EVA-Gaussian: 3D Gaussian-based Real-time Human Novel View Synthesis under Diverse Multi-view Camera Settings

A. More Implementation Details

Network architectures. Our Gaussian position estimation network $\mathcal{D}^P_{\boldsymbol{\theta_1}}$ utilizes a U-Net as the backbone. The architecture incorporates four stages of $2\times$ down-sampling using average pooling to extract essential feature details. Symmetrically, the network features four stages of $2\times$ up-sampling, achieved through transpose convolutional neural networks. The EVA module is incorporated before the $4\times$, $8\times$, $16\times$ down-sampling and up-sampling blocks. The channel dimension starts at 64 prior to the first down-sampling block, doubling after each down-sampling block and halving after each up-sampling block, which is facilitated by two residual blocks [2]. The Gaussian attribute estimation network $\mathcal{D}^A_{\boldsymbol{\theta_2}}$ also employs a U-Net backbone, but it does not include the EVA modules and performs only two stages of $2\times$ down-sampling. The architecture of the feature refiner, $\mathcal{D}^R_{\boldsymbol{\theta_3}}$, mirrors that of $\mathcal{D}^A_{\boldsymbol{\theta_2}}$, but operates in a recurrent manner.

More training details. The training hyper-parameters are set as follows: $\lambda_1 = 1, \lambda_2 = 1, \lambda_3 = 10^3, \lambda_{\text{opacity}} = 1, \lambda_{\text{opacity}} = 1, \lambda_{\text{render}} = 0.25$, and $t = 0.05$. The number of recurrent loops L for the feature refinement module is empirically set to $L = 1$ to enhance temporal efficiency. Each training batch contains 2 to 4 source view images, depending on the specific reconstruction task. For instance, the stereo reconstruction task in the experiment part utilizes 2 source view images. For novel view image supervision, 3 randomly selected views are chosen between each adjacent pair of source views to compute $\mathcal{L}_{\text{refine}}$ and $\mathcal{L}_{\text{render}}$. The learning rate for deep supervised pre-training and overall network training is initialized to 0.0002 and decreases linearly with the number of training epochs.

B. Real-world Data Assessment

We evaluate our model on HuMMan [1], a real-world dataset captured with RGB cameras at a resolution of 1024. Using images from the front two cameras (ID: 1 and ID: 9) as inputs, we infer the 3D Gaussians through EVA-Gaussian and render novel views from the viewpoint of ID: 0. The visualization results, as illustrated in Fig. 1, demonstrate that EVA-Gaussian produces high-quality novel view images in real-world settings. Notably, GPS-Gaussian fails to produce reasonable results due to the sparse input view angles of only 90°, which further underscores the robustness of our approach.

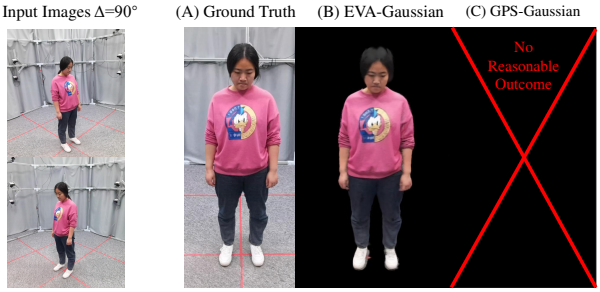

Figure 1. Visualization results on real-world data. Minor artifacts on the human boundary mainly arise from the noisy human mask. Notably, GPS-Gaussian cannot generate reasonable outcome.

C. More Visualization Results

In this section, we present additional visualization results in Fig. 2 to compare our method with SOTA approaches GPS-Gaussian and ENeRF on the Thuman2.0 [5] and Thumansit [6] datasets at a resolution of 1024×1024. The results demonstrate that EVA-Gaussian achieves the highest novel view fidelity across various camera viewpoint settings. In contrast, GPS-Gaussian struggles to handle the artifacts produced by errors in geometric predictions, while ENeRF generates much more blurry and low-fidelity results compared to both GPS-Gaussian and EVA-Gaussian. Notably, under settings of large viewpoint discrepancy, e.g., $\Delta = 90°$, EVA-Gaussian maintains robust performance, while GPS-Gaussian fails to function effectively in these scenarios.

