MacDiff: Unified Skeleton Modeling with Masked Conditional Diffusion (Supplementary Material)

Contents

1. Theoretical Proofs

1.1. Mutual Information

Proposition 1. *Let X, Y, Z be arbitrary random variables, then the mutual information of X and (Y, Z) can be written as:*

$$I((Y,Z);X) = I(Y;X) + I(Z;X|Y). \tag{1}$$

Proof. We directly prove this proposition from the joint distribution mutual information decomposition formula:

$$\begin{aligned}
I((Y,Z);X) &= \int p(\mathbf{x},\mathbf{y},\mathbf{z}) \log \frac{p(\mathbf{x},\mathbf{y},\mathbf{z})}{p(\mathbf{x})p(\mathbf{y},\mathbf{z})} \mathrm{d}\mathbf{x}\mathrm{d}\mathbf{y}\mathrm{d}\mathbf{z} \\
&= E_{\mathbf{X}}\left[\int p(\mathbf{y},\mathbf{z}|\mathbf{x}) \log \left(\frac{p(\mathbf{y}|\mathbf{x})}{p(\mathbf{y})} \cdot \frac{p(\mathbf{z}|\mathbf{y},\mathbf{x})}{p(\mathbf{z}|\mathbf{y})}\right) \mathrm{d}\mathbf{y}\mathrm{d}\mathbf{z}\right] \\
&= E_{\mathbf{X}}\left[\int p(\mathbf{y}|\mathbf{x}) p(\mathbf{z}|\mathbf{y},\mathbf{x}) \log \frac{p(\mathbf{y}|\mathbf{x})}{p(\mathbf{y})} \mathrm{d}\mathbf{y}\mathrm{d}\mathbf{z}\right] + E_{\mathbf{X}}\left[\int p(\mathbf{y}|\mathbf{x}) p(\mathbf{z}|\mathbf{y},\mathbf{x}) \log \frac{p(\mathbf{z}|\mathbf{y},\mathbf{x})}{p(\mathbf{z}|\mathbf{y})} \mathrm{d}\mathbf{y}\mathrm{d}\mathbf{z}\right] \\
&= E_{\mathbf{X}}\left[\int p(\mathbf{y}|\mathbf{x}) \left(\int p(\mathbf{z}|\mathbf{y},\mathbf{x}) \mathrm{d}\mathbf{z}\right) \log \frac{p(\mathbf{y}|\mathbf{x})}{p(\mathbf{y})} \mathrm{d}\mathbf{y}\right] + E_{\mathbf{Y}}\left[\int p(\mathbf{x},\mathbf{z}|\mathbf{y}) \cdot \log \frac{p(\mathbf{x},\mathbf{z}|\mathbf{y})}{p(\mathbf{x}|\mathbf{y}) \cdot p(\mathbf{z}|\mathbf{y})} \mathrm{d}\mathbf{x}\mathrm{d}\mathbf{z}\right] \\
&= I(Y;X) + I(Z;X|Y).
\end{aligned} \tag{2}$$

□

Further, if X and Z are both independent functions of Y, then Z does not provide more information about X than Y, *i.e.*, $I((Y,Z);X) = I(Y,X)$. In our paper, since the noisy view X_t and the representation of the masked view $Z = \mathcal{E}(X_m)$ are independently obtained from the original data X, we have $I(X;Z) = I((X,X_t);Z) = I(X_t;Z) + I(X;Z|X_t)$.

Proposition 2. *(Data Processing Inequality) Let three random variables form the Markov chain $X \to Y \to Z$, implying Z is conditionally independent of X and only depends on Y. Then no processing of Y can increase the information about X:*

$$I(Y;X) \geq I(Z;X), \tag{3}$$

with the equality $I(Y;X) = I(Z;X)$ if and only if $I(Y;X|Z) = 0$.

