Skeleton-in-Context: Unified Skeleton Sequence Modeling with In-Context Learning
–Supplementary Material–

Table 1. Importance of our proposed task-guided prompt (TGP) and task-unified prompt (TUP) employed in Skeleton-in-Context (SiC). TGP and TUP are important designs as they address the challenge of multi-task training. Please refer to Sec. B for more discussion.

Methods	MP. (AMASS)						PE. (H3.6M)		JC. (3DPW)			FPE. (H3.6M)		
	MPJPE ↓						MPJPE↓	N-MPJPE↓	MPJPE ↓			MPJPE ↓		
	80ms	160ms	200ms	320ms	400ms	Avg.	Avg.	Avg.	40%	60%	Avg.	200ms	300ms	Avg.
Task-Specific Model: one architecture for one task														
SiC (w/o TGP&TUP)	57.7	46.2	41.6	36.7	47.7	46.0	56.5	45.4	49.7	59.8	54.7	62.5	72.6	67.6
Multi-Task Model: one architecture (w. multiple task-specific heads) for multiple tasks														
SiC (w/o TGP&TUP)	62.9	49.8	44.3	36.3	46.5	47.9	59.1	47.4	51.3	61.6	56.5	76.4	86.8	81.6
In-Context Model: one task-agnostic architecture for multiple tasks														
Static-SiC	**11.0**	19.0	22.7	32.6	37.8	24.6	**51.6**	**41.8**	29.8	44.7	37.2	59.8	67.2	63.5
Dynamic-SiC	**11.0**	**18.9**	**22.6**	**32.2**	**37.4**	**24.4**	51.8	42.5	**29.5**	**44.0**	**36.8**	**59.2**	**66.5**	**62.9**

Overview. The supplementary material includes:

- Section A presents more details on our experiments and a more detailed description of the datasets.
- Section B verifies the effectiveness of our Skeleton-in-Context on multi-task synergistic training.
- Section C provides more detailed results both quantitatively and qualitatively.

A. Experimental Details

A.1. Implementation Details

We implement the proposed Skeleton-in-Context model with the number of layers $N = 5$, number of attention heads $H = 8$, and hidden feature dimension $C = 256$. For each prompt input/target and query input/target, the sequence length is $F = 16$. We implement Skeleton-in-Context with PyTorch. In the default setting, during both training and evaluation, for each query pair, we randomly select a prompt pair from the training set of the same task as the query pair. We use an AdamW optimizer with a linearly decaying learning rate, starting at 0.0002 and decreasing by 1% after every epoch. All tasks are unified into a one-off, end-to-end training process, without any task-specific designs. The training takes about 6 hours for 120 epochs on 4 NVIDIA GeForce RTX 4090 GPUs.

A.2. Datasets and Metrics

H3.6M (Human3.6M) is used for the pose estimation and future pose estimation tasks. It is a large-scale dataset, which contains 3.6 million video frames of actions involving 15 types and 7 actors. Following previous works [10], we use subjects 1, 5, 6, 7, and 8 for training, and subjects 9, 11 for testing, and preprocessing the poses. After preprocessing, each pose has 17 joints. We use the Stacked Hourglass (SH) networks [5] to extract the 2D skeletons from videos as input. For pose estimation, we expect the model to estimate the corresponding 3D skeletons. For future pose estimation, we expect the model to predict and estimate at the same time the future 3D skeletons in a time range of 300ms, given the history 300ms of 2D skeletons. As we use the same dataset on two tasks, it makes multi-tasking harder as the model needs further context to correctly perceive and then accomplish the task. In Skeleton-in-Context we use prompt to guide the model to learn in context.

AMASS is used for the motion prediction task. It integrates most of the existing marker-based Mocap datasets, which are parameterized with a unified representation. We follow common practices in human motion prediction [1, 3] to use AMASS-BMLrub for testing and the rest of the datasets for training. For motion prediction, we expect the model to predict the future 3D motion sequence given the history 3D motion sequence. The time ranges of the future and history motion sequences are both 400ms, which is 10 frames under the frame rate of 25 fps. As the model requires the sequence to be of 16 frames, we pad the last poses 6 times.