D. Proof of Depth Equality

In this section, we prove that for each pixel on the 3D Gaussian maps $\{\boldsymbol{M}_i\}_{i=1}^n$, the rendered depth equals to the predicted 3D Gaussian depth.

We begin by defining the collection for opacity parameters as $\boldsymbol{o} := [o_1, \cdots, o_i, \cdots, o_N] \in \mathbb{R}^N$ of all considered 3D Gaussians and the collection of all 3D Gaussian scaling

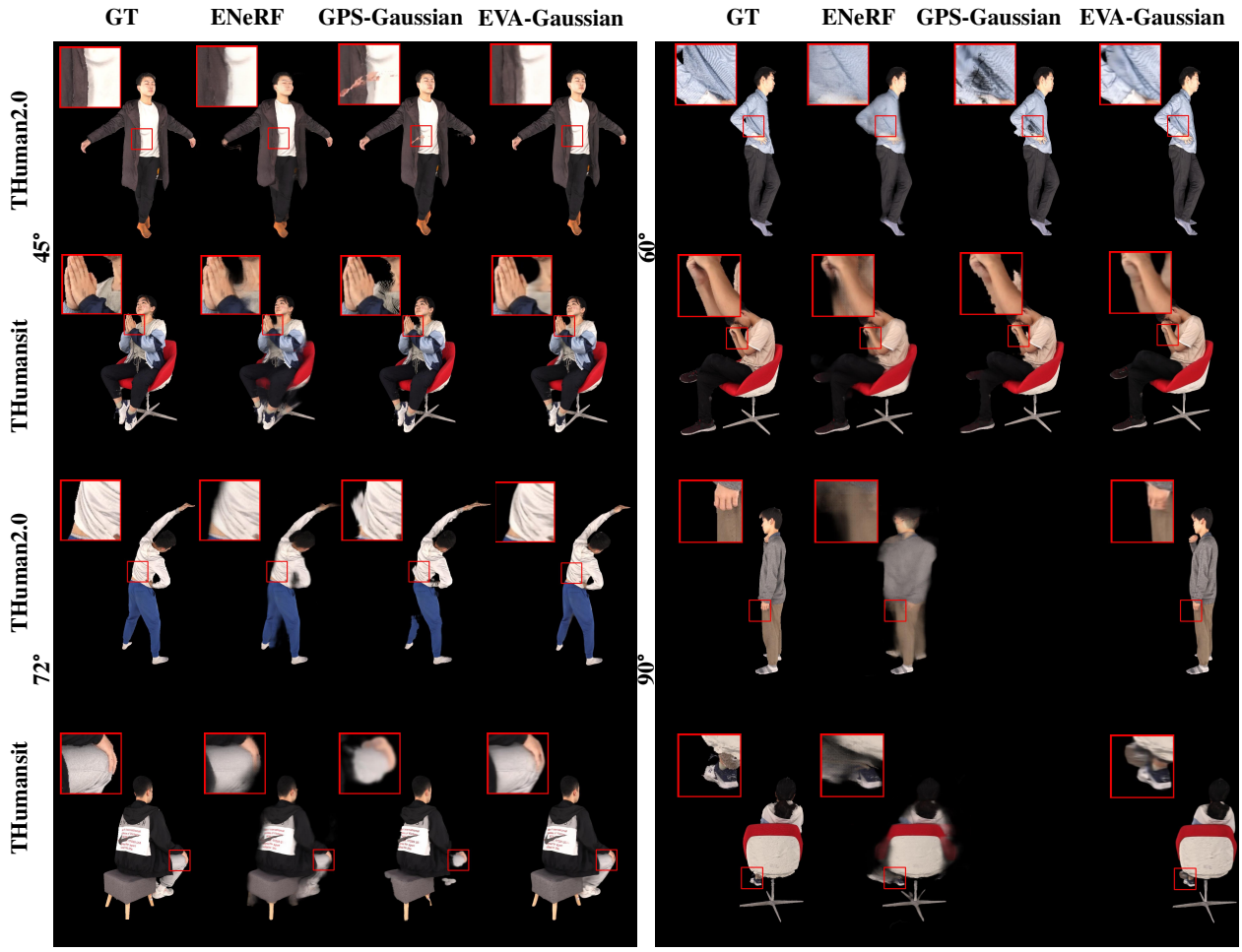

Figure 2. Qualitative comparison on THuman2.0 and THumansit. EVA-Gaussian demonstrates superior novel view rendering quality under diverse camera settings.