Proof. From Proposition 1, we have $I(Y;X) = I((Y,Z);X) - I(Z;X|Y)$. Similarly, $I(Z;X) = I((Y,Z);X) - I(Y;X|Z)$. Therefore, we can obtain that:

$$I(Y;X) - I(Z;X) = I(Y;X|Z) - I(Z;X|Y) \tag{4}$$

$$= I(Y;X|Z) \geq 0, \tag{5}$$

where $I(Z;X|Y) = 0$, if given $X \perp\!\!\!\perp Z|Y$. □

1.2. Bayes Error Rate of Representations

Theorem 3. *(Bayes Error Rate of Representations) Let Y denote the labels of the data and V denote a certain view of the data. For data representation distribution Z, its Bayes error rate can be estimated as:*

$$P_e \leq 1 - e^{-(H(Y) - I(Z;Y))} \tag{6}$$

$$\leq 1 - e^{-(H(Y) - I(Z;Y;V) - I(Z;Y|V))}. \tag{7}$$

Proof. According to [3], the relationship between the Bayes error rate P_e and the conditional entropy $H(Y|Z)$ is:

$$-\ln(1 - P_e) \leq H(Y|Z), \tag{8}$$

which is equivalent to

$$P_e \leq 1 - e^{-H(Y|Z)}. \tag{9}$$

From the definition of mutual information, the relationship between mutual information and entropy is $I(Z;Y) = H(Y) - H(Y|Z)$. Therefore, we have

$$P_e \leq 1 - e^{-(H(Y) - I(Z;Y))}. \tag{10}$$

Further, with the equation $I(Z;Y) = I(Z;Y|V) + I(Z;Y;V)$, we can obtain that

$$P_e \leq 1 - e^{-(H(Y) - I(Z;Y;V) - I(Z;Y|V))}. \tag{11}$$

□

2. Implementation Details

2.1. Generative Models for Comparison

For generative evaluation, we compare our method with the reconstruction-based method SkeletonMAE [12] and diffusion-based methods DDIM [10] and MDM [11] on the NTU 60 xsub dataset. For SkeletonMAE, we follow the re-implementation of [8], which also employs a vanilla Transformer as the backbone and achieves better performance than the original SkeletonMAE.

For DDIM and MDM, the designs of both methods focus on the targets of prediction and training losses and are agnostic to the specific network architecture. Specifically, DDIM follows the training target of DDPM [6] and predicts $\boldsymbol{\epsilon}$. MDM predicts $\boldsymbol{x}_0$ and calculates position loss, velocity loss, and foot contact loss. Therefore, we re-implement DDIM and MDM with the same Transformer architecture as ours (except that MacDiff replaces LN with AdaLN). The model depth is 5 and the embedding dimension is 256. In MDM, we set $\lambda_{pos} = 1, \lambda_{vel} = 1$ and $\lambda_{foot} = 0$ following the default setting. In addition, the original MDM embeds all joints in a single frame into a token, different from our embedding strategy. We implement the original MDM with temporal-only embedding, a model depth of 5, and an embedding dimension of 512. However, the latter implementation is outperformed by the former implementation on the large-scale NTU 60 xsub dataset.

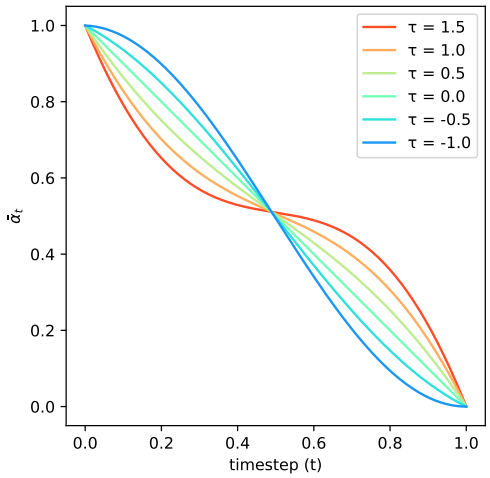

Figure 1. Noise schedules controlled by τ.

2.2. Generation Metrics

In this section, we describe the metrics for unconditional motion generation, *i.e.*, FID, KID, diversity, and precision/recall, which are all based on the latent features of generated samples. We choose the MAMP encoder with a hidden dimension of 256 pre-trained on NTU 60 xsub for feature extraction. We randomly sample 1000 sequences for calculating metric scores.

FID. Fréchet inception distance (FID) [4], borrowed from the image domain, evaluates the quality of samples generated by generative models. FID fits a Gaussian distribution to the feature distribution of generated data and that of real data (from the testing set), and then computes the Fréchet distance between those Gaussians, defined as:

$$\text{FID} = \| \mu - \mu' \|^2 + \text{Tr}(\Sigma + \Sigma' - 2(\Sigma\,\Sigma')^{\frac{1}{2}}), \tag{12}$$

where μ, μ' denote the means of the real and generated data, and Σ, Σ' denote the covariance matrices of the real and generated data. A lower FID implies better results.

