In this supplementary material, we begin by presenting the implementation details of our method by first explaining about the use of Multi-Resolution Hash Encoder as Displacement Encoder and Spherical Harmonics Encoder in detail in §1.1. We then provide more mathematical details on how we bind 3D Gaussians to mesh surface in §1.2. Our stage wise training strategies are explained in §1.3.

We present more qualitative restuls and comparision on PeopleSnapshot [1] and in-the-wild UBC-Fashion [12] dataset in §2. Then, we show challenging cases in §3. Finally, we show use of GART as template model for iHuman in §4 that handles clothing deformation better.

1 Implementation Details

1.1 Hash Encoding

A single scene reconstructed using 3D gaussian splatting contains a very large number of gaussians. Backpropagating the gradients to every gaussians in every iteration results in poor performance due to which we need an efficient sampling strategy is needed to select only a few gaussians to be updated every iteration. Inspired by [9] we encode the color and displacement for each gaussian using a multi-resolution hash encoder.

The color and displacement encoder include a hash encoder followed by a fully-fused MLPs implemented using the tiny-cuda-nn framework [10]. The implemented hash encoder has 16 levels, hash table size of 2^{17} with 4 features per entry in the table with a base resolution of 4 and resolution growth factor of 1.5. The fully fused MLPs consist of 2 hidden layer with 64 neurons per layer and uses relu as the activation function. Both the color and displacement encoder take the gaussian position as input and produce a color and displacement value of the same dimension as the input.

1.2 Binding Gaussian To Mesh Surface

As we explained in Section 3.3, we bind Gaussians at the centroid of the triangular face. Given face i_x, the centroid is given by:

$$x = \frac{i_x[1] + i_x[2] + i_x[3]}{3}. \tag{1}$$

Through this equation, we always maintain the position of Gaussian at the centroid of the face i_x. And, we can directly optimize the vertices V_c of canonical mesh $\mathcal{M} = (V_c, F)$.

Similar to SuGaR [5], we parameterize the 3D rotation of the Gaussians with only 2 parameters by encoding the rotation in complex 2D rotation form with $(x + iy)$. We limit their rotation to local 2D triangular face plane. We now explain how we convert the local 2D complex rotation to 3D rotation required for Gaussian Rasterizer.

Consider a set of face normals $\mathbf{n}_i$ for each face i of the mesh. We first normalize these normals to obtain a primary direction vector $\mathbf{R}_0$:

$$\mathbf{R}_0 = \frac{\mathbf{n}_i}{\|\mathbf{n}_i\|}. \tag{2}$$

The second direction vector $\mathbf{R}_1$ is calculated using the difference between the first two vertices of the triangle, yielding the edge vector $\mathbf{e}_{12}$, which is then normalized:

$$\mathbf{R}_1 = \frac{\mathbf{e}_{12}}{\|\mathbf{e}_{12}\|}. \tag{3}$$

The third direction vector $\mathbf{R}_2$ is produced by the cross product of $\mathbf{R}_0$ and $\mathbf{R}_1$, ensuring orthogonality:

$$\mathbf{R}_2 = \frac{\mathbf{R}_0 \times \mathbf{R}_1}{\|\mathbf{R}_0 \times \mathbf{R}_1\|}. \tag{4}$$

To incorporate the parameterized 2D rotation defined by complex number of 2 parameters for each Gaussians, we apply them to the base vectors $\mathbf{R}_1$ and $\mathbf{R}_2$ to obtain the rotated axes.

Finally, the rotation matrix $\mathbf{R}$ for each Gaussian distribution is constructed by combining the normalized primary, secondary, and tertiary direction vectors:

$$\mathbf{R} = [\mathbf{R}_0, \mathbf{R}_1, \mathbf{R}_2]. \tag{5}$$

These rotation matrices $\mathbf{R}$ are used to orient the Gaussians so that they follow the orientation of the mesh surface.

As scaling parameter S, we use:

$$S = (s_1, s_2, s_3) \tag{6}$$

where, $s_1 = \epsilon$, s_2 and s_3 are learnable parameters. s_1 corresponds with the normal vector. In practice, we set $\epsilon = 1mm$.