factors as:

$$\tilde{\boldsymbol{S}} = \left[\boldsymbol{s}^1, \boldsymbol{s}^2, \boldsymbol{s}^3\right]^T = \begin{pmatrix} s_1^1 & s_2^1 & \cdots & s_N^1 \\ s_1^2 & s_2^2 & \cdots & s_N^2 \\ s_1^3 & s_2^3 & \cdots & s_N^3 \end{pmatrix}, \quad (1)$$

where $\boldsymbol{s}^1 := [s_1^1, s_2^1, \cdots, s_N^1] \in \mathbb{R}^N$, $\boldsymbol{s}^2 := [s_1^2, s_2^2, \cdots, s_N^2] \in \mathbb{R}^N$ and $\boldsymbol{s}^3 := [s_1^3, s_2^3, \cdots, s_N^3] \in \mathbb{R}^N$. For the 3D Gaussian with the i-th greatest depth, the associated scaling matrix is constructed from the corresponding scaling factors as:

$$\boldsymbol{S}_i = \begin{pmatrix} s_i^1 & 0 & 0 \\ 0 & s_i^2 & 0 \\ 0 & 0 & s_i^3 \end{pmatrix}, \quad (2)$$

and its z-value is denoted by z_i. The rendered depth map is expressed as:

$$D(\boldsymbol{x}) = \sum_{i=1}^{N} z_i o_i G_i^{'}(\boldsymbol{x}) \prod_{j=1}^{i-1} (1 - o_j G_j^{'}(\boldsymbol{x}))), \quad (3)$$

where $\boldsymbol{x} \in \mathbb{R}^2$ is a variable on the coordinate system of the image plane and $G_i^{'}(\boldsymbol{x})$ is the 2D Gaussian that corresponds to the 3D Gaussian with the i-th greatest depth after splatting.

In the camera's coordinate system, we define a 3D Gaussian as on the reprojected ray of a pixel $\boldsymbol{x}'$, in condition that the center of this 3D Gaussian lies along the ray originating from the camera center and pointing toward the point $[\boldsymbol{x}', 1]$. We use $Z(\boldsymbol{x}')$ to denote the z-value of the first 3D Gaussian that appears on this reprojected ray.

Based on the above definitions, we have the following theorem:

Theorem D.1. *When the opacity $\boldsymbol{o}$ approaches $\mathbf{1}$ and each value in $\tilde{\boldsymbol{S}}$ is sufficiently small, it holds for each pixel $\boldsymbol{x}'$ on the image plane that:*

$$\lim_{\substack{\boldsymbol{o}\to\mathbf{1} \\ \tilde{\boldsymbol{S}}\to\mathbf{0}^+}} D(\boldsymbol{x}') = Z(\boldsymbol{x}'). \quad (4)$$

Theorem D.1 implies that the z-value of the 3D Gaussian

at pixel $\boldsymbol{x}$ is equal to the corresponding value on the depth map when the scale of Gaussian is sufficiently small and the opacity approaches 1. To prove Theorem D.1, we introduce the following lemma from the well-known Moore-Osgood Theorem in [3]:

Lemma D.1. ***(Moore-Osgood Theorem)*** *Let (Γ, d_Γ) be a metric space and $(\gamma_{k,p})_{k,p\in\mathbb{N}}$ be a sequence such that $\gamma_{\infty,p} := \lim_{k\to\infty}\gamma_{k,p}$ exists for every $p \in \mathbb{N}$ and $\gamma_{k,\infty} := \lim_{p\to\infty}\gamma_{k,p}$ exists for every $k \in \mathbb{N}$. If (i) $\lim_{p\to\infty}\sup_{k\in\mathbb{N}} d_\Gamma(\gamma_{k,p},\gamma_{k,\infty}) = 0$ and (ii) $\lim_{k\to\infty} d_\Gamma(\gamma_{k,p},\gamma_{\infty,p}) = 0, \forall p \in \mathbb{N}$, then the joint limit $\lim_{k,p\to\infty}\gamma_{n,k}$ exists. In particular, it holds that $\lim_{k,p\to\infty}\gamma_{k,p} = \lim_{p\to\infty}\gamma_{\infty,p} = \lim_{k\to\infty}\gamma_{k,\infty}$.*