KID. Kernel Inception Distance (KID) [2] is a metric similar to FID. KID compares skewness as well as the mean and variance by using a polynomial kernel to calculate the MMD between feature distributions. A lower FID implies better results.

Diversity. Diversity measures the variance of all generated samples by calculating the mean L2 distance between the features. A diversity closer to the diversity of real data implies better results. Since all the evaluated models yield lower diversity than real data, we mark it with an upwards pointing arrow indicating that a higher diversity is better.

$$\text{Diversity} = \frac{1}{N}\sum_{i=1}^{N} \| \boldsymbol{z}_i - \boldsymbol{z}'_i \|_2. \tag{13}$$

Precision and Recall. Precision measures the probability that a generated sample falls within the real distribution, while recall measures the probability that a real sample falls within the generated data distribution. Precision and recall are closely associated with fidelity and diversity, respectively. Higher precision and recall imply better results.

2.3. Noise Schedule

The cosine schedule, proposed by [1], is a widely-used noise schedule in diffusion models, defined as:

$$\bar{\alpha}_t^{cos} = cos\left(\frac{\pi}{2} \cdot \left(\frac{t + 0.008}{1.008}\right)^2\right), \tag{14}$$

where $t \in [0, 1]$ denotes the timestep (divided by the total timestep T). In MacDiff, we define a series of noise schedules controlled by a hyper-parameter τ, defined as:

$$\bar{\alpha}_t(\tau) = (1 + \tau) \cdot (1 - t) - \tau \cdot \bar{\alpha}_t^{cos}. \tag{15}$$

$\tau = -1$ is the cosine schedule, and $\tau = 1$ is the inverse-cosine schedule proposed by [7]. $\tau = 1$ corresponds to a schedule linear for $\bar{\alpha}_t$, which is different from the linear schedule defined by [6] in that the latter is linear for β_t. Fig. 1 shows the visualization of different noise schedules.

3. Qualitative Results and Discussion

3.1. Motion Generation

In Fig. 2, we provide unconditional motion generation results of MacDiff. The number of frames is 120, and we plot every 12 frames for visualization. As shown in Fig. 2, the MacDiff diffusion decoder is capable of generating diverse and high-quality skeleton data. In this paper, we do not explore other generation settings such as class-conditioned or text-conditioned generation. However, our conditional diffusion framework is able to integrate other types of condition, and we will leave this to future work.

3.2. Motion Reconstruction

We provide motion reconstruction results of MacDiff. We randomly occlude 10 frames out of 120 frames for our model to reconstruct, which is outlined in red. We plot every 4 frames and show a part of the sequence including the occluded frames. As shown in Fig. 3, MacDiff can accurately reconstruct static actions and smooth motions. MacDiff may also introduce some semantically reasonable changes, which we attribute to the stochasticity and imagination ability of diffusion models.

3.3. One-Step Denoising for Data Augmentation

In our proposed diffusion-based data augmentation, we first pre-calculate the representations of labeled samples and then perform one-step denoising from some timestep t_s guided by the representations. In addition to the quantitative ablation study, we provide qualitative results to verify our choices for (1) conditional denoising (using representations) for preserving labels, (2) a medium t_s, and (3) one-step denoising rather than multi-step sampling.

In Fig. 4, we compare the results of conditional denoising and unconditional denoising at $t_s = 500$. Samples generated with unconditional denoising lose some important label-relevant semantics, which are preserved via conditional sampling. For example, in "vomiting" the hands should be close to the mouse, in "chest pain" the hands should be around the chest, and in "kicking something" the feet should have obvious motion. Therefore, the guidance of the condition ensures that the generated samples are label-preserving.

In Fig. 5, we compare the results of different t_s. A small t_s brings little difference compared with the original data and fails to serve as a data augmentation. A large t_s may change the semantics of the augmented data, thus changing the labels. For example, the "brush hair" action is mainly identified by hand movements, which are blurred for $t_s = 900$ in Fig. 5.

In Fig. 6, we compare the results of one-step denoising and multi-step denoising (sampling). The two methods yield similar results that are difficult to differentiate in terms of effectiveness. However, one-step denoising has a smaller computational cost, which allows for generating diverse augmented data at different epochs and yields better performance. Therefore, we adopt one-step denoising for data augmentation in semi-supervised protocols.