1.3 Training

For the PeopleSnapshot [1] and Multi-Garment Dataset [3], we use the same hyper parameter across all the subjects. We use Adam [7] optimizer for parameter optimization. Each 3D Gaussian is defined by a center (x), scale (S), opacity (α), rotation (q), spherical harmonics (SH) and blend weights (w). We don't optimize α, q and w. We keep $\alpha = 1$. We optimize the joints position (J) of the canonical skeleton. For learning good geometric details, the number of 3D Gaussians should be enough to model the geometry. We use template mesh of about 220K triangular faces and same number of 3D Gaussians are initialized for PeopleSnapshot and Multi Garment Dataset.

We divide the training into three stages where we prioritize learning geometric information in the earlier stage and color information in the later stages. The first stage lasts till 4^{th} epoch, then the second stage starts from 4^{th} epoch and

ends at the 10^{th} epoch and the final stage starts from 11^{th} and lasts till the 20^{th} epoch. We use the same scale learning rate of 5e-3 and SH encoder learning rate of 5e-4 throughout the training process. We use a joints learning rate of 5e-4 for the first and second stage and do not optimize the joints location in the final stage. We use learning rate for displacment encoder of 1e-4 in the first and final stage and 8e-4 in the second stage. We keep the normal and photometric loss weight of 1 and normal consistency loss weight of 0.01 throughout the training process. For UBC-Fashion Dataset, we add a pose optimization step where we also optimize the body pose, global orientation and translation parameters of the SMPL body model by using a common learning rate of 1e-4 for all three parameters.

To obtain the ground truth normal, we use pretrained pix2pixHD network by PIFuHD [11]. In Nvidia RTX 4090, normal maps can be obtained for 20 images in less than a second. So, getting normal-map doesn't add any significant time bottleneck in data pre-processing step.

2 Additional Qualitative and Quantitative Results

2.1 Qualitative Results

3D Mesh Reconstruction. We show 3D mesh reconstruction of GART [8], Anim-NeRF [4] and our method on our synthetic Multi-Garment Dataset in Fig. 1. We show more qualitative results of mesh reconstruction of the proposed method on PeopleSnapshot in Fig. 2.

In Fig. 4, we show reconstructed normal map image. The quality of normal map image being close to the ground truth normal map also shows good quality and fidelity of the reconstructed mesh.

We show some qualitative results of 3D Mesh reconstruction on UBC-Fashion dataset in Fig. 3.

Novel View Synthesis. Compared to other SoTA methods, our method achieves novel view with less artifacts in less time and less number of input sequences 5.

As shown in Fig. 6, iHuman achieves good quality novel view synthesis even being trained with only 6 number of views.

We show novel pose synthesis results in Fig. 7 on PeopleSnapshot [1] of the subjects trained with only 20 input views.

2.2 Quantitative Results.

Instant Avatar with Test Time Pose Optimization

For fair comparison, Table 2 on the main paper shows metrics without test time pose optimization. To represent the paper as faithfully as possible, we report InstantAvatar [6] metrics on PeopleSnapshot with test time pose optimization in Table 1 . For 20 views using test time pose optimization the metrics improves but still for 6 views and 12 views, InstantAvatar [6] struggles to converge compared to our method and GART [8].