3DPW (3D Pose in the Wild) is used for the joint completion task. It has more than 51k frames with 3D annotations for challenging indoor and outdoor activities. For joint completion, we construct 2 settings, where we respectively randomly mask 40% and 60% of all the joints. We expect the model to reconstruct the missing joints.

Metrics. For Motion Prediction (MP.), Joint Completion (JC.), and Future Pose Estimation (FPE.), we report Mean Per Joint Position Error (mm) [2]. For Pose Estimation (PE.), we additionally report another indicator, Normalized Mean Per Joint Position Error (N-MPJPE) [10].

Table 2. Detailed results of pose estimation between our Skeleton-in-Context and multi-task models re-structured from task-specific models or multi-stage models. We report Mean Per Joint Position Error (MPJPE).

Methods	Venues	Dire.	Disc.	Eat.	Greet	Phone	Photo	Pose	Purch.	Sit.	SitD.	Smoke	Wait	Walk	WalkD.	WalkT.	Avg
siMLPe [4]	WACV'23	56.4	65.3	69.2	68.7	71.7	98.3	57.9	66.8	91.7	129.9	68.4	72.2	61.3	77.1	63.5	74.6
EqMotion [8]	CVPR'23	141.4	151.9	175.1	154.0	165.2	173.4	131.0	194.1	201.7	219.0	159.6	151.6	129.3	160.9	141.8	163.3
STCFormer [6]	CVPR'23	46.3	52.7	51.2	48.5	53.7	70.3	48.3	48.3	72.9	100.0	54.6	50.3	38.4	56.7	41.8	55.6
GLA-GCN [9]	ICCV'23	58.1	71.4	73.0	67.7	78.9	97.3	60.9	73.2	90.1	125.4	75.0	73.3	74.0	84.3	76.3	78.6
MotionBERT [10]	ICCV'23	47.3	52.7	49.8	49.0	56.9	73.2	48.7	49.5	67.8	94.4	55.3	52.7	40.2	57.2	42.8	55.8
Static-SiC (ours)	-	**45.8**	52.9	**46.6**	**46.6**	54.6	**66.8**	47.9	47.0	**60.4**	**74.5**	**51.7**	49.9	**37.0**	**51.9**	**40.0**	**51.6**
Dynamic-SiC (ours)	-	47.0	**52.1**	48.8	47.4	**52.6**	69.1	**47.1**	**46.6**	**60.4**	**74.5**	52.3	**49.1**	37.5	52.1	40.1	51.8

Table 3. Detailed results of future pose estimation between our Skeleton-in-Context and multi-task models re-structured from task-specific models or multi-stage models. We report Mean Per Joint Position Error (MPJPE).

Methods	Venues	Dire.	Disc.	Eat.	Greet	Phone	Photo	Pose	Purch.	Sit.	SitD.	Smoke	Wait	Walk	WalkD.	WalkT.	Avg
siMLPe [4]	WACV'23	61.2	73.4	63.9	67.6	64.2	79.9	62.2	76.1	68.8	88.0	62.1	65.1	69.7	80.4	71.9	66.4
EqMotion [8]	CVPR'23	73.5	82.8	92.3	85.8	86.9	94.8	70.5	100.7	94.1	102.1	82.8	81.0	104.5	93.1	101.1	88.6
STCFormer [6]	CVPR'23	59.8	68.9	58.3	69.4	62.7	80.2	63.5	85.2	72.0	90.4	62.1	68.1	72.4	81.5	80.0	67.0
GLA-GCN [9]	ICCV'23	59.9	68.4	67.0	70.6	67.6	82.6	67.7	70.5	62.9	79.8	64.4	66.6	73.1	80.2	86.5	64.4
MotionBERT [10]	ICCV'23	63.2	74.9	66.9	105.7	70.2	93.3	94.4	93.3	67.8	92.0	68.7	89.8	83.6	129.0	78.3	86.2
Static-SiC (ours)	-	**59.3**	68.4	**55.1**	**67.5**	60.3	79.8	61.3	69.9	62.8	**78.2**	**59.3**	60.6	48.9	71.3	51.0	63.5
Dynamic-SiC (ours)	-	59.7	**67.3**	55.4	67.8	**57.8**	**79.1**	**59.4**	**68.6**	**62.6**	79.4	59.4	**60.4**	**48.4**	**70.4**	**49.9**	**62.9**

B. Multi-Task Synergistic Training

In this section, we address the challenge brought by multi-task training and demonstrate the effectiveness of our model under multi-task learning of human motion representations. First, we analyze the effect of negative transfer and how it can limit multi-task training. Next, we evaluate the effectiveness of our proposed task-guided and task-unified prompts (TGPs and TUPs) in addressing this challenge and facilitating collaborative training between multiple tasks.