4. More Ablation Study

4.1. Conditioning Module Design

As shown in Tab. 1, we compare different designs of the conditioning module. Apart from the AdaLN layer, we explore two other designs. In self-attention conditioning, the representation token(s) $\boldsymbol{z}$ are directly concatenated with the input tokens of the decoder and guide the denoising process via the self-attention layers. $\boldsymbol{z}$ can either be the global representation $\boldsymbol{z}_{global}$ or the local representations $\boldsymbol{z}_{local}$, which correspond to the self-attention (global) and self-attention (local) settings in Tab. 1, respectively. In cross-attention conditioning, a cross-attention layer is added after each self-attention layer, where the local representation tokens attend to the decoder tokens.

The self-attention (global) conditioning performs significantly worse compared to its counterparts, indicating that a single global token fails to provide enough guidance for the decoder via self-attention. The self-attention (local) is similar to the

MAE [5] decoder design and requires the representations to contain more low-level details, which explains its high fine-tuning performance [13]. However, we aim to obtain a meaningful global representation, and thus we do not adopt this conditioning design. The cross-attention conditioning also performs badly, failing to learn a meaningful representation. MacDiff with AdaLN as conditioning modules works best.

Table 1. Ablation study on the conditioning module. We report results on NTU 60 xsub under the linear and fine-tuning evaluation protocol.

Conditioning module	Linear	Fine-tuning
Self-attention (global)	82.1	91.7
Self-attention (local)	84.8	**93.2**
Cross-attention	82.4	92.3
AdaLN	**86.4**	92.7

4.2. Diffusion Prediction Target

We compare different prediction targets of diffusion, *i.e.*, $\boldsymbol{\epsilon}$-prediction, $\boldsymbol{x}_0$-prediction and $\boldsymbol{v}$-prediction [9]. Note that we focus on the effects of prediction targets on representation learning rather than generation in this paper. As shown in Tab. 2, $\boldsymbol{\epsilon}$-prediction performs better than $\boldsymbol{x}_0$-prediction and $\boldsymbol{v}$-prediction.

Table 2. Ablation study on the prediction target. We report results on NTU 60 xsub under the linear and fine-tuning evaluation protocol.

Prediction	Linear	Fine-tuning
$\boldsymbol{\epsilon}$	**86.4**	**92.7**
$\boldsymbol{x}_0$	83.8	92.0
$\boldsymbol{v}$	84.5	92.4

4.3. Global-Local Conditioning

In MacDiff, we obtain the condition $\boldsymbol{z}$ by unshuffling the local representations $\boldsymbol{z}_{local}$ and filling the masked positions with $\boldsymbol{z}_{global}$. We compare this global-local conditioning with global-only conditioning, which simply broadcasts $\boldsymbol{z}_{global}$ to obtain $\boldsymbol{z}$. By allowing the decoder to leverage and directly optimize the local representations, global-local conditioning yields significantly better results than its counterpart.

Table 3. Ablation study on the global-local conditioning strategy. We report results on NTU 60 xsub under the linear and fine-tuning evaluation protocol.

Global-local conditioning	Linear	Fine-tuning
	83.7	91.3
✓	**86.4**	**92.7**

Figure 2. Unconditional motion generation results. We plot every 12 frames.