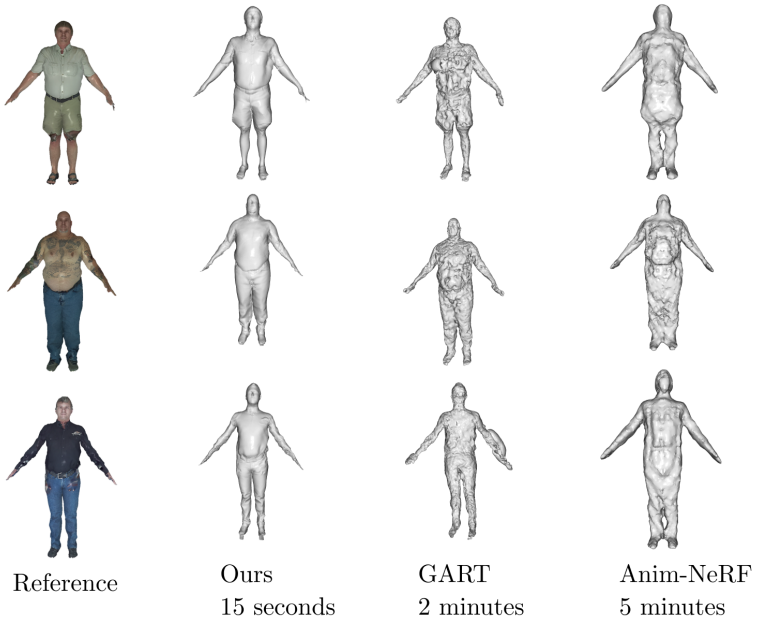

Fig. 1: Visualization of 3D reconstruction on Multi Garment. Mesh from GART [8] and Anim-NeRF [4] suffers from heavy artifacts while our method produces high fidelity mesh. Better surface reconstruction than another 3D Gaussian based method, GART can be attributed to our mesh binding and explicitly computed normal map optimization.

3 Challenging Cases

UBC-Fashion dataset [12] contains subjects in long clothing that undergoes deformation. As shown in Fig. 8, with heavy clothing deformation, though the reconstructed view looks good, there are some geometric implausible views. Even with heavy deforming scene, the reconstructed mesh doesn't contain floating artifacts.

4 GART as Template Model in iHuman Pipeline

One limitations with SMPL template is modeling some clothing topology. Better modeling of deforming clothes require better template models than SMPL. Indeed, we use the blending weights, joints and, vertices and trianlges from SMPL and create our own deformer that can be replaced with any forward LBS based template model. iHuman is modular and the SMPL template can be *replaced trivially* by another which supports i) forward skinning and ii) mesh model. In

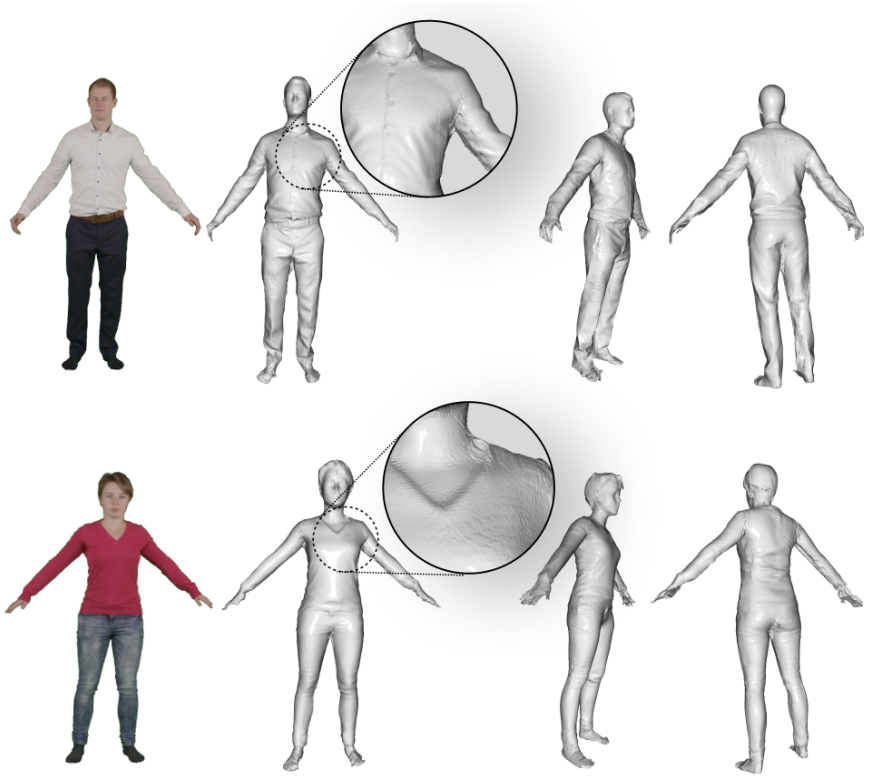

Fig. 2: More 3D reconstruction results of our method on PeopleSnapshot [2]. Our method accurately reconstructs surface details like shirt collar and cloths wrinkles.

Fig. 9, we replace SMPL with GART template and obtain better loose clothing deformation. So, our method can benefit from better templates while we focus more on obtaining mesh and better surface geometry from 3DGS. Note that we use 40K to 200K vertices, and recommend over 200K vertices for loose clothing.

