Normalizing Flows on the Product Space of SO(3) Manifolds for Probabilistic Human Pose Modeling

Supplementary Material

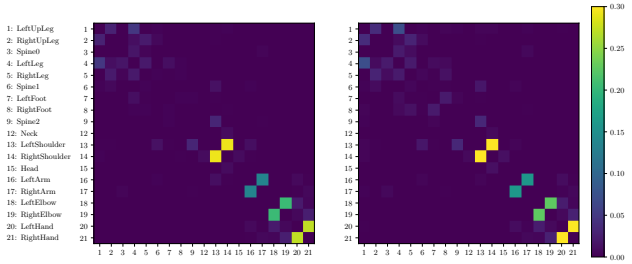
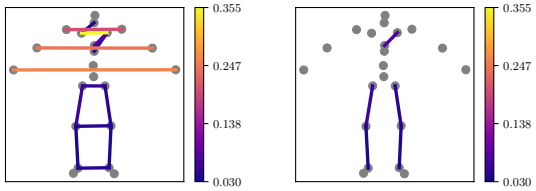


Figure 6. Spherical correlation coefficient of samples from HuProSO3 (left) and from AMASS training dataset (right), excluding toe bases and root joints. Entries on the diagonal are set to zero because their correlation coefficient $\hat{\rho}_{XY} = 1$ is always equal to one. The coefficients are computed using 100k samples.



(a) All coefficients. (b) Along kinematic tree.

Figure 7. Color-coded spherical correlation coefficients illustrated along a skeleton model of the human body. For clarity, only the coefficients with a value $\hat{\rho}_{XY} > 0.03$ are depicted.

To complement the main paper, this supplementary material delves into additional aspects not covered in detail previously. We include an analysis of the AMASS database, provide extended qualitative and quantitative evaluations, and share details on the implementation and training.

A. Dataset Analysis

Given that both the development and assessment of our method are grounded in a statistical analysis of the AMASS database, we present the key findings and insights from this analysis in this section.

A.1. Correlation Between Different Joints

Computing a correlation or dependence coefficient on SO(3) is not straightforward. Hence, we use a unit quaternion representation of orientations and follow [9] to compute a spherical correlation coefficient

$$\hat{\rho}_{XY} = \frac{\det\left(\frac{\sum_i X_i Y_i^T}{n}\right)}{\sqrt{\det\left(\frac{\sum_i X_i X_i^T}{n}\right)\det\left(\frac{\sum_i Y_i Y_i^T}{n}\right)}} \tag{12}$$

for the n samples X_i, Y_i on the 3-sphere $\mathcal{S}^3$, on which all quaternions reside. We visualize the spherical correlation coefficients for all dynamic joints of the AMASS database (excluding toe bases and the root orientation) in Fig. 6. The illustration depicted in Appendix A.1 provides a better intuition of these correlation coefficients, comparing the correlation coefficients along the kinematic tree and for all joints. Notably, high correlations are observed particularly for joints at different leaves of the kinematic tree, such as for the left and right arm joints. We note that the used correlation coefficient only captures certain dependencies on the 3-sphere.

A.2. Distribution Gaps for the AMASS Datasets

Evaluating an unconditional prior typically assumes *iid* samples in training and test datasets, which is not the case for the AMASS [19] database. To compare the distributions of two datasets based on a sample-based similarity metric of the rotations, we compute the earth mover's distance (EMD) [28] using the geodesic distance as the distance measure. We compare the EMDs for individual joints and for all joints for the common AMASS datasets split [19] in Fig. 8 and the EMD between various datasets of the AMASS database in Fig. 9. For computational reasons, we use only 2k samples for the transport problem, acknowledging potential inaccuracies in higher dimensions. Nevertheless, the distribution gap is highlighted because the entries of the correlation matrix that correspond to the auto-correlation are markedly lower than the cross-correlation. This aligns with the notable performance disparity between training and test datasets shown in the main paper (Tab. 1).

