Local Action-Guided Motion Diffusion Model for Text-to-Motion Generation Supplementary Material

Peng Jin[1,2,3], Hao Li[1,2,3], Zesen Cheng[1,3], Kehan Li[1,3], Runyi Yu[1,3], Chang Liu[4*], Xiangyang Ji[4], Li Yuan[1,2,3*], and Jie Chen[1,2,3]

[1] School of Electronic and Computer Engineering, Peking University, Shenzhen, China
[2] Peng Cheng Laboratory, Shenzhen, China
[3] AI for Science (AI4S)-Preferred Program, Peking University Shenzhen Graduate School, Shenzhen, China
[4] Department of Automation and BNRist, Tsinghua University, Beijing, China
jp21@stu.pku.edu.cn, liuchang2022@tsinghua.edu.cn, yuanli-ece@pku.edu.cn

Abstract. This appendix provides implementation details (Appendix A), several additional analyses (Appendix B), additional discussions (Appendix C), and details of motion representations and metric definitions (Appendix D).

A Implementation Details

A.1 Semantic Graph Parsing

To obtain actions, action attributes, and the semantic role of each attribute in relation to the corresponding action, we implement a semantic parser of motion descriptions based on a semantic role parsing toolkit [16][‡]. Specifically, the parser, when given a motion description, identifies the verbs within the sentence, extracts attribute phrases corresponding to each verb, and determines the semantic role of each attribute phrase. In the semantic graph, the entire sentence is treated as the global motion node. Verbs serve as action nodes and are connected to the motion node through direct edges, facilitating implicit learning of temporal relationships among different actions during graph reasoning. Attribute phrases represent specific nodes, linked to action nodes. The edge type between an action node and a specific node is determined by the semantic role of the specifics in relation to the action. As shown in Tab. A, we extract three types (motions, actions, and specifics) of nodes and twelve types of edges to depict various associations among these nodes.

* Corresponding author: Li Yuan, Chang Liu.
‡ https://allenai.org/allennlp

Table A: Node types and edge types in the semantic graph. Each edge type corresponds to a type of semantic role.

Node type	Description
Motion	global motion description
Action	verb
Specific	attribute of action
Edge type	**Description**
ARG0	agent
ARG1	patient
ARG2	instrument, benefactive
ARG3	start point
ARG4	end point
ARGM-LOC	location (where)
ARGM-MNR	manner (how)
ARGM-TMP	time (when)
ARGM-DIR	direction (where to/from)
ARGM-ADV	miscellaneous
ARGM-MA	motion-action dependencies
OTHERS	other argument types, *e.g.*, action

A.2 Implementation Details for Different Datasets

Following MLD [2], we utilize a frozen text encoder of the CLIP-ViT-L-14 [14] model for text representation. The dimension of node representation is set to 768. The dimension of latent embedding is set to 256.

For the motion variational autoencoder, motion encoder and decoder all consist of 9 layers and 4 heads with skip connection [15]. To generate motion from coarse to fine step by step, we encode motion independently into three latent representation spaces $\boldsymbol{z}^m \in \mathbb{R}^{Q^m \times D'}$, $\boldsymbol{z}^a \in \mathbb{R}^{Q^a \times D'}$ and $\boldsymbol{z}^s \in \mathbb{R}^{Q^s \times D'}$, where the number of tokens gradually increases, *i.e.*, $Q^m \leq Q^a \leq Q^s$. We set the token sizes Q^m to 2, Q^a to 4, and Q^s to 8.

Following previous works, our denoiser network is learned with classifier-free diffusion guidance [7]. The classifier-free diffusion guidance improves the quality of samples by reducing diversity in conditional diffusion models. Concretely, it learns both the conditioned and the unconditioned distribution (10% dropout [18]) of the samples. Finally, we perform a linear combination in the following manner, which is formulated as:

$$\begin{aligned}
\widehat{\boldsymbol{\epsilon}^m} &= \alpha\phi_m(\boldsymbol{z}^m, t^m, \mathcal{V}^m) + (1-\alpha)\phi_m(\boldsymbol{z}^m, t^m, \varnothing), \\
\widehat{\boldsymbol{\epsilon}^a} &= \alpha\phi_a(\boldsymbol{z}^a, t^a, [\mathcal{V}^m, \mathcal{V}^a, \boldsymbol{z}^m]) + (1-\alpha)\phi_a(\boldsymbol{z}^a, t^a, \varnothing), \\
\widehat{\boldsymbol{\epsilon}^s} &= \alpha\phi_s(\boldsymbol{z}^s, t^s, [\mathcal{V}^m, \mathcal{V}^a, \mathcal{V}^s, \boldsymbol{z}^a]) + (1-\alpha)\phi_s(\boldsymbol{z}^s, t^s, \varnothing),
\end{aligned} \tag{A}$$

Table B: Evaluation of Inference time costs on the HumanML3D test set. We evaluate the average time per sample with different diffusion schedules and FID. "↓" denotes that lower is better. "✘" denotes that this method does not apply this parameter. We set T^m, T^a, and T^s to 50 for optimal performance. It is important to note that because local actions can be pre-generated, the inference time costs do not encompass the time taken for the pre-generation of local actions.