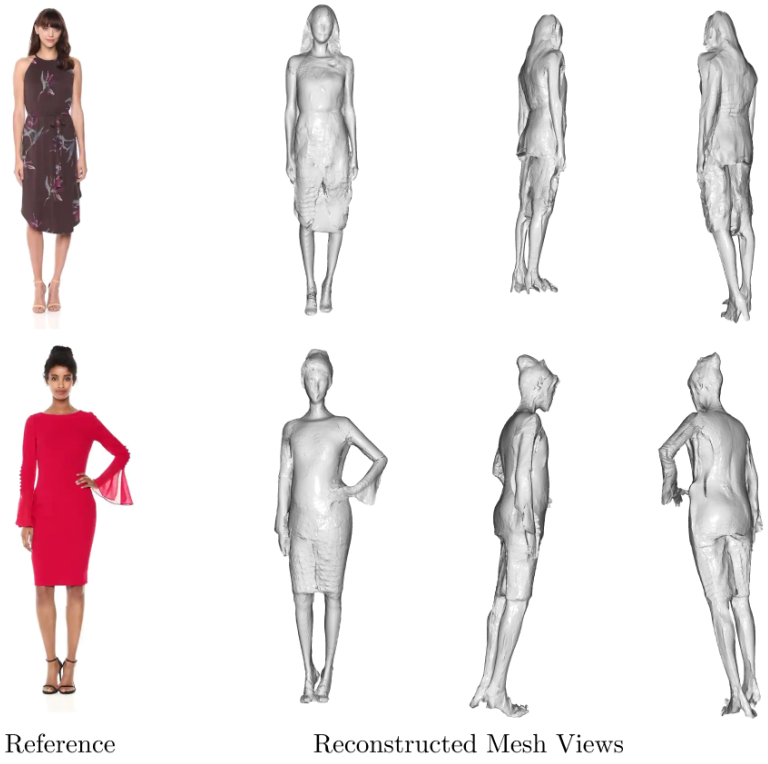

Fig. 3: 3D Reconstruction on UBC-Fashion [12].

Methods	male-3			male-4			female-3			female-4		
	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS	PSNR	SSIM	LPIPS
Instant Avatar(6 views)	15.24	0.8279	0.2624	16.51	0.803	0.3138	16.30	0.8316	0.2736	22.44	0.9292	0.0906
Instant Avatar(12 views)	23.10	0.9328	0.0931	17.63	0.8227	0.2769	24.08	0.9477	0.0693	19.73	0.8912	0.1544
Instant Avatar(20 views)	27.95	0.9604	0.0316	25.99	0.9484	0.0537	25.19	0.9511	0.0715	28.39	0.9619	0.0209

Table 1: Result of Novel View Synthesis of Instant Avatar with test time pose optimization on PeopleSnapshot [2] dataset. We report PSNR, SSIM, and LPIPS of InstantAvatar [6] with computational budget of maximum 5 minutes along with test time pose optimization.

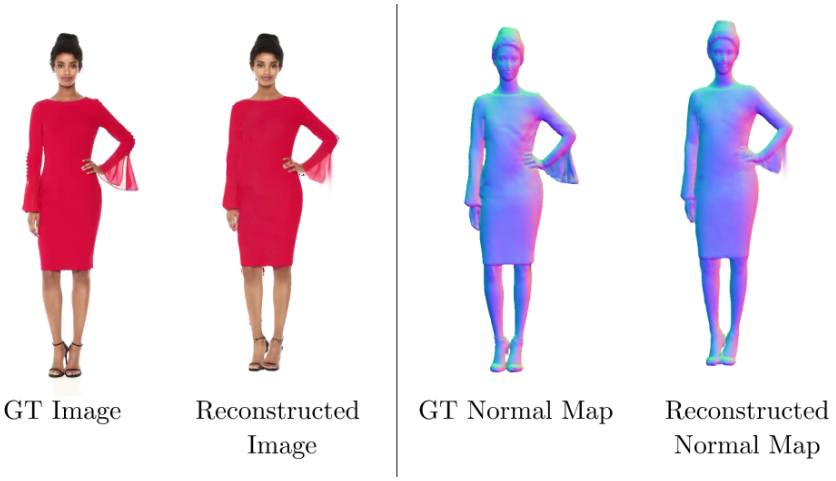

Fig. 4: View Reconstruction and Normal Map Visualization on UBC-Fashion [12]. (Left) View reconstruction on one subject of UBC-Fashion. (Right) GT normal map and reconstructed normal map visualization.