B. Implementation and Training Details

In the following, we elucidate the implementation and training details for HuProSO3 and the integration of the other evaluated methods.

B.1. HuProSO3

We trained the following HuProSO3 models: one as an unconditional prior, one for inverse kinematics, one for inverse kinematics with randomly occluded 3D key points, and one for 2D to 3D uplifting.

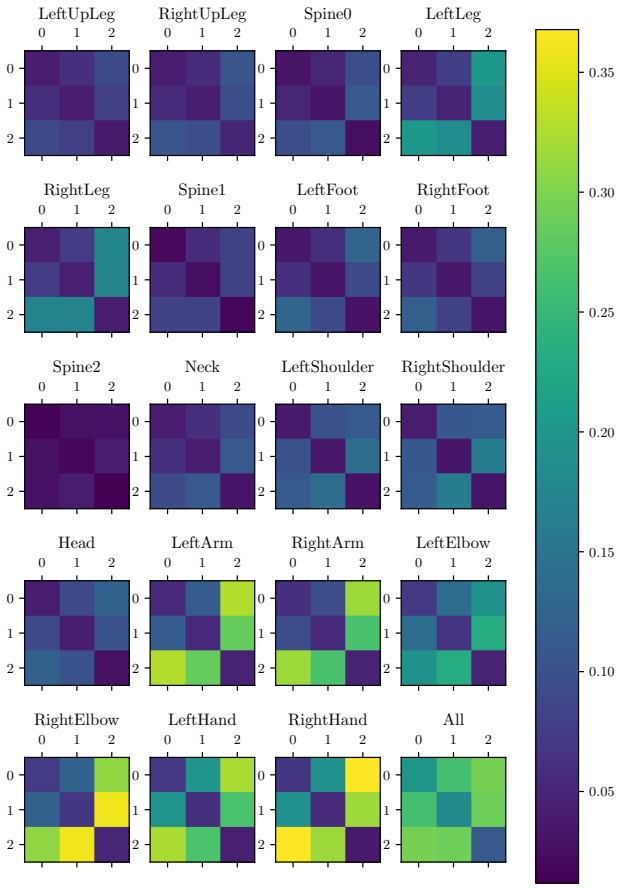


Figure 8. Earth mover's distance between AMASS training (0), validation (1), and test (2) datasets based on the geodesic distance for all dynamic SMPL joints and when comparing *All* joints. For the *All* category, the EMD is computed using the average geodesic distance across multiple joint rotations.

Architecture. The normalizing flow architecture for learning the density $p(\mathbf{R})$ is consistent across all experiments. We use 12 Möbius coupling layers, with each layer (except the final one) followed by a quaternion affine transformation, totaling 11 quaternion affine layers. The parameters of the Möbius transformation are computed by an MLP $g_{\text{c-M}}(\cdot)$ with three hidden layers and ReLU activations and a hidden dimension of 16. For the presented applications that require conditioning (inverse kinematics and 2D to 3D uplifting), an MLP $\mathbf{c} = g(\mathbf{c}_{\text{feat}})$ computes the relevant features from the input context vector $\mathbf{c}_{\text{feat}}$. We use an MLP with one hidden layer and a hidden dimension of 64. The output dimension of the feature is $\mathbf{c} \in \mathbb{R}^{64}$. The complete model that is conditioned on the 3D pose has around 1.5 million parameters.

Training. Our models are trained with a batch size of 1k, utilizing the Adam optimizer. We set the initial learning rate to 5e-3 and employ a step learning rate scheduler with a multiplicative factor of 0.5 for learning rate decay.