Methods	Diffusion Steps			Time (s) ↓
	T^m	T^a	T^s	
1000 diffusion steps with DDPM [6]				
MDM [20] ICLR23	1000	✘	✘	178.7699
MotionDiffuse [22] TPAMI24	1000	✘	✘	5.5045
50 diffusion steps with DDIM [17]				
MLD [2] CVPR23	50	✘	✘	0.7471
GuidedMotion (Ours)	20	15	15	0.8115
GuidedMotion (Ours)	15	20	15	0.7684
GuidedMotion (Ours)	15	15	20	0.7603
150 diffusion steps with DDIM [17]				
MLD [2] CVPR23	150	✘	✘	2.4998
GuidedMotion (Ours)	50	50	50	2.5189

Where α is the guidance scale and $\alpha > 1$ can strengthen the effect of guidance [2]. We set α to 7.5 in practice following MLD.

All our models are trained with the AdamW [9, 12] optimizer using a fixed learning rate of 1e-4. We use 4 Tesla V100 GPUs for the training, and there are 128 samples on each GPU, so the total batch size is 512. The number of diffusion steps of each level is 1,000 during training, and the step sizes β_t are scaled linearly from 8.5×1e-4 to 0.012. We keep running a similar number of iterations on different data sets. For the HumanML3D dataset, the model is trained for 6,000 epochs during the motion variational autoencoder stage and 3,000 epochs during the diffusion stage. For the KIT dataset, the model is trained for 30,000 epochs during the motion variational autoencoder stage and 15,000 epochs during the diffusion stage.

B Additional Analyses

B.1 Analysis of the Inference Time

In Tab. B, we provide the evaluation of inference time costs on the HumanML3D test set. It is important to note that because local actions can be pre-generated, the inference time costs presented in Tab. B do not encompass the time taken for the pre-generation of local actions. As shown in Tab. B, despite decomposing the diffusion process into three parts, our method demonstrates efficiency

Table C: Evaluation of the motion VAE models on the motion part. "↑" denotes that higher is better. "↓" denotes that lower is better. "→" denotes that results are better if the metric is closer to the real motion. Among the three levels, the performance of the specific level is the best.

Methods	Token Size	R-Precision ↑			FID ↓	Diversity →
		Top-1	Top-2	Top-3		
VAE models on the HumanML3D dataset						
Real Motion	-	0.511	0.703	0.797	0.002	9.503
Motion Level	2	0.498	0.692	0.791	1.906	9.675
Action Level	4	0.514	0.703	0.793	0.068	**9.610**
Specific Level	8	**0.525**	**0.708**	**0.800**	**0.019**	9.863
VAE models on the KIT dataset						
Real Motion	-	0.424	0.649	0.779	0.031	11.08
Motion Level	2	**0.431**	0.623	0.745	1.196	10.66
Action Level	4	0.413	**0.644**	**0.770**	0.396	10.85
Specific Level	8	0.414	0.640	0.760	**0.361**	**10.86**

comparable to one-stage diffusion methods during the inference stage. This is achievable by controlling the total number $T^m + T^a + T^s$ of iterations, ensuring it aligns with that of one-stage diffusion methods. Our method exhibits inference speeds comparable to existing methods with the same total number of diffusion steps, underscoring the efficiency of our method.

B.2 Analysis of the motion VAE models

We provide the evaluation of the motion VAE models. In Tab. C, we show the results on the HumanML3D test set and KIT test set. Among the three levels, the performance of the specific level is the best, which indicates that increasing the token size enhances the reconstruction ability of the motion VAE models. We take the output at the specific level as the final result and use the motion decoder to decode the latent representation into the motion sequence.

B.3 Additional Visualization Results

In Fig. A, we provide additional qualitative motion results. These results demonstrate that our method can generate diverse and accurate motion sequences.