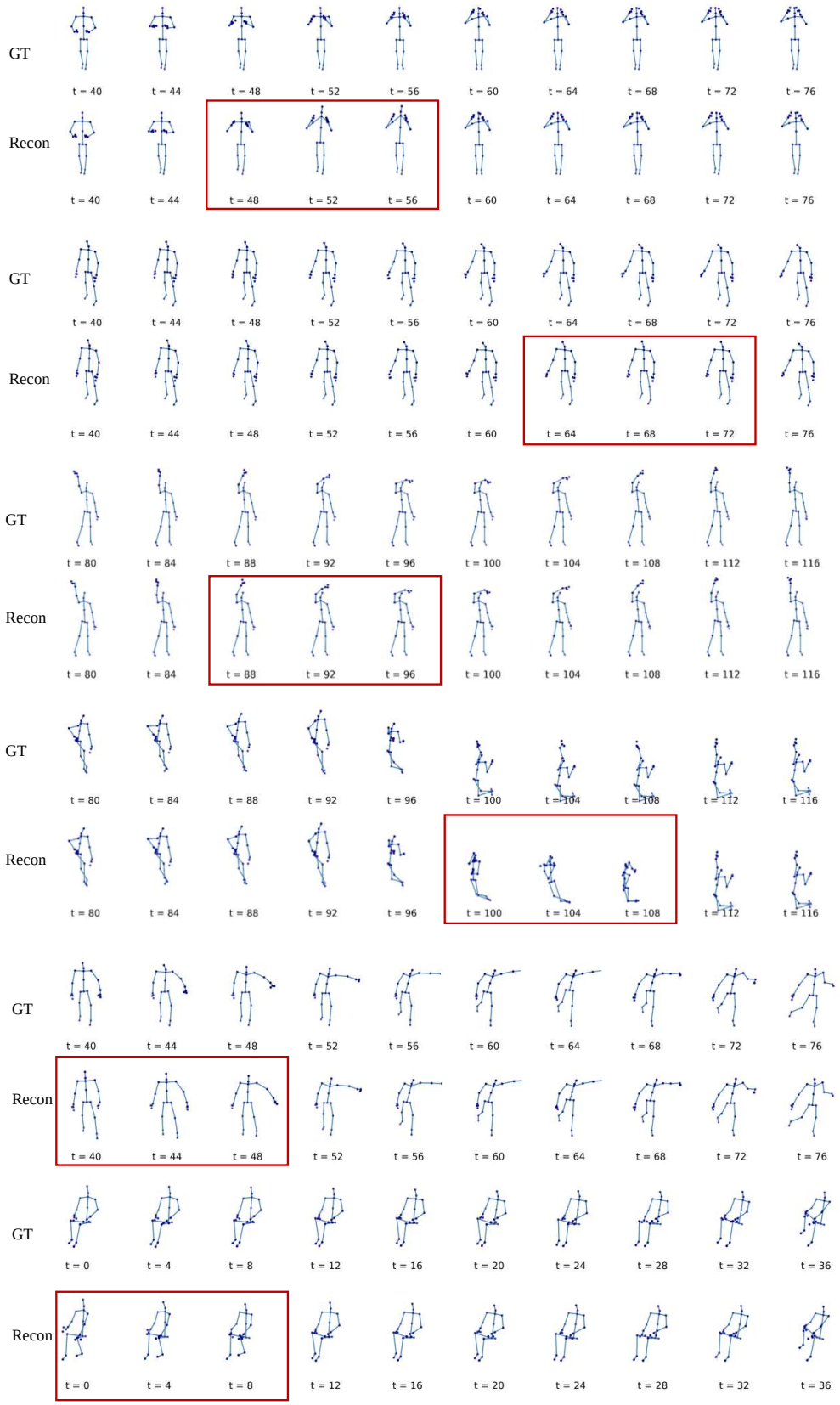

Figure 3. Motion reconstruction results. GT and Recon represents the ground truth data and reconstructed data, respectively, and the reconstructed frames are outlined in red. We plot every 4 frames.

