Topic L - Action-conditioned 3D human motion synthesis

January 17, 2022

Siwar Mhadhbi
Télécom Paris
siwar.mhadhbi@telecom-paris.fr

Eya Ghamgui
Télécom Paris
eya.ghamgui@telecom-paris.fr

Abstract

Synthesizing realistic and controllable human motion sequences is essential to facilitate human-centric video generation, virtual reality, and character control. Nevertheless, human motion synthesis remains a challenging task despite decades of research. In this context, many deep neural networks were developed to solve this issue. The goal of our project is to tackle this pro blem and try to generate more realistic and diverse 3D human motion sequences conditioned on actions with ACTOR [8] by employing PARE [4], an improved version of the VIBE model [3]. Furthermore, we discuss the weights of the construction losses in order to improve the model. In last part, we evaluate our model on a real-world dataset we chose, "Youtube Action Data Set"[5].

1. ACTOR

1.1. Architecture of the model

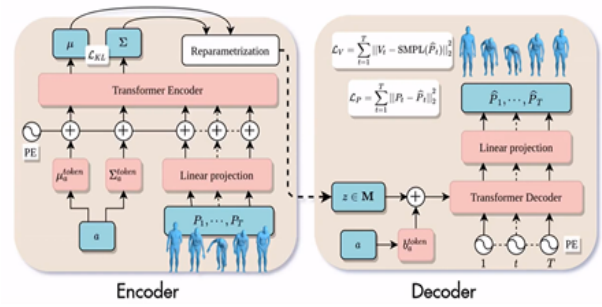


Figure 1: Architecture of ACTOR [8].

The model is an action-conditioned transformer VAE. It consists of two parts: the encoder and the decoder, each consisting of Transformer layers. The encoder takes as input a sequence of poses of arbitrary length and the corresponding action label, and produces distribution parameters μ and Σ of the latent motion space. Using the VAE reparametrization trick, a latent vector z is sampled from this distribution. The decoder takes a single latent vector z and an action label a, and generates a human pose for a given duration. A differentiable SMPL [7] layer is then used to obtain the vertices and joints based on the pose parameters provided by the decoder.

Several losses are used to train the model. The Kullback-Leibler loss ($\mathcal{L}_{KL}$) is the standard loss term to regularize the latent space. The reconstruction loss is composed of two terms, both using the SMPL body model: one L_2 loss defined on the SMPL poses represented as rotations ($\mathcal{L}_P$), another L_2 loss defined on the vertex coordinates obtained by the differentiable SMPL layer ($\mathcal{L}_V$). The resulting total loss is defined as the sum of different terms:

$$\mathcal{L} = \mathcal{L}_P + \mathcal{L}_V + \lambda_{KL}\,\mathcal{L}_{KL} \;\; ; \;\; \text{where} \;\; \lambda_{KL} = 10^{-5}$$

1.2. Datasets: SMPL pose estimates

Our model architecture takes as input SMPL pose estimates [7]. This Simple Body Model goes beyond sparse joints and outputs the body surface, which paves the way for better modeling interaction with the environment. In the first part of our project, we work on three different action recognition datasets: HumanAct12 [1], UESTC [2] and NTU13 [6, 9]. We are provided with SMPL pose estimates for HumanAct12 and UESTC but not for NTU13. Hence, our first step is to generate the SMPL estimates for the NTU13 dataset.

1.2.1 SMPL pose estimation of NTU13 dataset

We apply the VIBE method [3] which is a monocular human motion estimation that produces realistic and kinematically plausible 3D body poses from a given RGB video. The results of VIBE on NTU13 are shown in Appendix B.

