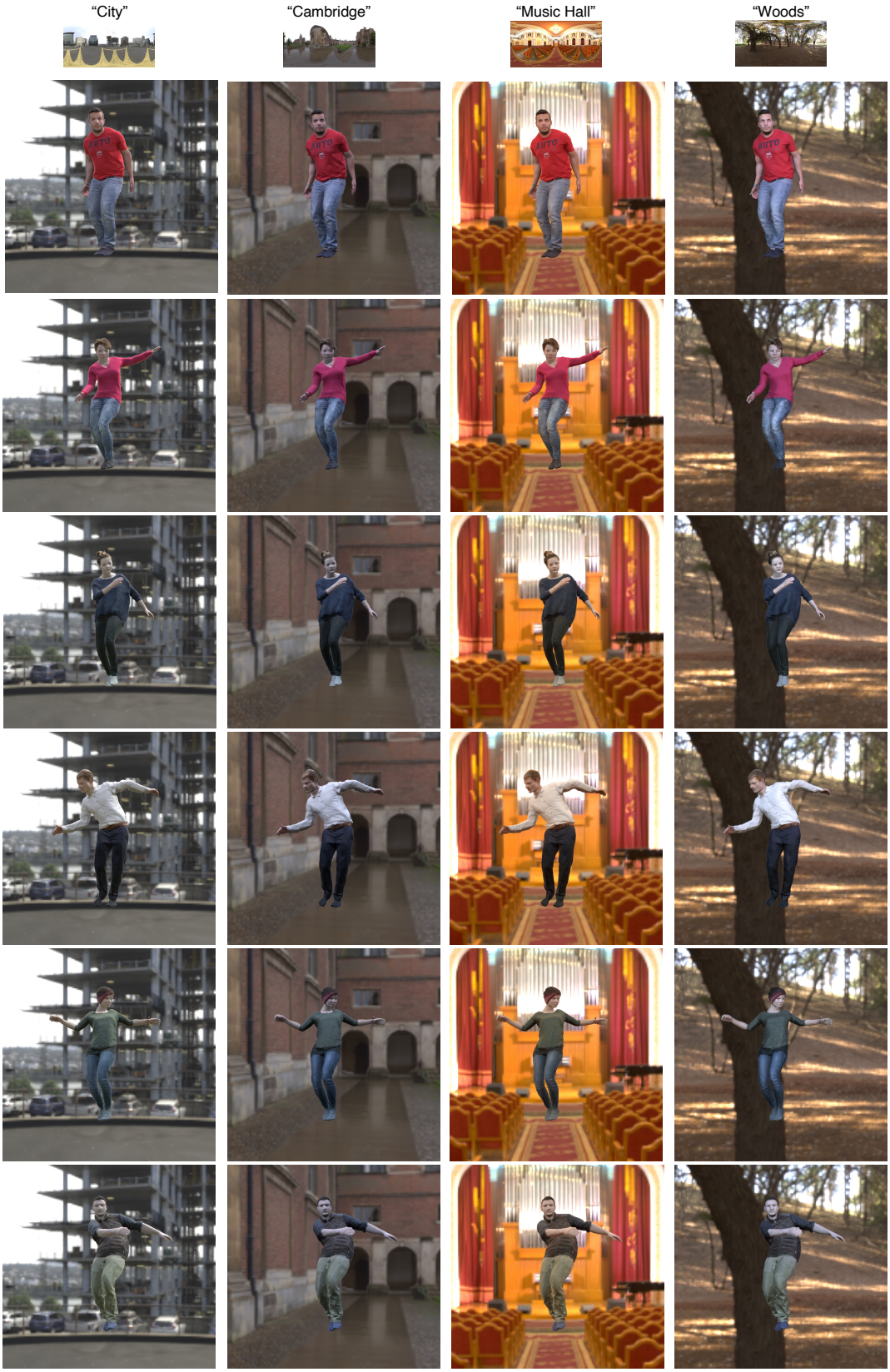

Figure 9. **Qualitative Relighting on PeopleSnapshot Dataset.**

Subject	Method	Albedo			Normal	Relighting (Novel Pose)		
		PSNR ↑	SSIM ↑	LPIPS ↓	Error ↓	PSNR ↑	SSIM ↑	LPIPS ↓
Subject 01	R4D	20.04	0.8525	0.2079	33.61 °	18.22	0.8425	0.1612
	IA	24.11	0.8679	0.1827	12.05 °	18.48	0.8859	0.1219
	Ours-D	23.90	0.8580	0.1834	11.41 °	19.42	0.8905	0.1252
	Ours-F					19.48	0.8884	0.1315
Subject 02	R4D	12.13	0.7690	0.2599	28.34 °	14.38	0.8128	0.1787
	IA	20.94	0.8892	0.1854	9.29 °	19.08	0.8812	0.1323
	Ours-D	20.76	0.8773	0.1675	9.04 °	19.86	0.8875	0.1285
	Ours-F					20.03	0.8891	0.1297
Subject 05	R4D	19.74	0.8151	0.2488	26.14 °	17.72	0.8469	0.1780
	IA	22.24	0.8591	0.2071	9.52 °	17.47	0.8769	0.1453
	Ours-D	22.26	0.8527	0.1798	9.07 °	18.89	0.8876	0.1377
	Ours-F					18.97	0.8873	0.1411
Subject 06	R4D	21.57	0.7992	0.2177	25.83 °	17.54	0.8866	0.1636
	IA	22.94	0.8233	0.1928	8.89 °	18.14	0.8932	0.1271
	Ours-D	22.91	0.8163	0.1752	9.03 °	18.67	0.8960	0.1289
	Ours-F					18.72	0.8953	0.1341
Subject 33	R4D	18.35	0.8426	0.1887	25.24 °	16.78	0.8173	0.1859
	IA	21.67	0.8703	0.1351	9.52 °	18.03	0.8426	0.1366
	Ours-D	21.18	0.8450	0.1544	8.92 °	19.13	0.8546	0.1331
	Ours-F					19.23	0.8557	0.1332
Subject 36	R4D	23.80	0.9100	0.1611	24.76 °	17.05	0.8574	0.1707
	IA	24.88	0.8900	0.1324	9.22 °	17.46	0.8726	0.1284
	Ours-D	24.43	0.8785	0.1384	9.27 °	18.18	0.8764	0.1293
	Ours-F					18.26	0.8773	0.1389
Subject 46	R4D	18.13	0.8777	0.1238	33.27 °	16.30	0.8338	0.1649
	IA	22.47	0.9391	0.0725	10.69 °	17.08	0.8406	0.1000
	Ours-D	22.36	0.9298	0.0793	10.25 °	17.47	0.8415	0.1039
	Ours-F					17.62	0.8426	0.1041
Subject 48	R4D	12.10	0.7370	0.2264	21.84 °	14.98	0.7985	0.1776
	IA	23.36	0.9137	0.1857	10.49 °	19.70	0.8849	0.1313
	Ours-D	23.39	0.9034	0.1707	9.62 °	19.82	0.8808	0.1329
	Ours-F					19.97	0.8816	0.1328
Average	R4D*	18.23	0.8254	0.2043	27.38 °	16.62	0.8370	0.1726
	IA	**22.83**	**0.8816**	0.1617	9.96 °	18.18	0.8722	0.1279
	Ours-D	22.65	0.8701	**0.1561**	**9.58** °	18.93	0.8769	**0.1275**
	Ours-F					**19.04**	**0.8772**	0.1307

Table 6. **Per-Subject Metrics on the RANA dataset.**