Fig. 5: Qualitative Comparison of performance of Instant Avatar [6], Anim-NeRF [4], GART [8] and Ours for different number of views in X pose. Instant Avatar and Anim-NeRF fails to reconstruct the subject whereas GART has artifacts around the arms and between legs. Our method is robust even for only 6 input sequences.

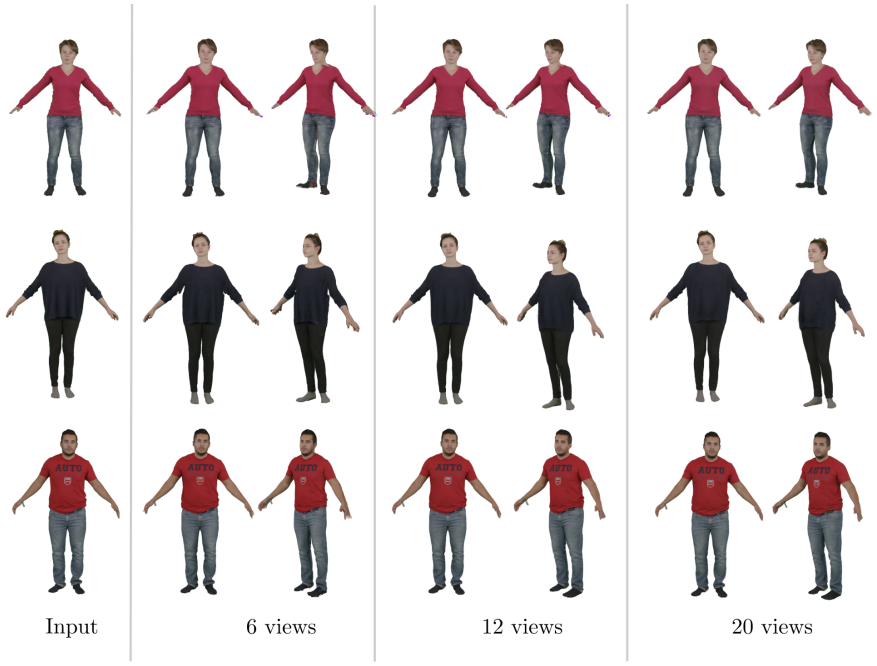

Fig. 6: Qualitative results on View Synthesis on People Snapshot [1] on different number of input sequences. Even with only 6 input views, our method achieves good quality view reconstruction.

Fig. 7: Novel Pose Synthesis on PeopleSnapshot [1]. We can feed the SMPL input pose to our iHuman Template Model to synthesize novel poses. Though trained on only A-pose, our method stably renders new image under complex poses.

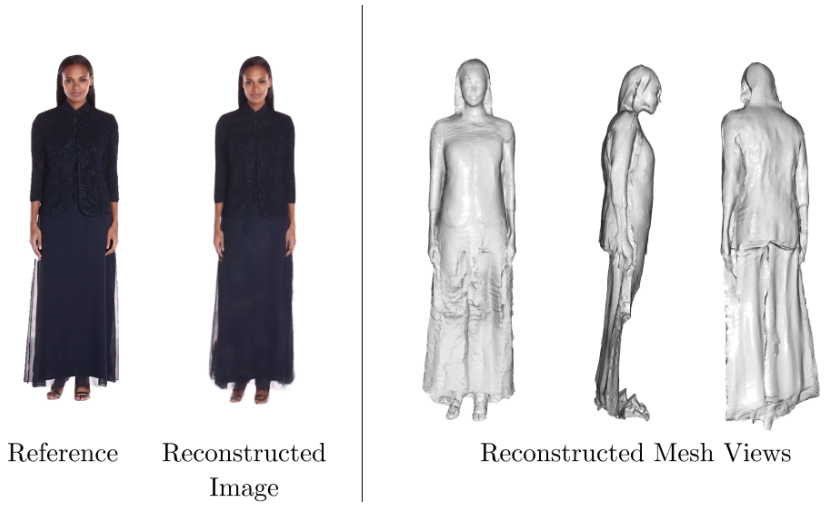

Fig. 8: Challenging Case. Under heavy clothing with deformation, the view reconstruction is shown (Left). 3D Mesh reconstruction is implausible for some view direction (Right).