Lemma D.1 can be regarded as a special case of Theorem 7.11 from [4]. This lemma states that for a doubly-indexed sequence, if the sequence converges uniformly with respect to one index while converging pointwise with respect to the other index, then the limit of the sequence exists. Moreover, this limit is equivalent to the individual limits obtained by separately considering each index, regardless of the order in which the limiting processes are performed. This result can be extended to continuous multi-variable functions.

Specifically, if a continuous function demonstrates uniform convergence with respect to one variable and pointwise convergence with respect to another variable, then the joint limit of the function with respect to both variables can be decomposed into the separate limits with respect to each variable considered independently.

Based on this theoretical foundation, we are now ready to proceed with the proof of Theorem D.1.

Proof. When the opacity value $o_i \in \mathbb{R}$ approaches 1 and the scale factor $\boldsymbol{s}_i = [s_i^1, s_i^2, s_i^3] \in \mathbb{R}^3$ is sufficiently small for each Gaussian, the depth value is given by:

$$\lim_{\substack{\boldsymbol{o}\to\mathbf{1}\\ \hat{\boldsymbol{S}}\to\mathbf{0}^+}} D(\boldsymbol{x}') = \lim_{\substack{\boldsymbol{o}\to\mathbf{1}\\ \hat{\boldsymbol{S}}\to\mathbf{0}^+}} \sum_{i=1}^{N}(z_i o_i G_i'(\boldsymbol{x}')\prod_{j=1}^{i-1}(1-o_j G_j'(\boldsymbol{x}'))) \tag{5}$$

$$= \lim_{\substack{\boldsymbol{o}\to\mathbf{1}\\ \hat{\boldsymbol{S}}\to\mathbf{0}^+}} \sum_{i=1}^{N}(z_i o_i e^{-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_i)^T(\boldsymbol{J}\boldsymbol{W}\boldsymbol{R}_i\boldsymbol{S}_i\boldsymbol{S}_i^T\boldsymbol{R}_i^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_i)}$$
$$\prod_{j=1}^{i-1}(1-o_j e^{-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_j)^T(\boldsymbol{J}\boldsymbol{W}\boldsymbol{R}_j\boldsymbol{S}_j\boldsymbol{S}_j^T\boldsymbol{R}_j^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_j)})) \tag{6}$$

$$\overset{(a)}{=} \lim_{\hat{\boldsymbol{S}}\to\mathbf{0}^+}\lim_{\boldsymbol{o}\to\mathbf{1}} \sum_{i=1}^{N}(z_i o_i e^{-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_i)^T(\boldsymbol{J}\boldsymbol{W}\boldsymbol{R}_i\boldsymbol{S}_i\boldsymbol{S}_i^T\boldsymbol{R}_i^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_i)}$$
$$\prod_{j=1}^{i-1}(1-o_j e^{-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_j)^T(\boldsymbol{J}\boldsymbol{W}\boldsymbol{R}_j\boldsymbol{S}_j\boldsymbol{S}_j^T\boldsymbol{R}_j^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_j)})), \tag{7}$$

where (a) is from Lemma D.1. Specifically, the function $\sum_{i=1}^{N}(z_i o_i G_i'(\boldsymbol{x})\prod_{j=1}^{i-1}(1-o_j G_j'(\boldsymbol{x})))$ is continuous with respect to the two variables $\boldsymbol{o}$ and $\boldsymbol{s}$. Besides, it converges uniformly as $\hat{\boldsymbol{S}} \to \mathbf{0}^+$ and as $\boldsymbol{o} \to \mathbf{1}$. This implies that the joint limit of $\boldsymbol{o}$

and $\boldsymbol{s}$ can be decomposed into the separate limits of $\boldsymbol{o}$ and $\boldsymbol{s}$. Thus, we have:

$$\begin{aligned}
&\lim_{\hat{\boldsymbol{S}}\to\mathbf{0}^+}\lim_{\boldsymbol{o}\to\mathbf{1}}\sum_{i=1}^{N}(z_i o_i e^{-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_i)^T(\boldsymbol{JWR}_i\boldsymbol{S}_i\boldsymbol{S}_i^T\boldsymbol{R}_i^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_i)}\\
&\prod_{j=1}^{i-1}(1-o_j e^{-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_j)^T(\boldsymbol{JWR}_j\boldsymbol{S}_j\boldsymbol{S}_j^T\boldsymbol{R}_j^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_j)}))\\
=&\lim_{\hat{\boldsymbol{S}}\to\mathbf{0}^+}\sum_{i=1}^{N}(\lim_{o_i\to 1} z_i o_i e^{-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_i)^T(\boldsymbol{JWR}_i\boldsymbol{S}_i\boldsymbol{S}_i^T\boldsymbol{R}_i^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_i)}\\
&\lim_{(o_j,\cdots,o_{i-1})\to\mathbf{1}}\prod_{j=1}^{i-1}(1-o_j e^{-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_j)^T(\boldsymbol{JWR}_j\boldsymbol{S}_j\boldsymbol{S}_j^T\boldsymbol{R}_j^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_j)}))\qquad(8)\\
=&\lim_{\hat{\boldsymbol{S}}\to\mathbf{0}^+}\sum_{i=1}^{N}(z_i e^{-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_i)^T(\boldsymbol{JWR}_i\boldsymbol{S}_i\boldsymbol{S}_i^T\boldsymbol{R}_i^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_i)}\\
&\prod_{j=1}^{i-1}(1-e^{-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_j)^T(\boldsymbol{JWR}_j\boldsymbol{S}_j\boldsymbol{S}_j^T\boldsymbol{R}_j^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_j)})).\qquad(9)
\end{aligned}$$

The 3D Gaussians typically assume an ellipsoidal geometric shape. However, when the scaling factors are sufficiently small, the ellipsoid can be approximated as a sphere, such that $\boldsymbol{s}^1=\boldsymbol{s}^2=\boldsymbol{s}^3$. As a result, the scaling matrix for the 3D Gaussian with the i-th greatest depth becomes:

$$\boldsymbol{S}_i^{'}:=\begin{pmatrix}s_i^1 & 0 & 0\\ 0 & s_i^1 & 0\\ 0 & 0 & s_i^1\end{pmatrix}.\qquad(10)$$

Consequently, we have:

$$\begin{aligned}
&\lim_{\hat{\boldsymbol{S}}\to\mathbf{0}^+}\sum_{i=1}^{N}(z_i e^{-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_i)^T(\boldsymbol{JWR}_i\boldsymbol{S}_i\boldsymbol{S}_i^T\boldsymbol{R}_i^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_i)}\\
&\prod_{j=1}^{i-1}(1-e^{-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_j)^T(\boldsymbol{JWR}_j\boldsymbol{S}_j\boldsymbol{S}_j^T\boldsymbol{R}_j^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_j)}))\\
=&\lim_{\substack{\hat{\boldsymbol{S}}\to\mathbf{0}^+\\ \boldsymbol{s}^1=\boldsymbol{s}^2=\boldsymbol{s}^3}}\sum_{i=1}^{N}(z_i e^{-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_i)^T(\boldsymbol{JWR}_i\boldsymbol{S}_i\boldsymbol{S}_i^T\boldsymbol{R}_i^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_i)}\\
&\prod_{j=1}^{i-1}(1-e^{-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_j)^T(\boldsymbol{JWR}_j\boldsymbol{S}_j\boldsymbol{S}_j^T\boldsymbol{R}_j^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_j)}))\qquad(11)\\
=&\lim_{\boldsymbol{s}^1\to\mathbf{0}^+}\sum_{i=1}^{N}(z_i e^{-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_i)^T(\boldsymbol{JWR}_i\boldsymbol{S}_i^{'}\boldsymbol{S}_i^{'T}\boldsymbol{R}_i^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_i)}\\
&\prod_{j=1}^{i-1}(1-e^{-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_j)^T(\boldsymbol{JWR}_j\boldsymbol{S}_j^{'}\boldsymbol{S}_j^{'T}\boldsymbol{R}_j^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_j)})).\qquad(12)
\end{aligned}$$