Three detection cases arise: one person detection, no detection and multi-person detection. Thus, a post-processing step is needed. In case of no detection, we discard the corresponding samples. In case of multi-person detection, we process each sample independently and select the person of interest by visualizing the VIBE output superposed on the initial video. Doing so, two cases draw our attention. First case, same person is detected twice as shown in Figure 5, we select the first detection, but one might wonder if such

	NTU13			
Method	FID$_{tr}$ ↓	Acc.↑	Div. →	Multimod.→
Real	$0.02^{\pm 0.00}$	$99.8^{\pm 0.0}$	$7.07^{\pm 0.02}$	$2.25^{\pm 0.01}$
Real*	$0.02^{\pm 0.00}$	$98.8^{\pm 0.02}$	$7.10^{\pm 0.03}$	$2.36^{\pm 0.01}$
ACTOR	$0.11^{\pm 0.00}$	$97.1^{\pm 0.2}$	$7.08^{\pm 0.04}$	$2.08^{\pm 0.01}$
ACTOR*	$\mathbf{0.10^{\pm 0.00}}$	$\mathbf{97.3^{\pm 0.18}}$	$\mathbf{7.08^{\pm 0.03}}$	$\mathbf{2.10^{\pm 0.01}}$

Table 1: Comparison of quantitative results between the paper and our model* (wVIBE) on NTU13 dataset.

action can rise a confusion between "pick up" and "touch toe" actions. Second case, failure detection as the chair is detected a sitting person as shown in Figure 3c.

2. Training & Experimental results

2.1. Training parameters

During training, we use the AdamW optimizer with a fixed learning rate of 10^{-4}. The size of the mini-batches is set to 20 and the number of epochs is equal to 2000, 5000 and 1000 for NTU13, HumanAct12 and UESTC datasets, respectively.

2.2. ACTOR wVIBE results

2.2.1 Performance metrics [1]

- Frechet Inception Distance (FID): is an important and widely used metric to evaluate the overall quality of the generated motions.

$$FID = ||\mu - \mu_w||_2^2 + tr(\Sigma + \Sigma_w - 2(\Sigma^{\frac{1}{2}}\Sigma_w\Sigma^{\frac{1}{2}})^{\frac{1}{2}})$$

- Accuracy: indicates the correlation of the motion and its action type.

- Diversity: measures the variance of the generated motions across all action categories.

$$Diversity = \frac{1}{S_d}\sum_{i=1}^{S_d}||v_i - v_i^{'}||_2$$

where v and $v^{'}$ are motion feature vectors of the same size.

- Multimodality: measures how much the generated motions diversify within each action type.

$$Multimodality = \frac{1}{C \times S_l}\sum_{c=1}^{C}\sum_{i=1}^{S_l}||v_{c,i} - v_{c,i}^{'}||_2$$

2.2.2 Quantitative results

By training the ACTOR model on the three different datasets with the same hyperparameters as in [8] and using the initial resolution of the RGB videos, we obtain performances very close to those found by the authors (cf. Tables 1,3,4). For NTU13 and UESTC, we notice a slight improvement for all metrics. For HumanAct12, only diversity and multimodality are slightly better.

2.2.3 Qualitative results on NTU13

We successfully generate the motions of all the actions learned during training. We observe a great diversity in the way a given action is executed. For example, the action "side-kick" is performed with the left or right leg (cf. Figure 6). Furthermore, the model retains the essence of the action's semantics while changing the nuances (angles, speed, phase) or body parts movements. For example, the model changes the position of hands in the "sit" action (cf. Figure 7), and modifies the way of "throwing" (cf. Figure 8).

3. Ideas for improvement

3.1. ACTOR wPARE results

PARE model [4] was introduced as an improvement of the VIBE model in estimating the body poses from a given video. This was our motivation to use PARE estimates instead of VIBE estimates to train a new ACTOR model aiming to produce better motion generation results.

3.1.1 SMPL pose estimation with PARE on NTU13

In this part, we'll focus on NTU13 dataset. The PARE estimates are presented in Appendix C. We notice that the PARE model detects all persons of interest plus some persons from background in particular cases. This is an improvement over VIBE as we do not discard any samples with no detection at all. Nonetheless, few cases with erroneous detection such as "person twice detected" and "chair detected as sitting person" remain the same with PARE model. Thus, a post-processing step is needed as well.