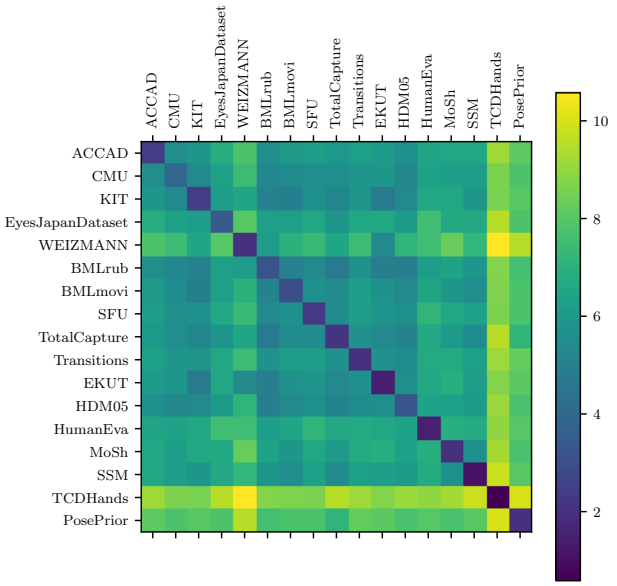


Figure 9. Earth mover distances between various datasets in the AMASS database, computed based on the sum of the geodesic distances of all dynamic joints of the AMASS database.

Run time. Sampling from an autoregressive model is slow and scales with the number of dimensions. Parallelizing the evaluation of the model is in general possible. However, our current implementation does not support this. Therefore, evaluating one batch requires around 1.1s on a NVIDIA A40 GPU, while sampling one batch takes around 2.1s.

B.2. Learning and Optimizing Baseline Methods

We compare our method with implementations of VPoser [25], GAN-S [4], Pose-NDF [34], and HuManiFlow [32]. We use the pre-trained models for the priors VPoser, GAN-S, and Pose-NDF. The normalizing flow based on the 6D representation is implemented using the *CircularAutoregressiveRationalQuadraticSpline* module of the normflows libary [33], with the PDF defined on $\mathbf{x} \in \mathbb{R}^{6\cdot 19}$.

For conditional tasks, we optimize GAN-S and Pose-NDF for inverse kinematics and 2D to 3D uplifting as outlined in the main paper in Sec. 4.2 based on 21 SMPL joints. For GAN-S, we utilize the L-BFGS optimizer, set the learning rate to 1 and perform 500 iterations. We use the existing Pose-NDF repository to optimize for occluded joints, and we mask the joint positions in case of occlusions as outlined in Eq. (11) in the main paper.

We train the extracted model of HuManiFlow [32] using the Adam optimizer with a learning rate of 5e-5 with a step learning rate scheduler and step factor of 0.5, and a batch

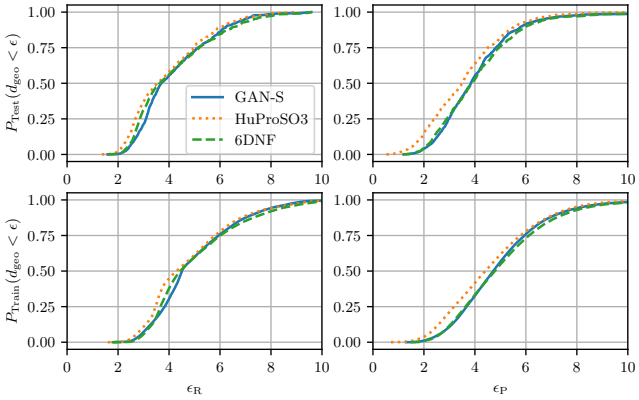


Figure 10. Precision (P) and recall (R) curves for AMASS training and test dataset based on the summed geodesic distances as presented in the main paper, computed for GAN-S, HuProSO3, and 6D normalizing flow. Higher values indicate better quality in all charts.