From (12), we see that when $\boldsymbol{x}'=\boldsymbol{\mu}_i$, it gives

$$e^{-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_i)^T(\boldsymbol{JWR}_i\boldsymbol{S}_i^{'}\boldsymbol{S}_i^{'T}\boldsymbol{R}_i^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_i)}=1.\qquad(13)$$

Otherwise, if $\boldsymbol{x}' \neq \boldsymbol{\mu}_i$, we have

$$\begin{aligned}
&\lim_{s_i^1 \to 0^+} e^{-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_i)^T(\boldsymbol{J}\boldsymbol{W}\boldsymbol{R}_i\boldsymbol{S}_i'\boldsymbol{S}_i'^T\boldsymbol{R}_i^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_i)} \\
=&\lim_{s_i^1 \to 0^+} e^{-\frac{1}{2(s_i^1)^2}(\boldsymbol{x}'-\boldsymbol{\mu}_i)^T(\boldsymbol{J}\boldsymbol{W}\boldsymbol{R}_i\boldsymbol{R}_i^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_i)} \\
=&\, 0. \qquad (14)
\end{aligned}$$

By combining Eq. 5 – Eq. 14, we have

$$\begin{aligned}
\lim_{\substack{\boldsymbol{o}\to\boldsymbol{1} \\ \hat{\boldsymbol{S}}\to\boldsymbol{0}^+}} D(\boldsymbol{x}') =& \sum_{i=1}^{N}\Big(\lim_{s_i^1\to 0^+} z_i e^{-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_i)^T(\boldsymbol{J}\boldsymbol{W}\boldsymbol{R}_i\boldsymbol{S}_i'\boldsymbol{S}_i'^T\boldsymbol{R}_i^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_i)} \\
&\prod_{j=1}^{i-1}\lim_{s_j^1\to 0^+}\big(1-e^{-\frac{1}{2}(\boldsymbol{x}'-\boldsymbol{\mu}_j)^T(\boldsymbol{J}\boldsymbol{W}\boldsymbol{R}_j\boldsymbol{S}_j'\boldsymbol{S}_j'^T\boldsymbol{R}_j^T\boldsymbol{W}^T\boldsymbol{J}^T)^{-1}(\boldsymbol{x}'-\boldsymbol{\mu}_j)}\big)\Big) \qquad (15) \\
=& Z(\boldsymbol{x}'), \qquad (16)
\end{aligned}$$

which completes the proof. □

References

[1] Zhongang Cai, Daxuan Ren, Ailing Zeng, Zhengyu Lin, Tao Yu, Wenjia Wang, Xiangyu Fan, Yang Gao, Yifan Yu, Liang Pan, Fangzhou Hong, Mingyuan Zhang, Chen Change Loy, Lei Yang, and Ziwei Liu. HuMMan: Multi-modal 4d human dataset for versatile sensing and modeling. In *17th European Conference on Computer Vision, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part VII*, pages 557–577. Springer, 2022. 1

[2] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016. 1

[3] Antonis Papapantoleon, Dylan Possamaï, and Alexandros Saplaouras. Stability of backward stochastic differential equations: the general lipschitz case. *Electronic Journal of Probability*, 28:1–56, 2023. 3

[4] Walter Rudin et al. *Principles of mathematical analysis*. McGraw-hill New York, 1964. 3

[5] Tao Yu, Zerong Zheng, Kaiwen Guo, Pengpeng Liu, Qionghai Dai, and Yebin Liu. Function4d: Real-time human volumetric capture from very sparse consumer rgbd sensors. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR2021)*, 2021. 1

[6] Jiajun Zhang, Yuxiang Zhang, Hongwen Zhang, Boyao Zhou, Ruizhi Shao, Zonghai Hu, and Yebin Liu. Ins-hoi: Instance aware human-object interactions recovery. *arXiv preprint arXiv:2312.09641*, 2023. 1