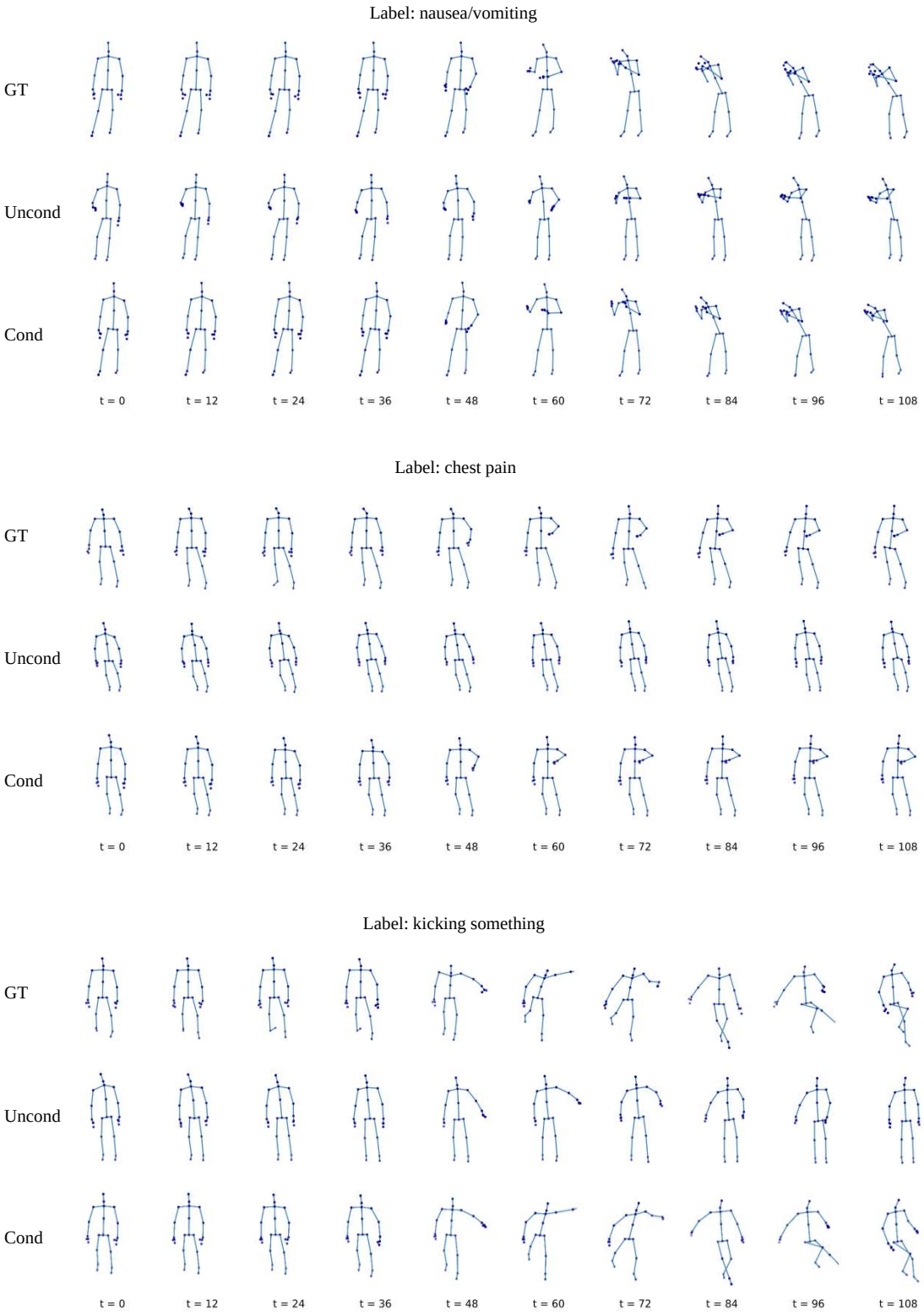

Figure 4. Comparison of unconditional denoising results and conditional denoising results. We use one-step denoising from timestep $t_s = 500$. Conditional denoising is significantly better at preserving the label-relevant semantics while introducing some detail changes. We plot every 12 frames.