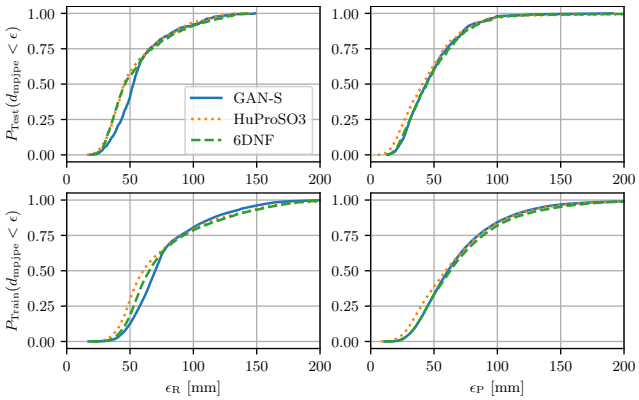


Figure 11. Precision (P) and recall (R) curves for the AMASS training and test datasets based on the MPJPE metric, computed for GAN-S, HuProSO3, and 6D normalizing flow. Higher values indicate better quality in all charts.

size of 500 until convergence. We apply the same masking strategy as for HuProSO3.

B.3. Visualization Techniques

In Fig. 5, we adopt the visualization technique introduced by [20] to display samples on SO(3). A sample on SO(3) is visualized by projecting it onto a 2-sphere and visualizing the third rotation angle through color coding.

C. Additional Qualitative Results

C.1. Correlation Coefficients for Learned Prior

To demonstrate that our prior has effectively learned correlations between different joint rotations, we plot the spherical correlation coefficients computed on sampled poses from the prior in figure Fig. 6 next to spherical correlation coefficients computed from samples of the datasets.

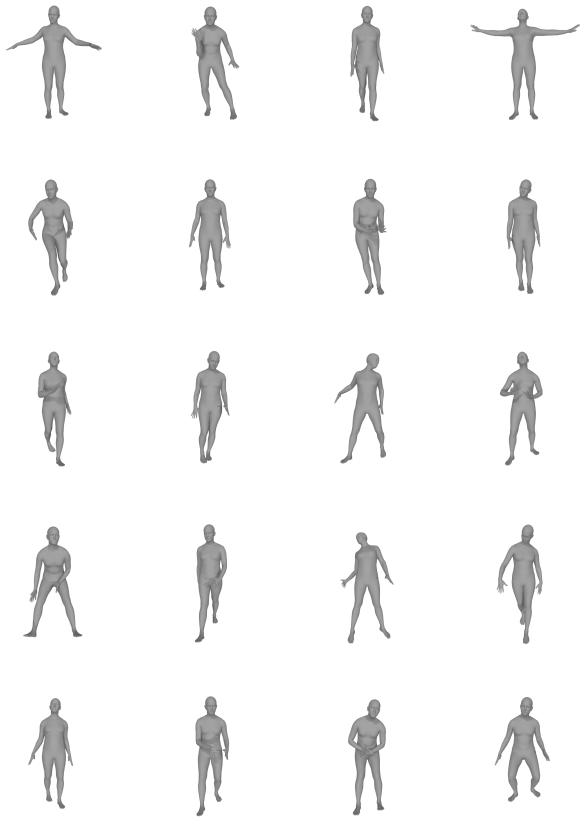


Figure 12. Renderings of randomly sampled human poses from the unconditional HuProSO3 prior.

Table 6. Summary of precision and recall statistics for the AMASS database [19], both on test and training datasets. The values indicate the MPJPEs [mm] for all SMPL joints between samples from the dataset and the evaluated pose prior after applying forward kinematics.

	Test (mean [median])		Train (mean [median])	
	Recall	Precision	Recall	Precision
GAN-S [4]	50.6 [48.4]	68.7 [64.2]	39.7 [35.0]	56.2 [49.7]
6D NF	46.3 [**39.9**]	65.8 [57.8]	39.4 [34.2]	57.2 [49.4]
Ours	**45.4** [40.2]	**61.2 [52.1]**	**34.3 [28.8]**	**51.3 [43.2]**

C.2. Precision Recall Curves

To evaluate the priors, we compute the precision and recall curves. We plot the curves in Fig. 10. To assess the priors, we compute and plot the precision and recall curves, as shown in Fig. 10 and Fig. 11. This comparison, similar to the one presented in Tab. 1 of the main paper, is based on the MPJPE metric. Additionally, the mean and median values of these metrics are detailed in Tab. 6.