3.1.2 Quantitative results on NTU13

We compute the same evaluation metrics for both ground truth real data (Real⋆) and generated data (ACTOR⋆) in Table 2. We notice that the results of the generated sequences are close to the real ones. In our evaluation, we

	NTU13			
Method	FID$_{tr}$ ↓	Acc.↑	Div. →	Multimod.→
Real*	$0.02^{\pm 0.00}$	$93.1^{\pm 0.16}$	$6.99^{\pm 0.01}$	$2.88^{\pm 0.01}$
ACTOR*	$0.17^{\pm 0.00}$	$92.7^{\pm 0.28}$	$7.02^{\pm 0.01}$	$2.45^{\pm 0.01}$

Table 2: Quantitative results of ACTOR wPARE on NTU13 dataset.

also use the average shape parameter ($\beta = \vec{0}$) when obtaining joint coordinates from the mesh for both real and generated sequences. Comparing these results to those when using VIBE estimates, they are not better; we obtain higher FID and lower Accuracy. Nonetheless, they are not trivial to make comparisons and understand results. Instead, we interpret the qualitative results and compare them with ACTOR wVIBE in next section.

3.1.3 Qualitative results on NTU13

The qualitative results of ACTOR with PARE estimates are represented in Appendix E. Our generated motion sequences with PARE appear to be more realistic while still diverse as they present different angles, side-views, body parts, as shown with VIBE results in section 2.2.3. We show in Figure 9 three examples of action-conditioned 3D human motion sequences: “sit”, “run” and “kick”. Concerning the first action, the motion with VIBE is barely ”sitting”, the generation appears rather as a standing human shape. As for the second action, the shape generated with VIBE barely moves its arms and legs compared to PARE generation. In the third action, clearly the ”kicking” appears way more realistic with PARE than with VIBE.

3.2. Tuning the Loss Weighting Parameters

We retrain the ACTOR model with different combinations of weights for the two construction losses searching for the best combination that could improve the performance of the model. The comparison analysis between all metrics is shown in Figure 2. We can say that by increasing both construction losses equally to 2, we are able to improve the accuracy and diversity metrics. In fact, the accuracy improved by 0.6% and the diversity by 0.01. By giving higher weight to $\mathcal{L}_P$, we obtain better performance for the multimodality metric. However, the FID metric is still better for the original model.

4. Performance on a real-world dataset

In this section, we train the model on a real-world dataset, ”Youtube Action Data Set” [5], which contains non-frontal video sequences more in the wild. We consider five actions (”tennis swing”, ”walking dog”, ”playing basketball”, ”golf swing”, and ”soccer juggling”) to train the model, evaluate it and test whether it is able to generalize to these videos or not. Each action is composed of 150 videos with a maximum duration of 10 seconds. We then generate PARE estimates from these videos, train the ACTOR model on the obtained dataset with centered joints for 5000 epochs, and finally generate some sequences per action as shown in Figure 10. We found that the quality of the generated sequences is poor for the actions ”walking dog” and ”juggling soccer”. The other motions are coherent with their corresponding action. In fact, for ”golf swing”, ”tennis swing” and ”playing basketball”, focusing on hands, the generated sequences can express well the motion. In the future, we can modify the hyperparameters, increase the size of the dataset, or apply domain adaptation approach in order to improve the performance of the ACTOR model on this dataset.

Conclusion & Future Perspectives

In this project, we started by building the VIBE estimates of the NTU13 dataset. Then, we retrained the ACTOR model on the three datasets built from the VIBE model on NTU13, UESTC and HumanAct12. We managed to obtain performances close to those of the paper. As improvement, we applied the PARE model on NTU13 and retrained the ACTOR model. We managed to obtain better qualitative results in the sense of more realistic 3D human motions.

Furthermore, to improve the model trained on the VIBE dataset, we proposed to modify the weights of the construction losses. By increasing their values, we obtained better accuracy. Moreover, we evaluated the model on a real-world dataset extracted from ”YouTube Action Data Set”.

A limitation of the ACTOR model is that the maximum duration it can generate is dependent on computational resources since it produces the entire sequence in one shot.

In the future, we can leverage this model to impose priors on motion estimation or action recognition problems. Another future work will be to explore open vocabulary actions. In addition, future work can exploit a regression block on sequence duration so that the generator is able to provide an accurate motion sequence without needing to put duration as input.