B.1. Negative Transfer in Multi-Task

A straight way to train a generalist multi-task model is to collect the data from multiple tasks, format them into the same shape, and directly train the model on them. However, as explained in [7], a phenomenon named *negative transfer* in multi-dataset training will occur and lead to poor results in the above training method. As different tasks have their own unique goals, they may confuse the model during training without guidance from context, leading to performance drop in testing. To demonstrate the effect of negative transfer, as shown in Tab. 1, we train and test the backbone of our SiC separately on four tasks, whose results are in Tab. 1 with gray background. Note that the backbone here does not include TUP and TGP. When we train the four tasks together, the performance of the model decreases due to the data variability between the individual tasks, which can be attributed to the *negative transfer* phenomenon [7]. The experiments show that, without context guidance from prompts (TUP and TGP), the multi-task training is limited by negative transfer and not able to achieve satisfactory performance.

B.2. Multi-Task Synergistic Training

Our Skeleton-in-Context enables multiple tasks to be trained in a unified way, we term it Multi-Task Synergistic Training. With our proposed Task-Guided Prompt (TGP), SiC can perceive context (the corresponding task-specific information) from TGP, and accomplish the task accurately. Meanwhile, in order to deal with data from different datasets and tasks, which have different patterns and may affect the performance if not processed properly, we introduce the Task-Unified Prompt (TUP) as prior knowledge of the query target. As a task-agnostic module, the TGP is able to encode the unified human motion representations of various tasks. As Tab. 1 shows, with the proposed TGP and TUP, our SiC is able to achieve impressive results on four tasks simultaneously.

C. Detailed Results and Visualization

C.1. Detailed Results

We report the action-wise results in pose estimation and future pose estimation on H3.6M [2]. As shown in Tab. 2 and Tab. 3, our model outperforms other multi-task models in each action on H3.6M, which verifies our model's ability to handle multi-tasks. Additionally, Dynamic-SiC performs better than Static-SiC in most actions.

C.2. More Visualization

We provide more visualization results on four tasks in comparison with all the multi-task baselines that we mention in the main text, as shown in Fig. 1, 2, 3, 4. Other models are obviously unable to balance the resources of each task during training, which leads to unsatisfactory results.