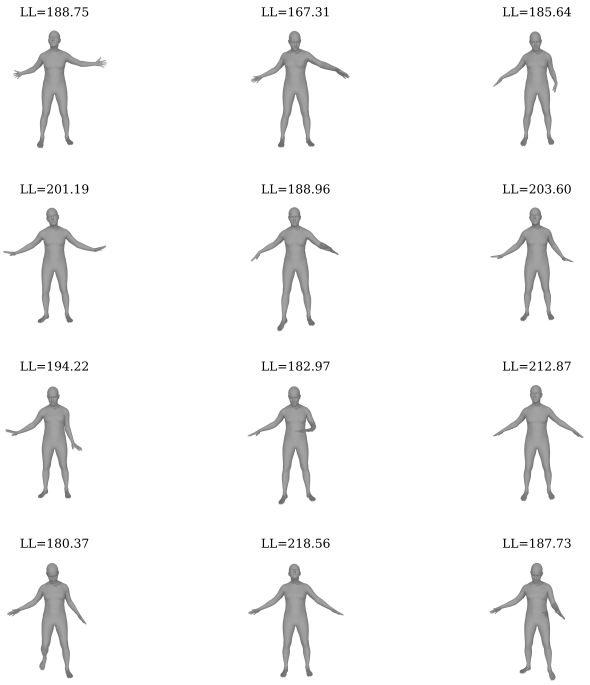


Figure 13. Renderings of human poses sampled from the predicted distribution, conditioned on the 2D key point positions of all joints, excluding the left arm and the right leg. The log likelihood corresponding to a normalized density is displayed above the considered sample.

C.3. Rendered Samples from Unconditional Prior

To demonstrate HuProSO3's capability as a generative model for sampling realistic and diverse human poses, we present renderings of randomly selected samples in Fig. 12.

C.4. Rendered Samples from 2D to 3D Uplifting

Based on the setting and conditioning in Fig. 5, we render 12 poses that are sampled from the learned distribution $p(\mathbf{R}|\mathbf{c})$ in Fig. 13, where $\mathbf{c}$ is derived from 2D key points with the left arm and the right leg occluded. These samples, displayed in Fig. 13, reveal that while the model often predicts straight right legs, the left arm's pose varies significantly, which follows the training dataset's distribution. For joints where the given 2D key point positions allow inferring their rotations, the estimates show less variability, as we also visualize in Fig. 5 of the main paper for the standard deviation of the joint's positions and rotations.

Optimization-based methods are limited to inferring a single pose from occluded key points. While this approach might yield accurate results on average, as reflected in mean metrics, it fails to capture the inherent ambiguity in these scenarios.

Table 7. Log likelihood evaluation for inverse kinematics (IK), rotation distribution estimation given 2D key points (2D to SO(3)), and an unconditional prior. Unless specified otherwise, results pertain to the AMASS test dataset.

Method	IK	2D to SO(3)	Prior	Prior (Train)
HuManiFlow [32]	100.8	83.6	-	-
Ours	217.5	202.2	137.6	184.7

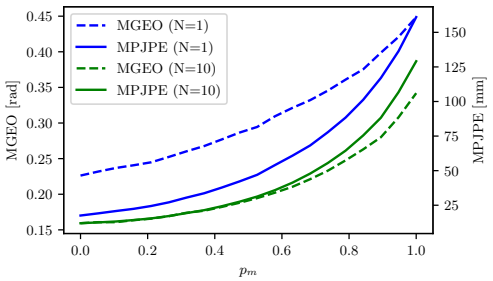
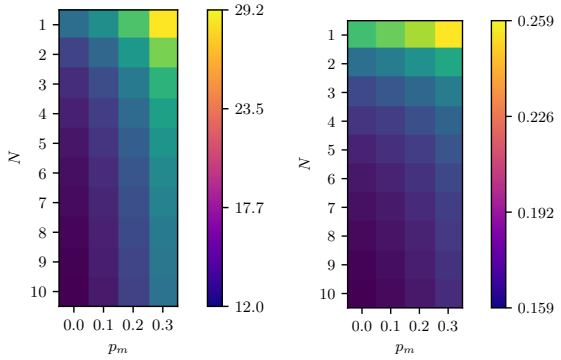