References

[1] 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 *Proceedings of the 28th ACM International Conference on Multimedia*, pages 2021–2029, 2020. 1, 2

[2] Yanli Ji, Feixiang Xu, Yang Yang, Fumin Shen, Heng Tao Shen, and Wei-Shi Zheng. A large-scale rgb-d database for arbitrary-view human action recognition. In *Proceedings of the 26th ACM international Conference on Multimedia*, pages 1510–1518, 2018. 1

[3] Muhammed Kocabas, Nikos Athanasiou, and Michael J. Black. Vibe: Video inference for human body pose and shape estimation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2020. 1

[4] Muhammed Kocabas, Chun-Hao P. Huang, Otmar Hilliges, and Michael J. Black. PARE: Part attention regressor for 3D human body estimation. In *Proc. International Conference on Computer Vision (ICCV)*, pages 11127–11137, Oct. 2021. 1, 2

[5] Jingen Liu, Jiebo Luo, and Mubarak Shah. Recognizing realistic actions from videos "in the wild". In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1996–2003. IEEE, 2009. 1, 3

[6] Jun Liu, Amir Shahroudy, Mauricio Perez, Gang Wang, Ling-Yu Duan, and Alex C Kot. Ntu rgb+ d 120: A large-scale benchmark for 3d human activity understanding. *IEEE transactions on pattern analysis and machine intelligence*, 42(10):2684–2701, 2019. 1

[7] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J Black. Smpl: A skinned multi-person linear model. *ACM transactions on graphics (TOG)*, 34(6):1–16, 2015. 1

[8] Mathis Petrovich, Michael J. Black, and Gül Varol. Action-conditioned 3D human motion synthesis with transformer VAE. In *International Conference on Computer Vision (ICCV)*, 2021. 1, 2

[9] Amir Shahroudy, Jun Liu, Tian-Tsong Ng, and Gang Wang. Ntu rgb+ d: A large scale dataset for 3d human activity analysis. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1010–1019, 2016. 1

Appendix

A. Additional quantitative results

	HumanAct12			
Method	FID$_{tr}$ ↓	Acc.↑	Div. →	Multimod.→
Real	$0.02^{\pm 0.00}$	$99.4^{\pm 0.0}$	$6.86^{\pm 0.03}$	$2.60^{\pm 0.01}$
Real*	$0.02^{\pm 0.00}$	$99.4^{\pm 0.03}$	$6.93^{\pm 0.02}$	$2.58^{\pm 0.02}$
ACTOR	$\mathbf{0.12^{\pm 0.00}}$	$\mathbf{95.5^{\pm 0.8}}$	$6.84^{\pm 0.03}$	$2.53^{\pm 0.02}$
ACTOR*	$0.13^{\pm 0.00}$	$95.2^{\pm 0.72}$	$\mathbf{6.85^{\pm 0.02}}$	$\mathbf{2.61^{\pm 0.01}}$

Table 3: Comparison of quantitative results between the paper and our model* (wVIBE) on HumanAct12 dataset.↑

	UESTC				
Method	FID$_{tr}$ ↓	FID$_{test}$ ↓	Acc.↑	Div. →	Multimod.→
Real	$2.93^{\pm 0.26}$	$2.79^{\pm 0.29}$	$98.8^{\pm 0.1}$	$33.34^{\pm 0.32}$	$14.16^{\pm 0.16}$
ACTOR	$20.49^{\pm 2.31}$	$23.43^{\pm 2.20}$	$91.1^{\pm 0.3}$	$31.96^{\pm 0.36}$	$14.66^{\pm 0.03}$
ACTOR*	$\mathbf{17.31^{\pm 0.74}}$	$\mathbf{19.62^{\pm 1.61}}$	$\mathbf{92.6^{\pm 0.40}}$	$\mathbf{32.68^{\pm 0.32}}$	$\mathbf{14.3^{\pm 0.12}}$

Table 4: Comparison of quantitative results between the paper and our model* (wVIBE) on UESTC dataset.↑