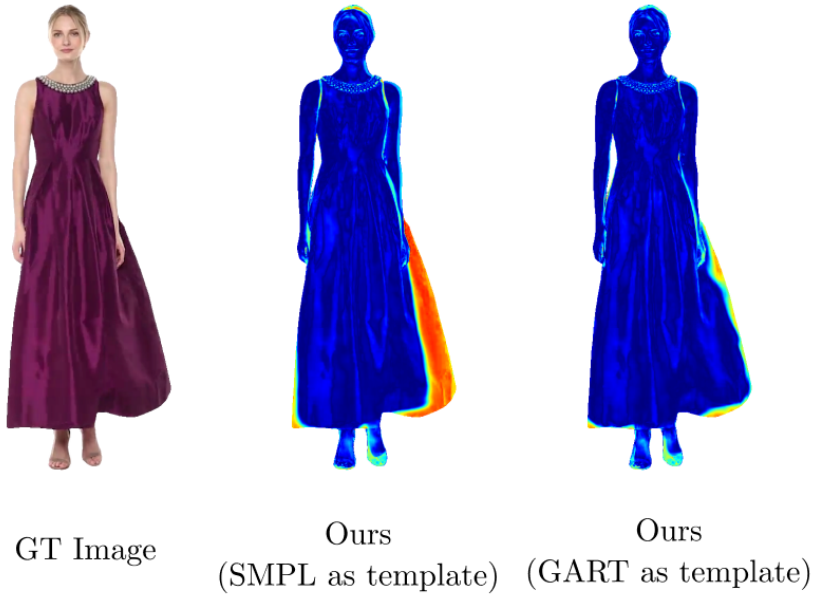

Fig. 9: Reconstruction accuracy: GART vs. SMPL as iHuman's template for loose clothing.

References

1. Alldieck, T., Magnor, M., Xu, W., Theobalt, C., Pons-Moll, G.: Detailed human avatars from monocular video. In: 2018 International Conference on 3D Vision (3DV). pp. 98–109. IEEE (2018) 1, 2, 3, 8
2. Alldieck, T., Magnor, M.A., Xu, W., Theobalt, C., Pons-Moll, G.: Video based reconstruction of 3d people models. In: 2018 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2018, Salt Lake City, UT, USA, June 18-22, 2018. pp. 8387–8397. Computer Vision Foundation / IEEE Computer Society (2018). https://doi.org/10.1109/CVPR.2018.00875, http://openaccess.thecvf.com/content_cvpr_2018/html/Alldieck_Video_Based_Reconstruction_CVPR_2018_paper.html 5, 6
3. Bhatnagar, B.L., Tiwari, G., Theobalt, C., Pons-Moll, G.: Multi-garment net: Learning to dress 3d people from images. In: Proceedings of the IEEE/CVF international conference on computer vision. pp. 5420–5430 (2019) 2
4. Chen, J., Zhang, Y., Kang, D., Zhe, X., Bao, L., Jia, X., Lu, H.: Animatable neural radiance fields from monocular rgb videos. arXiv preprint arXiv:2106.13629 (2021) 3, 4, 7
5. Guédon, A., Lepetit, V.: Sugar: Surface-aligned gaussian splatting for efficient 3d mesh reconstruction and high-quality mesh rendering (2023) 1
6. Jiang, T., Chen, X., Song, J., Hilliges, O.: Instantavatar: Learning avatars from monocular video in 60 seconds. CVPR (2023) 3, 6, 7
7. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization (2015) 2
8. Lei, J., Wang, Y., Pavlakos, G., Liu, L., Daniilidis, K.: GART: Gaussian Articulated Template Models (Nov 2023), http://arxiv.org/abs/2311.16099, arXiv:2311.16099 [cs] 3, 4, 7
9. Liu, Y., Huang, X., Qin, M., Lin, Q., Wang, H.: Animatable 3d gaussian: Fast and high-quality reconstruction of multiple human avatars. arXiv preprint arXiv:2311.16482 (2023) 1
10. Müller, T.: tiny-cuda-nn (4 2021), https://github.com/NVlabs/tiny-cuda-nn 1
11. Saito, S., Simon, T., Saragih, J., Joo, H.: Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. CVPR (2020) 3
12. Zablotskaia, P., Siarohin, A., Zhao, B., Sigal, L.: Dwnet: Dense warp-based network for pose-guided human video generation. arXiv preprint arXiv:1910.09139 (2019) 1, 4, 6, 7