Figure 14. MGEO and MPJPE for a variation of the mask probability p_m based on one sample and the mean over 10 samples drawn from HuProSO3 conditioned on partially given 3D key point information.



(a) Positional error MPJPE [mm]. (b) Rotational error MGEO [rad].

Figure 15. MPJPEs and MGEOs for different masking probabilities p_m and numbers of samples from the learned distribution N. The metrics are calculated based on the mean of the sampled poses by computing the average rotation for each joint.

D. Additional Quantitative Results

D.1. Evaluation of Likelihood

We evaluate the likelihoods for the unconditional prior with HuProSO3 and for conditional distributions for HuProSO3 and the HuManiFlow approach in Tab. 7. For the unconditional prior, the likelihood evaluations also reveal a significant gap between the training and test distributions.

D.2. Comparison to Pose-NDF as Pose Prior

We evaluate Pose-NDF as a pose prior by computing the precision and recall statistics. In a first experiment, we evaluate when initializing with noise. However, this does not

Table 8. Summary of precision and recall statistics for the AMASS database [19], both on test and training datasets. The reported values represent the cumulative geodesic distances for all joint rotations between samples from the dataset and the evaluated pose prior. Pose-NDF 1 was optimized using random intialization, Pose-NDF 2 using slightly noised poses from the AMASS test dataset.

	Test (mean [median])		Train (mean [median])	
	Recall	Precision	Recall	Precision
Pose-NDF 1 [34]	17.7 [17.7]	14.5 [14.4]	17.5 [17.2]	14.7 [14.4]
Pose-NDF 2 [34]	4.83 [4.77]	5.63 [5.42]	6.11 [5.92]	6.88 [6.65]
Ours	3.44 [2.95]	4.24 [3.71]	2.93 [2.64]	3.90 [3.59]

Table 9. Summary of precision and recall statistics for the AMASS database [19], both on test and training datasets. The reported values represent the cumulative geodesic distances for all joint rotations between samples from the dataset and the evaluated pose prior.

	Test (mean [median])		Train (mean [median])	
	Recall	Precision	Recall	Precision
GAN-S [4]	3.76 [3.34]	4.51 [4.23]	3.57 [3.34]	4.38 [4.13]
6D NF	3.66 [3.16]	4.50 [4.00]	3.55 [3.32]	4.42 [4.10]
Ours	3.44 [2.95]	4.24 [3.71]	2.93 [2.64]	3.90 [3.59]

Table 10. Comparison to GFPose-rot: Minimum MGEO and minimum MPJPE are computed based on 20 generated samples for the occlusion of leg (L), arm and hand (A), and upper arm (S). The results are presented for 10k random samples from the AMASS test datasets. The GEO metrics are averaged over 21 joints.

Method	minMGEO			minMPJPE		
Occlusion	L	A	S	L	A	S
GFPose-rot (N=1)	0.104	0.108	0.089	7.9	14.3	4.9
GFPose-rot (N=20)	0.103	0.107	0.088	7.8	14.1	4.8
Ours (N=1)	0.217	0.254	0.208	20.8	39.0	18.7
Ours (N=20)	0.070	0.081	0.067	5.7	11.2	5.2

result in realistic poses since Pose-NDF generates realistic poses when the initialized poses are close to the training distribution. In a second experiment, we add a small amount of noise to poses of the test distribution ($\sigma_{\text{noise}} = 0.1$), which provides realistic poses. However, such an initialization inherently biases the optimization towards in-distribution samples. Therefore, it is highly depending on the similarity between training and test distribution.