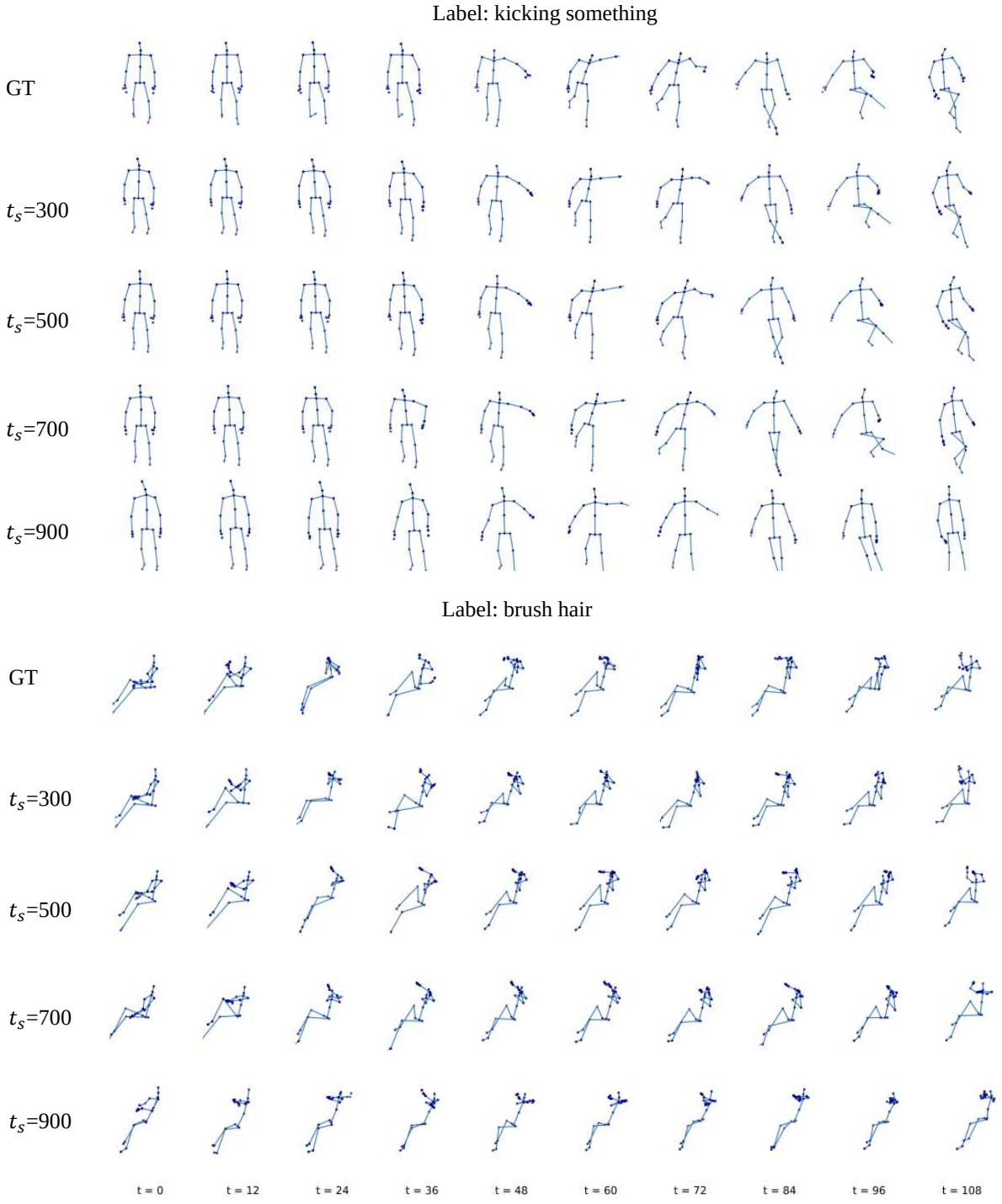

Figure 5. Comparison of different timesteps t_s for one-step denoising. A small t_s brings little difference compared with the original data, while a large t_s may change the semantics of the augmented data. But generally, our method is robust to the choice of t_s, so we simply set a medium timestep $t_s = 500$ by default. We plot every 12 frames.