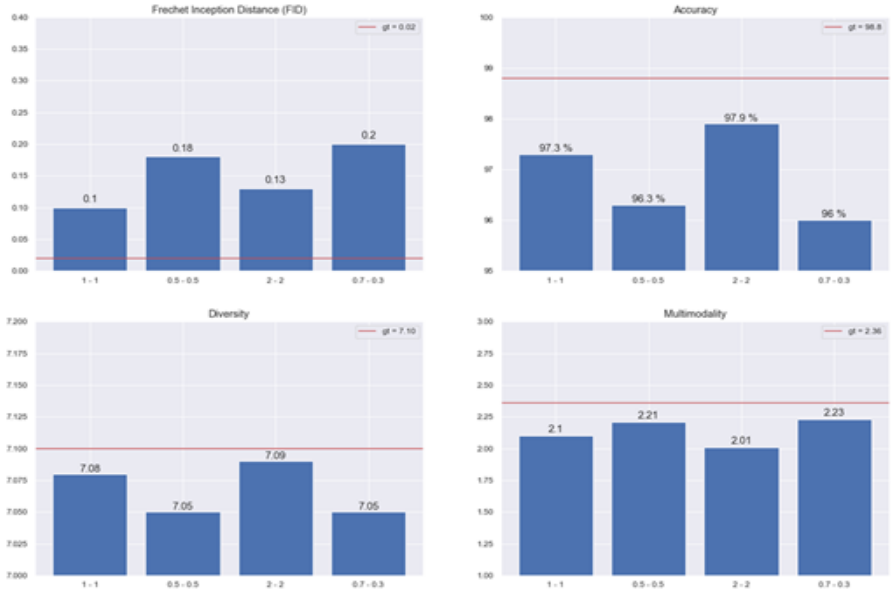
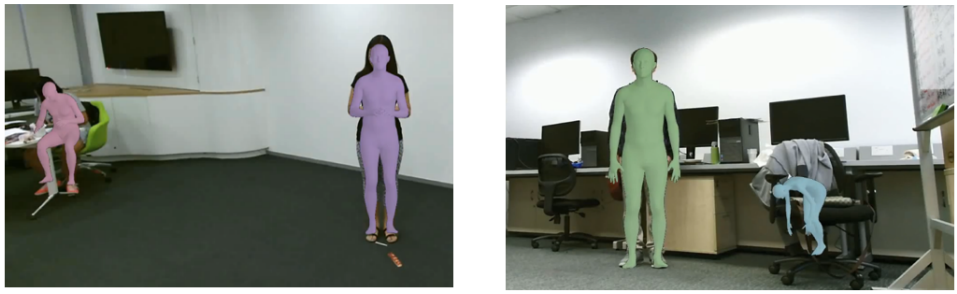


Figure 2: Metrics values while changing weights of the construction losses: λ_P - λ_V.↑