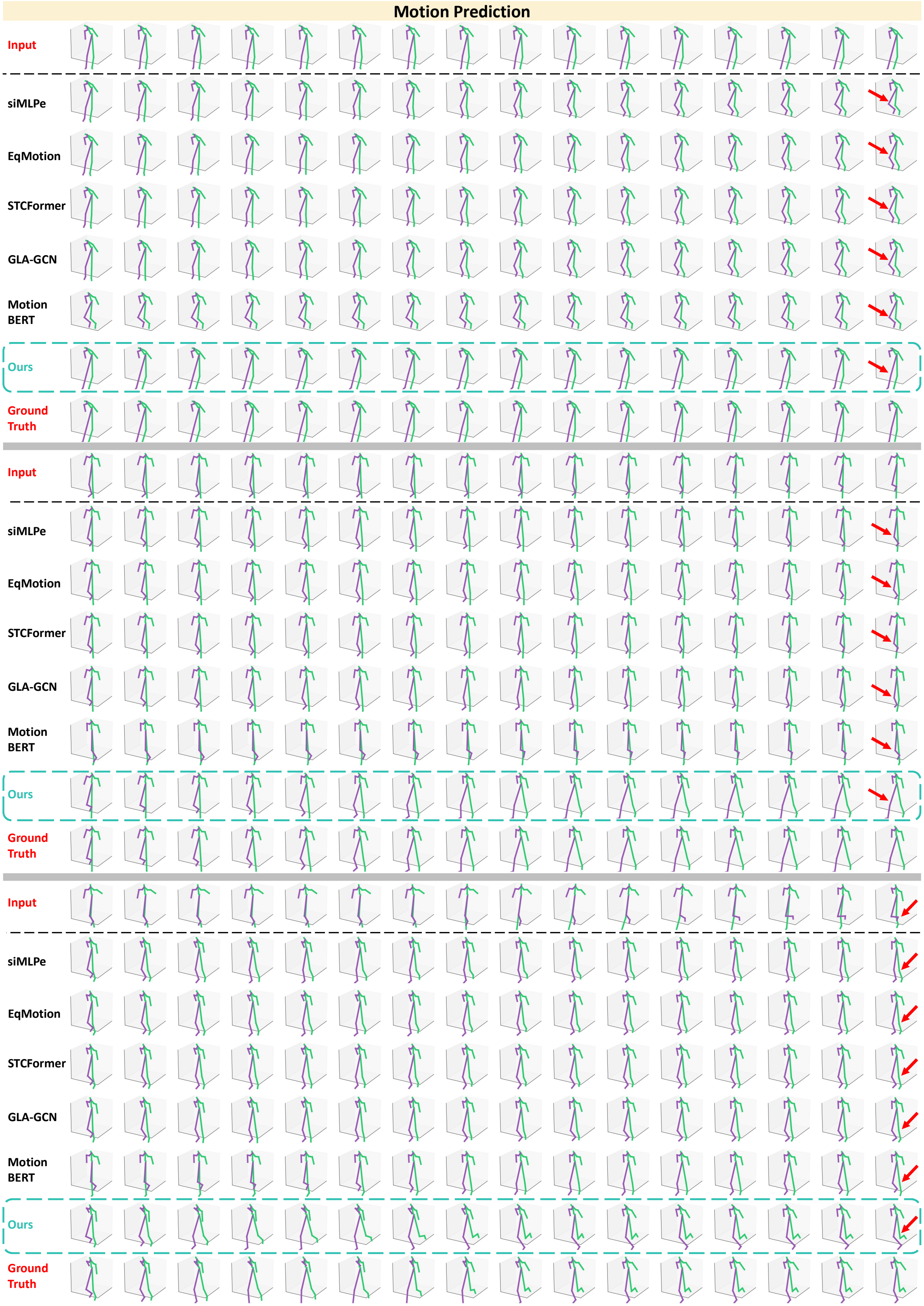

Figure 1. Comparison of visualization between multi-task models and our Skeleton-in-Context on motion prediction.