C Additional Discussions

C.1 Limitations of our Work

While our method has shown notable advancements, there exist several limitations that warrant further investigation. (1) Although our method can generate

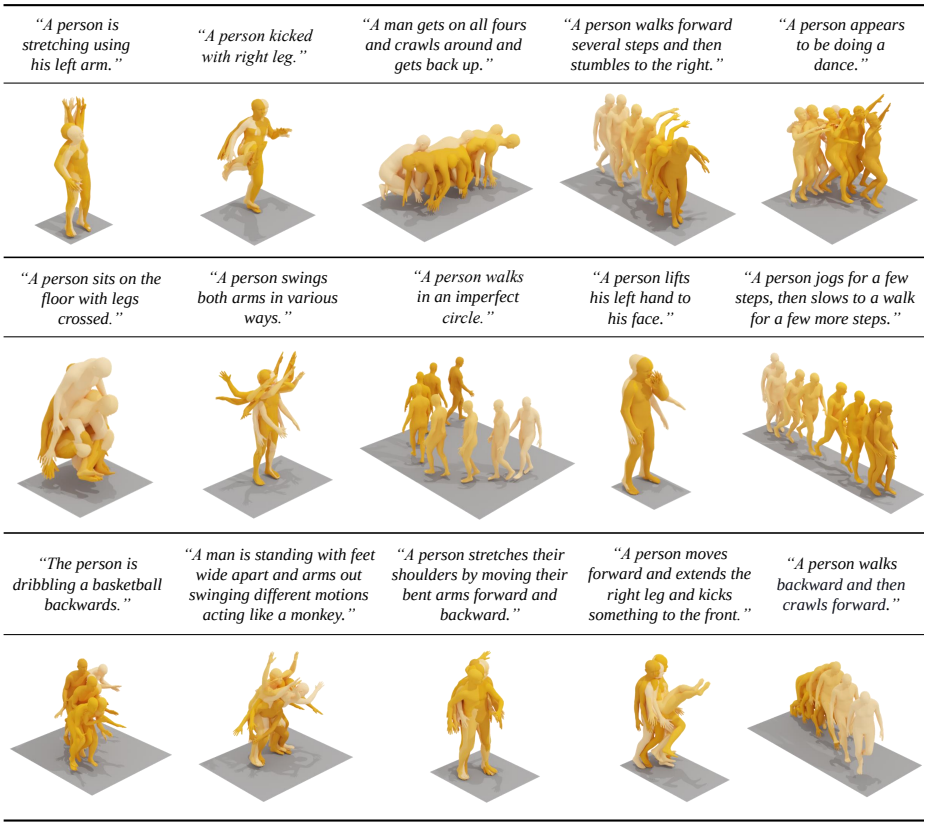

Fig. A: Additional qualitative motion results are generated with text prompts. The darker colors indicate the later in time. These results demonstrate that our method can generate diverse and accurate motion sequences.

results of arbitrary lengths, it still adheres to the maximum length observed in the dataset. Modeling a continuous human motion with temporal consistency introduces a fascinating challenge. (2) Given that our method employs a diffusion process in the motion latent space rather than directly on the raw motion sequences, it is better suited for high-level motion editing rather than low-level motion editing, such as modifying the position of a single joint. Exploring low-level motion editing within latent space holds great promise and poses an exciting avenue for future research. (3) Our method inherits the inherent randomness of diffusion models. While this trait enriches diversity, it is crucial to recognize that it may occasionally result in less desirable outcomes. (4) The human motion synthesis capabilities of our method are constrained by the performance of the pre-trained motion variational autoencoders, which we discuss in experiments (Tab. C). This defect also exists in the existing methods, such as MLD [2] and T2M-GPT [21], which also use motion variational autoencoder. Furthermore,

delving into the realm of a more efficient motion latent space holds significant promise as a compelling avenue for future research. (5) Though our method brings negligible extra cost to computations, it is still limited by the slow inference speed of existing diffusion models. However, with the development of diffusion models, we anticipate a progressive mitigation of this limitation. We discuss the inference time in Tab. B. This defect also exists in the existing state-of-the-art methods, such as MDM [20], MLD [2], GraphMotion [8], and ReMoDiffuse [23], which also use diffusion models.

C.2 Future Work

In this work, we focus on improving the controllability of text-driven human motion generation. Our method empowers users to combine preferred local actions freely, thereby generating motions that resonate with their mental imagery. In future research, we aim to explore additional avenues for controlling motion generation. Particularly, delving into low-level motion editing within the latent space holds significant promise and presents an exciting direction.

D Motion Representations and Metric Definitions

D.1 Motion Representations

Motion representation can be summarized into the following four categories, and we follow the previous work of representing motion in latent space.