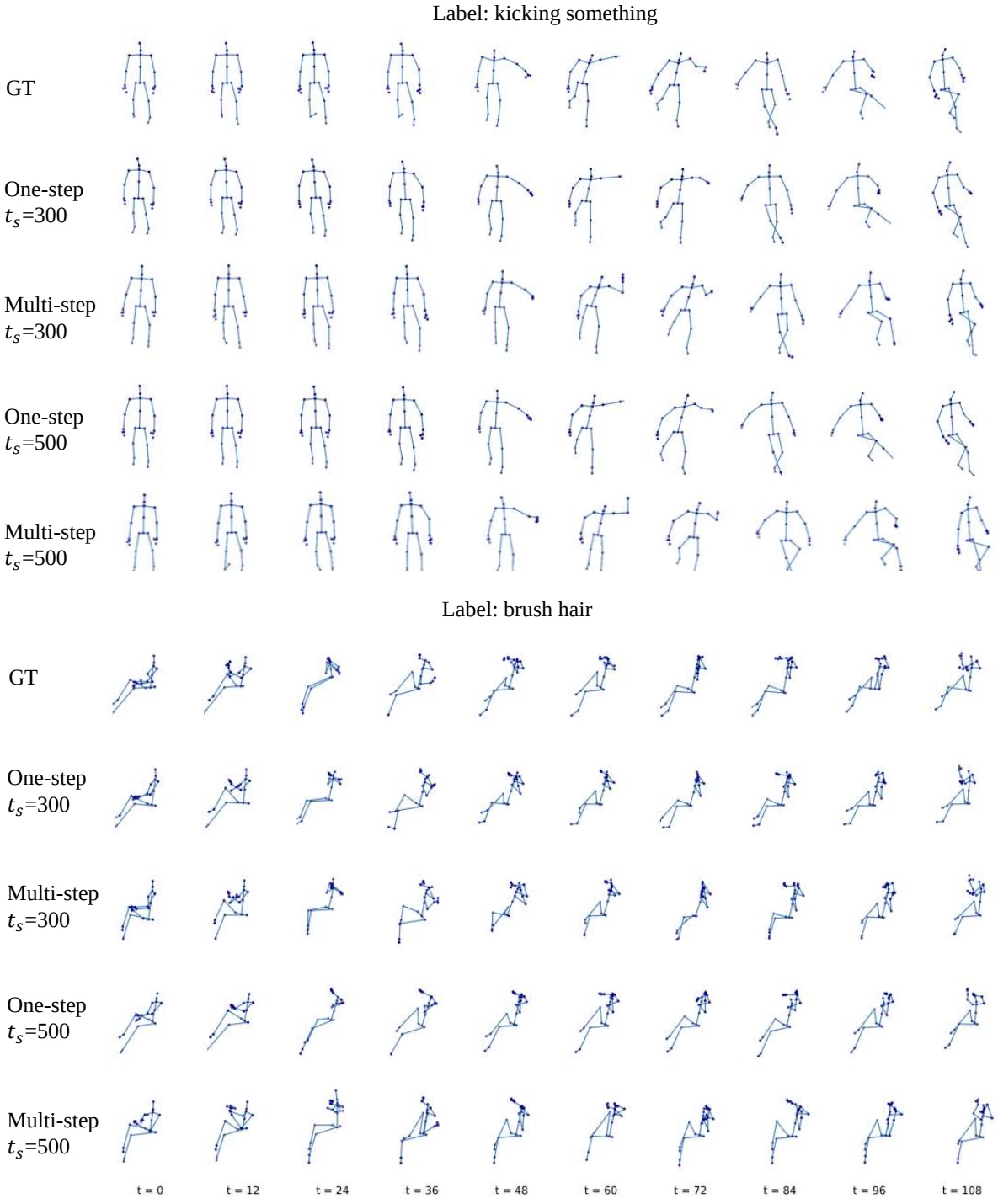

Figure 6. Comparison of one-step denoising and multi-step denoising. The two methods yield similar results that are difficult to differentiate in terms of effectiveness. Therefore, we adopt one-step denoising with smaller computational cost, which allows for generating diverse augmented data at different epochs. We plot every 12 frames.

References

[1] Prafulla Dhariwal Alex Nichol. Improved denoising diffusion probabilistic models. *arXiv preprint arXiv:2102.09672*, 2021.

[2] Mikołaj Binkowski, Dougal J. Sutherland, Michael Arbel, and Arthur Gretton. Demystifying MMD GANs. In *ICLR*, 2018.

[3] Meir Feder and Neri Merhav. Relations between entropy and error probability. *IEEE Transactions on Information theory*, 40(1):259—-266, 1994.

[4] Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. Action2Motion: Conditioned generation of 3d human motions. In *ACM MM*, 2020.

[5] Kaiming He, Xinlei Chen, Saining Xie, Yanghao Li, Piotr Dollár, and Ross Girshick. Masked autoencoders are scalable vision learners. In *CVPR*, 2022.

[6] Jonathan Ho, Ajay Jain, and Pieter bbeel. Denoising diffusion probabilistic models. *arXiv preprint arXiv:2006.11239*, 2020.

[7] Drew A. Hudson, Daniel Zoran, Mateusz Malinowski, Andrew K. Lampinen, Andrew Jaegle, James L. McClelland, Loic Matthey, Felix Hill, and Alexander Lerchner. SODA: Bottleneck diffusion models for representation learning. *arXiv preprint arXiv:2311.17901*, 2023.

[8] Yunyao Mao, Jiajun Deng, Wengang Zhou, Yao Fang, Wanli Ouyang, and Houqiang Li. Masked motion predictors are strong 3D action representation learners. In *ICCV*, 2023.

[9] Tim Salimans and Jonathan Ho. Progressive distillation for fast sampling of diffusion models. *arXiv preprint arXiv:2202.00512*, 2022.

[10] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv preprint arXiv:2010.02502*, 2020.

[11] Guy Tevet, Sigal Raab, Brian Gordon, Yoni Shafir, Daniel Cohen-or, and Amit Haim Bermano. Human motion diffusion model. In *ICLR*, 2023.

[12] Wenhan Wu, Yilei Hua, Ce Zheng, Shiqian Wu, Chen Chen, and Aidong Lu. SkeletonMAE: Spatial-temporal masked autoencoders for self-supervised skeleton action recognition. *arXiv preprint arXiv:2209.02399*, 2022.

[13] Qi Zhang, Yifei Wang, and Yisen Wang. How mask matters: Towards theoretical understandings of masked autoencoders. *arXiv preprint arXiv:2210.08344*, 2022.