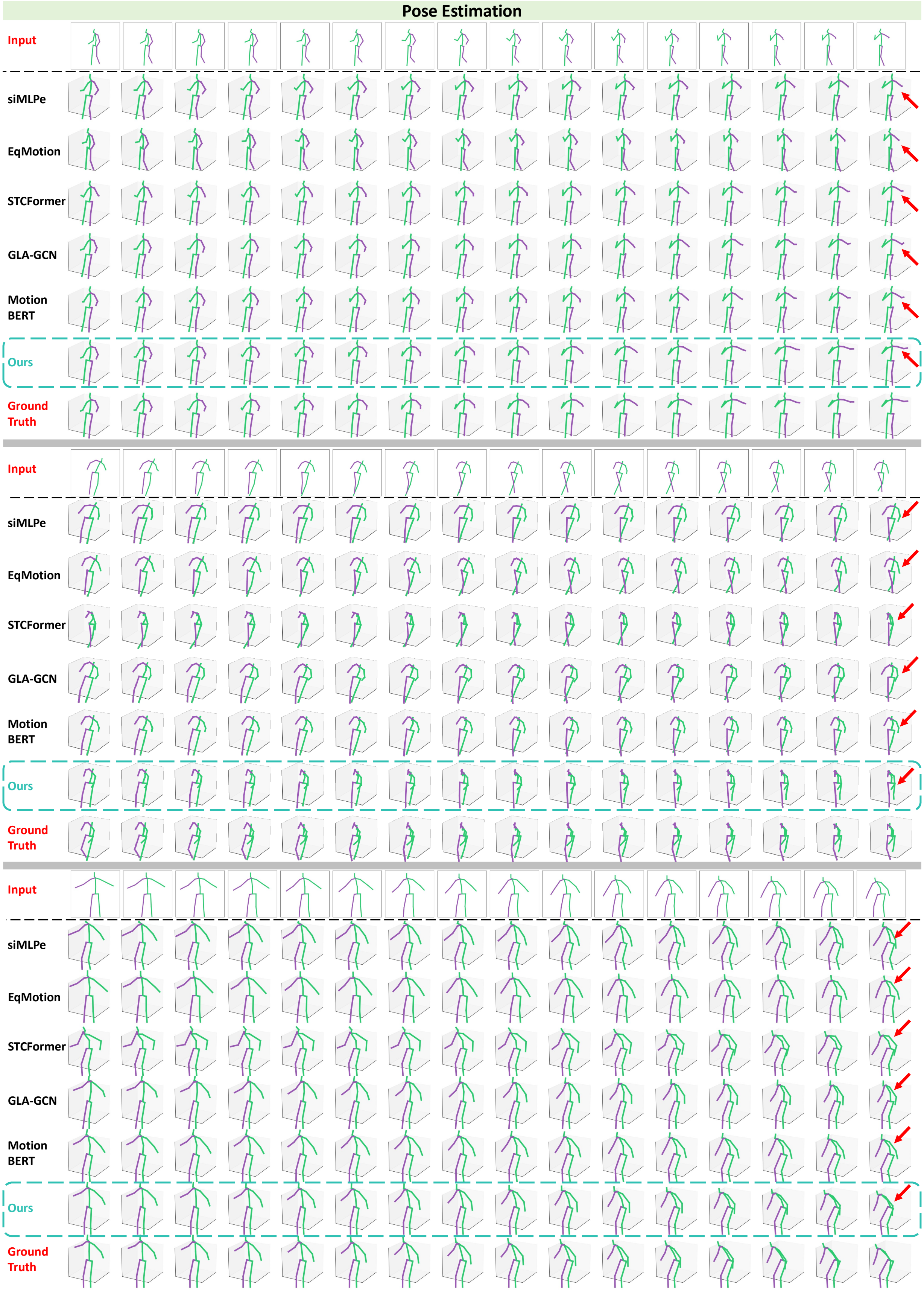

Figure 2. Comparison of visualization between multi-task models and our Skeleton-in-Context on pose estimation.