D.3. Comparison to GFPose-rot

Directly comparing to GFPose [3] is not possible since it was not trained on the AMASS database and it is parameterized with joint positions. Therefore, we adapt the implementation of GFPose and we train it on the AMASS database using the axis-angle representation using the same hyperparameters as in the original repository. For the occlusions, we apply the same masking strategy as in our implementations. We follow [3] and evaluate using the minimum error sample. We use 10 times fewer samples than GFPose (N=20) and we report the results for minimum geodesic distance and joint position error in Tab. 10. In our experiments, GFPose-rot collapses to the mean pose. While our results are worse for single sample evaluations, our model provides more diverse samples than GFPose-rot.

Here, a disadvantage of our model becomes apparent: Since our base distribution is uniform on SO(3), computing the mode as when considering a Gaussian distribution is not possible. This might be a reason, why generating an individual sample does not achieve competing results.

While GFPose-rot achieves partially better results, it does not provide a normalized density.

Table 11. Comparison of per-vertex errors [mm] across various occluded joints in the AMASS test dataset: left leg (L), left arm and hand (A+H), and right shoulder and upper arm (S+UA). The results are based on 60 frames as reported in [34].

Method	L	A+H	S + UA
VPoser [25]	25.3	85.1	99.8
HuMoR [26]	56.0	78.3	47.5
Pose-NDF [34]	**24.9**	78.1	76.3
Ours (N=10)	34.0	**57.5**	**34.5**

D.4. Per-Vertex Errors for Inverse Kinematics and Occluded Joints

We compare HuProSO3's per-vertex error performance with HuMoR [26] using *TestOpt*, VPoser [25], and Pose-NDF [34] for 60 frames, following the protocol in [34]. We present the per-vertex error results for inverse kinematics and occluded joints in Tab. 11. While the optimization-based methods achieve a better performance for the MPJPE metric, the presented results in Tab. 11 also support that the wrong rotation estimates result in worse performance when comparing all mesh vertices on the rendered human.

D.5. Evaluations for Varying Numbers of Samples

We present additional evaluation results for varying numbers of samples that are used for computing positional and rotational errors in Fig. 15 and Fig. 14. Fig. 15 extends the visualizations from Fig. 14 by presenting MPJPE and MGEO metrics for different masking probabilities and sample counts drawn from HuProSO3, with each joint masked with a probability of p_m.

Baselines and Setup. In addition to HuProSO3, as detailed in the main paper, and the model based on the HuManiFlow implementation, we also compare with an implementation that uses normalizing flows defined on SO(3) as for HuProSO3, but with fixed ancestor-conditioning as suggested in [32]. We train this model following the same strategy as for HuProSO3. For evaluation, joints are ran-

domly masked with a probability of $p_m = 0.3$. We present the errors for the average joint rotations (as in the main paper) and, additionally, based on the pose out of the N sampled poses that results in the lowest considered error metric (minimum error sample). We compute all results based on 10k randomly drawn samples from the AMASS test dataset.

Discussion. Our analysis reveals that normalizing flows designed with the SO(3) modeling approach seem to more effectively capture the joint distribution than the approach by [8] applied in [32].While the error for a single sample ($N = 1$) is similar across both ancestor-conditioned models (*SO3 AC* and *HF AC*) for the selected masking strategy, the SO(3)-based model yields lower minimum errors with an increased number of samples. However, ancestor-conditioning along the kinematic tree does not fully capture all statistical dependencies. Consequently, HuProSO3, which is not limited to fixed conditioning sequences and operates on the product space of all SO(3) manifolds, demonstrates notably lower minimum and average pose errors, reflecting its superior capability in learning dependencies of the joint rotations.