Latent Format. Following previous works [2, 13, 21], we encode the motion into the latent space with a motion variational autoencoder [10].

HumanML3D Format. HumanML3D [4] proposes a motion representation $\boldsymbol{x}^{1:L}$ inspired by motion features in character control. Specifically, the i_{th} pose $\boldsymbol{x}^i$ is defined by a tuple consisting of the root angular velocity $r^a \in \mathbb{R}$ along the Y-axis, root linear velocities $(r^x, r^z \in \mathbb{R})$ on the XZ-plane, root height $r^y \in \mathbb{R}$, local joints positions $\boldsymbol{j^p} \in \mathbb{R}^{3N_j}$, velocities $\boldsymbol{j^v} \in \mathbb{R}^{3N_j}$, and rotations $\boldsymbol{j^r} \in \mathbb{R}^{6N_j}$ in root space, and binary foot-ground contact features $\boldsymbol{c^f} \in \mathbb{R}^4$ obtained by thresholding the heel and toe joint velocities. Here, N_j denotes the joint number. Finally, the HumanML3D format can be defined as:

$$\boldsymbol{x}^i = \{r^a, r^x, r^z, r^y, \boldsymbol{j^p}, \boldsymbol{j^v}, \boldsymbol{j^r}, \boldsymbol{c^f}\}. \tag{B}$$

SMPL-based Format. SMPL [11] is one of the most widely used parametric human models. SMPL and its variants propose motion parameters θ and shape parameters β. $\theta \in \mathbb{R}^{3\times 23+3}$ is rotation vectors for 23 joints and a root, while β represents the weights for linear blended shapes. The global translation r is also incorporated to formulate the representation as follows:

$$\boldsymbol{x}^i = \{r, \theta, \beta\}. \tag{C}$$

MMM Format. Master Motor Map [19] (MMM) representations propose joint angle parameters based on a uniform skeleton structure with 50 degrees of

freedom (DoFs). In text-to-motion tasks, recent methods [1, 3, 13] converts joint rotation angles into $J = 21$ joint XYZ coordinates. Given the global trajectory t_{root} and $p_m \in \mathbb{R}^{3J}$, the preprocessed representation is formulated as:

$$\boldsymbol{x}^i = \{p_m, t_{root}\}. \tag{D}$$

D.2 Metric Definitions

Following previous works, we use the following five metrics to measure the performance of the model. Note that global representations of motion and text descriptions are first extracted with the pre-trained network in [4].

R-Precision. Under the feature space of the pre-trained network in [4], given one motion sequence and 32 text descriptions (1 ground-truth and 31 randomly selected mismatched descriptions), motion-retrieval precision calculates the text and motion Top 1/2/3 matching accuracy.

Frechet Inception Distance (FID). We measure the distribution distance between the generated and real motion using FID [5] on the extracted motion features [4]. The FID is calculated as:

$$\text{FID} = \|\mu_{gt} - \mu_{pred}\|^2 - \text{Tr}(\Sigma_{gt} + \Sigma_{pred} - 2(\Sigma_{gt}\Sigma_{pred})^{\frac{1}{2}}), \tag{E}$$

where Σ is the covariance matrix. Tr denotes the trace of a matrix. μ_{gt} and μ_{pred} are the mean of ground-truth motion features and generated motion features.

Multimodal Distance (MM-Dist). Given N randomly generated samples, we calculate the average Euclidean distances between each text feature f_t and the generated motion feature f_m from that text. The multimodal distance is calculated as:

$$\text{MM-Dist} = \frac{1}{N}\sum_{i=1}^{N} \|f_{t,i} - f_{m,i}\|, \tag{F}$$

where $f_{t,i}$ and $f_{m,i}$ are the features of the i_{th} text-motion pair.

Diversity. All generated motions are randomly sampled to two subsets (*i.e.*, $\{\boldsymbol{x}_1, \boldsymbol{x}_2, ..., \boldsymbol{x}_{X_d}\}$ and $\{\boldsymbol{x}_1^{'}, \boldsymbol{x}_2^{'}, ..., \boldsymbol{x}_{X_d}^{'}\}$) of the same size X_d. Then, we extract motion features [4] and compute the average Euclidean distances between the two subsets:

$$\text{Diversity} = \frac{1}{X_d}\sum_{i=1}^{\boldsymbol{X}_d} \|\boldsymbol{x}_i - \boldsymbol{x}_i^{'}\|. \tag{G}$$