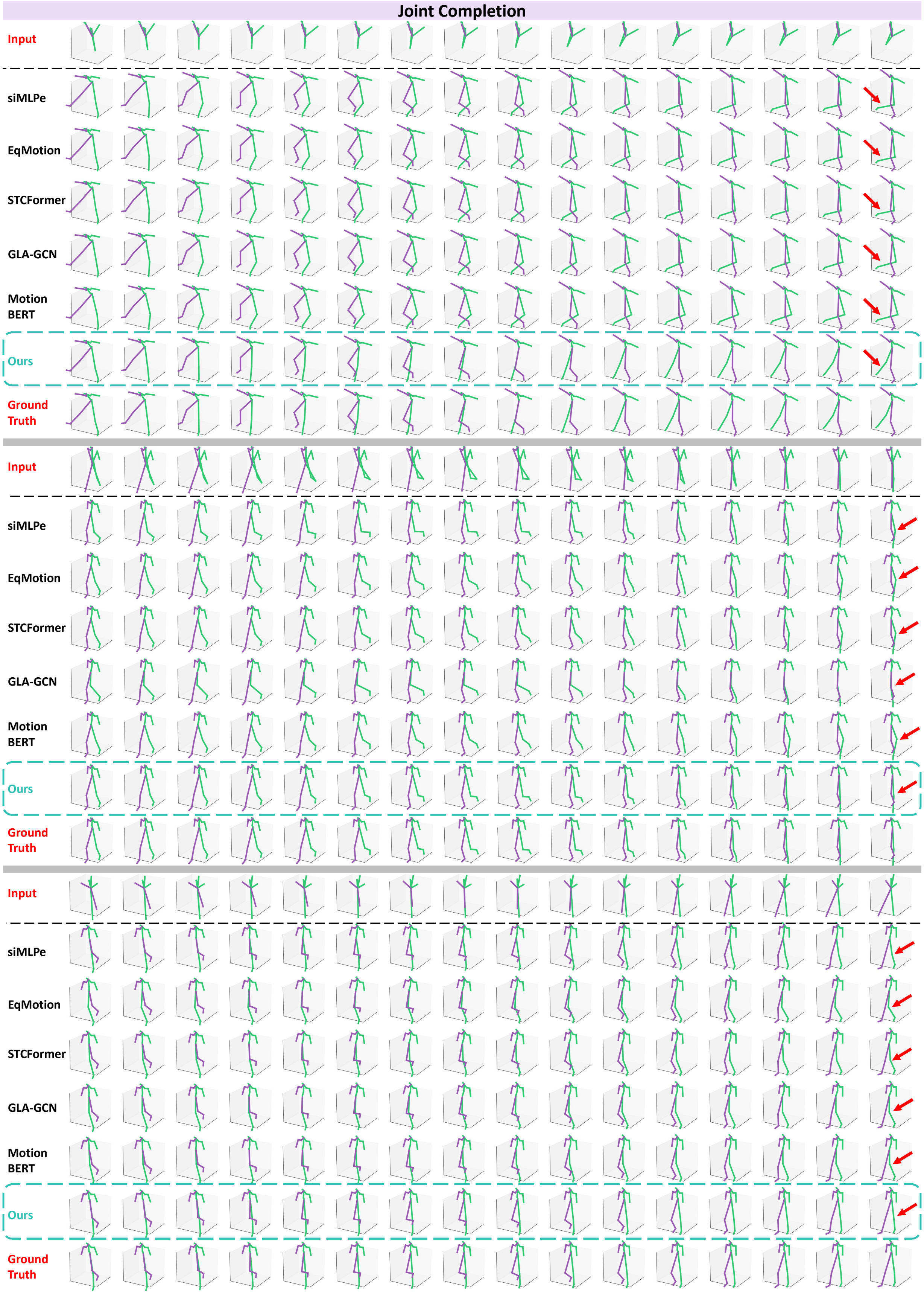

Figure 3. Comparison of visualization between multi-task models and our Skeleton-in-Context on joint completion.

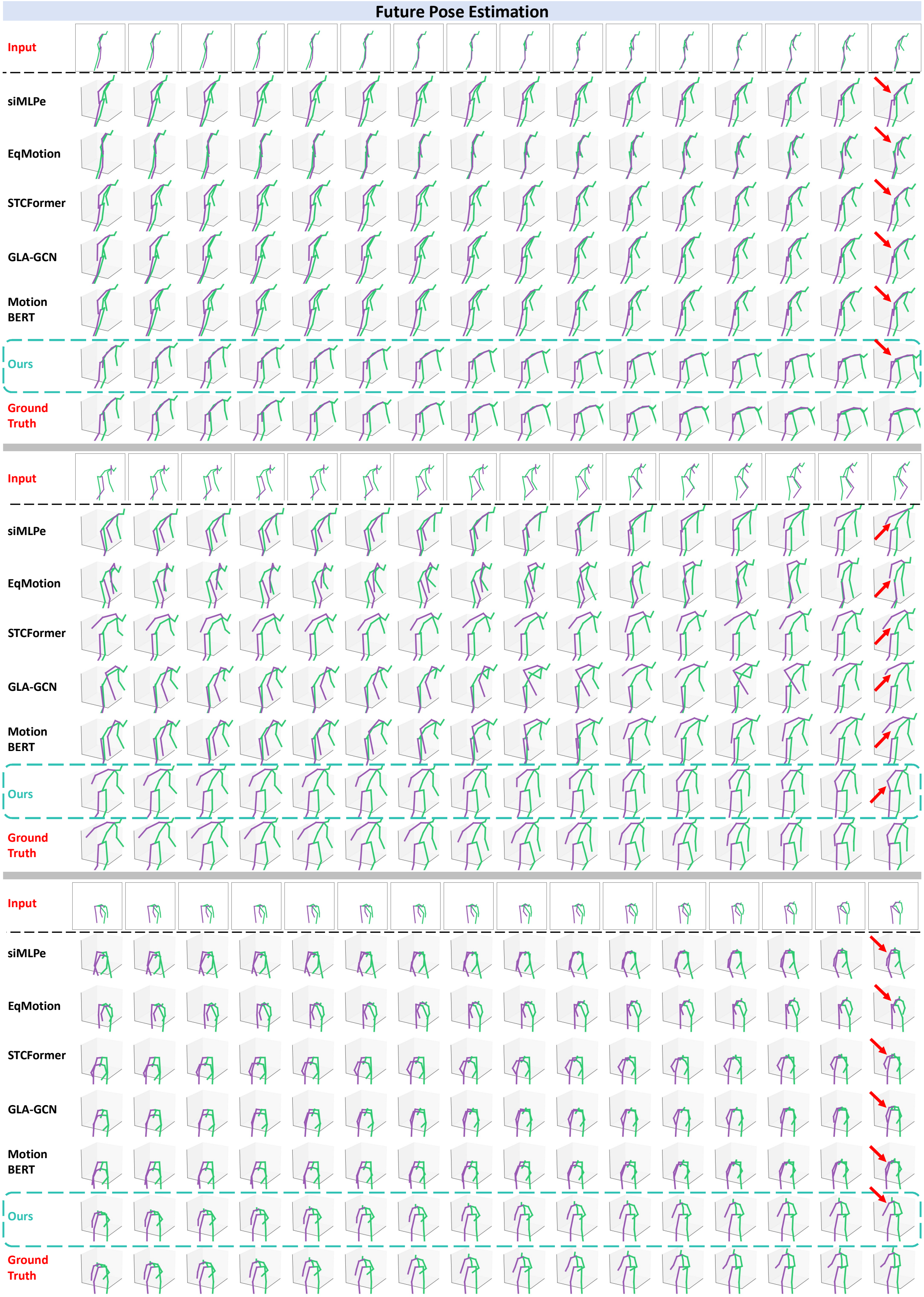

Figure 4. Comparison of visualization between multi-task models and our Skeleton-in-Context on future pose estimation.

References

[1] Wen Guo, Yuming Du, Xi Shen, Vincent Lepetit, Xavier Alameda-Pineda, and Francesc Moreno-Noguer. Back to mlp: A simple baseline for human motion prediction. In *WACV*, 2023. 1

[2] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3. 6m: Large scale datasets and predictive methods for 3d human sensing in natural environments. *PAMI*, 2013. 1, 2

[3] Wei Mao, Miaomiao Liu, and Mathieu Salzmann. History repeats itself: Human motion prediction via motion attention. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XIV 16*, pages 474–489. Springer, 2020. 1

[4] Julieta Martinez, Rayat Hossain, Javier Romero, and James J Little. A simple yet effective baseline for 3d human pose estimation. In *ICCV*, 2017. 2

[5] Alejandro Newell, Kaiyu Yang, and Jia Deng. Stacked hourglass networks for human pose estimation. In *ECCV*, 2016. 1

[6] Zhenhua Tang, Zhaofan Qiu, Yanbin Hao, Richang Hong, and Ting Yao. 3d human pose estimation with spatio-temporal criss-cross attention. In *CVPR*, 2023. 2

[7] Xiaoyang Wu, Zhuotao Tian, Xin Wen, Bohao Peng, Xihui Liu, Kaicheng Yu, and Hengshuang Zhao. Towards large-scale 3d representation learning with multi-dataset point prompt training. *arXiv preprint arXiv:2308.09718*, 2023. 2

[8] Chenxin Xu, Robby T Tan, Yuhong Tan, Siheng Chen, Yu Guang Wang, Xinchao Wang, and Yanfeng Wang. Eqmotion: Equivariant multi-agent motion prediction with invariant interaction reasoning. In *CVPR*, 2023. 2

[9] Bruce XB Yu, Zhi Zhang, Yongxu Liu, Sheng-hua Zhong, Yan Liu, and Chang Wen Chen. Gla-gcn: Global-local adaptive graph convolutional network for 3d human pose estimation from monocular video. In *ICCV*, 2023. 2

[10] Wentao Zhu, Xiaoxuan Ma, Zhaoyang Liu, Libin Liu, Wayne Wu, and Yizhou Wang. Motionbert: A unified perspective on learning human motion representations. In *ICCV*, 2023. 1, 2