Multimodality (MModality). We randomly sample a set of text descriptions with size J_m from all descriptions. For each text description, we generate $2 \times X_m$ motion sequences, forming X_m pairs of motions. We extract motion features and calculate the average Euclidean distance between each pair. We report the average of all text descriptions. We define features of the j_{th} pair of the i_{th} text description as $(\boldsymbol{x}_{j,i}, \boldsymbol{x}_{j,i}^{'})$. The multimodality is calculated as:

$$\text{MModality} = \frac{1}{J_m \times X_m}\sum_{j=1}^{J_m}\sum_{i=1}^{X_m} \|\boldsymbol{x}_{j,i} - \boldsymbol{x}_{j,i}^{'}\|. \tag{H}$$

References

1. Ahuja, C., Morency, L.P.: Language2pose: Natural language grounded pose forecasting. In: 3DV. pp. 719–728 (2019)
2. Chen, X., Jiang, B., Liu, W., Huang, Z., Fu, B., Chen, T., Yu, J., Yu, G.: Executing your commands via motion diffusion in latent space. In: CVPR. pp. 18000–18010 (2023)
3. Ghosh, A., Cheema, N., Oguz, C., Theobalt, C., Slusallek, P.: Synthesis of compositional animations from textual descriptions. In: ICCV. pp. 1396–1406 (2021)
4. Guo, C., Zou, S., Zuo, X., Wang, S., Ji, W., Li, X., Cheng, L.: Generating diverse and natural 3D human motions from text. In: CVPR. pp. 5152–5161 (2022)
5. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: GANs trained by a two time-scale update rule converge to a local nash equilibrium. In: NeurIPS (2017)
6. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. In: NeurIPS. pp. 6840–6851 (2020)
7. Ho, J., Salimans, T.: Classifier-free diffusion guidance. arXiv preprint arXiv:2207.12598 (2022)
8. Jin, P., Wu, Y., Fan, Y., Sun, Z., Wei, Y., Yuan, L.: Act as you wish: Fine-grained control of motion diffusion model with hierarchical semantic graphs. In: NeurIPS (2023)
9. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)
10. Kingma, D.P., Welling, M.: Auto-encoding variational bayes. arXiv preprint arXiv:1312.6114 (2013)
11. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: Smpl: A skinned multi-person linear model. TOG **34**(6), 1–16 (2015)
12. Loshchilov, I., Hutter, F., et al.: Fixing weight decay regularization in adam. arXiv preprint arXiv:1711.05101 (2017)
13. Petrovich, M., Black, M.J., Varol, G.: Temos: Generating diverse human motions from textual descriptions. In: ECCV. pp. 480–497 (2022)
14. Radford, A., Kim, J.W., Hallacy, C., Ramesh, A., Goh, G., Agarwal, S., Sastry, G., Askell, A., Mishkin, P., Clark, J., Krueger, G., Sutskever, I.: Learning transferable visual models from natural language supervision. In: ICML. pp. 8748–8763 (2021)
15. Ronneberger, O., Fischer, P., Brox, T.: U-Net: Convolutional networks for biomedical image segmentation. In: MICCAI. pp. 234–241 (2015)
16. Shi, P., Lin, J.: Simple bert models for relation extraction and semantic role labeling. arXiv preprint arXiv:1904.05255 (2019)
17. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. arXiv preprint arXiv:2010.02502 (2020)
18. Srivastava, N., Hinton, G., Krizhevsky, A., Sutskever, I., Salakhutdinov, R.: Dropout: a simple way to prevent neural networks from overfitting. JMLR **15**(1), 1929–1958 (2014)
19. Terlemez, Ö., Ulbrich, S., Mandery, C., Do, M., Vahrenkamp, N., Asfour, T.: Master motor map (mmm)—framework and toolkit for capturing, representing, and reproducing human motion on humanoid robots. In: 2014 IEEE-RAS International Conference on Humanoid Robots. pp. 894–901 (2014)
20. Tevet, G., Raab, S., Gordon, B., Shafir, Y., Cohen-Or, D., Bermano, A.H.: Human motion diffusion model. In: ICLR (2023)

21. Zhang, J., Zhang, Y., Cun, X., Huang, S., Zhang, Y., Zhao, H., Lu, H., Shen, X.: T2m-gpt: Generating human motion from textual descriptions with discrete representations. In: CVPR (2023)
22. Zhang, M., Cai, Z., Pan, L., Hong, F., Guo, X., Yang, L., Liu, Z.: Motiondiffuse: Text-driven human motion generation with diffusion model. TPAMI (2024)
23. Zhang, M., Guo, X., Pan, L., Cai, Z., Hong, F., Li, H., Yang, L., Liu, Z.: Remodiffuse: Retrieval-augmented motion diffusion model. In: ICCV (2023)