B. VIBE estimates on NTU13

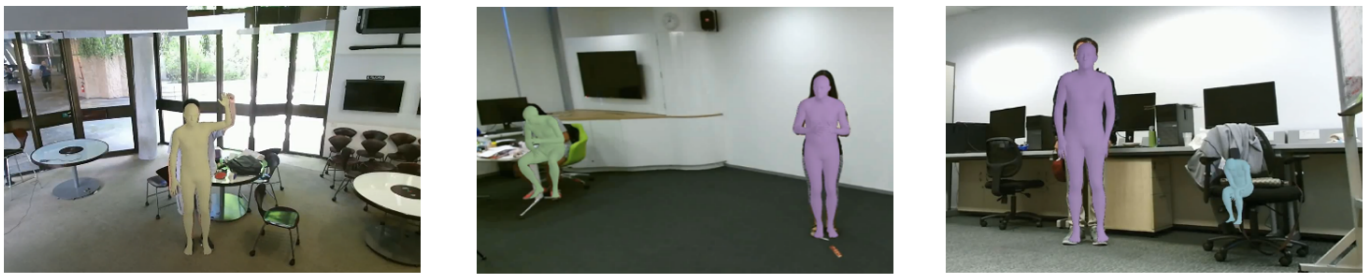
a) No detection b) One person detected



c) Multi-person detection d) Failure detection

Figure 3: Examples of VIBE estimates on NTU13 dataset.↑

C. PARE estimates on NTU13



a) One person detection b) Multi-person detection c) Failure detection

Figure 4: Examples of PARE estimates on NTU13 dataset.↑

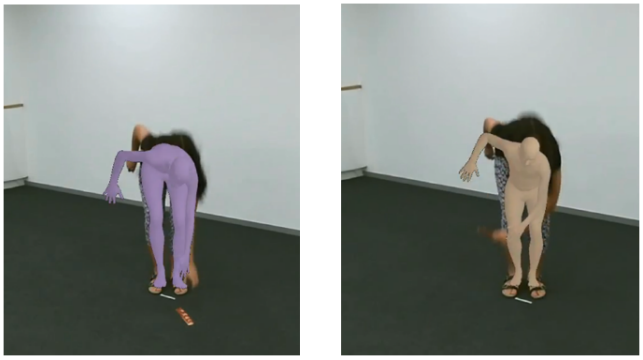


Figure 5: ”pick up” - Person detected twice.
Left: one person detected while going down before picking.
Right: one person detected while going up after picking.↑

D. Qualitative results wVIBE on NTU13

Figure 6: ”Side-kick” action.↑

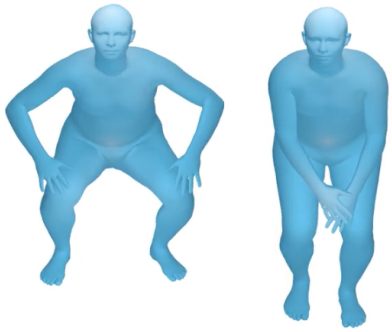


Figure 7: ”Sit” action.↑

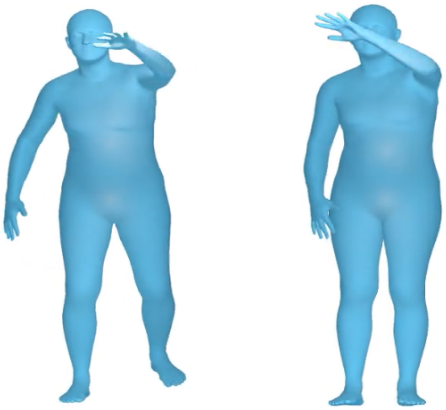


Figure 8: ”Throw” action.↑

E. Qualitative results wPARE on NTU13

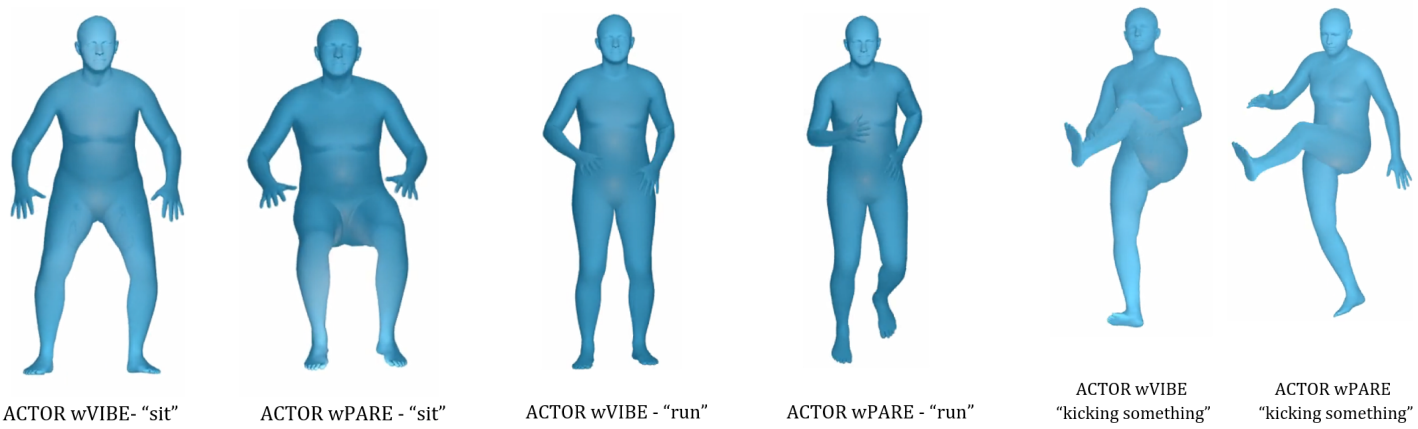


Figure 9: Comparison of ACTOR wVIBE / wPARE: examples of qualitative results.↑

F. Qualitative results wPARE on the real-world dataset

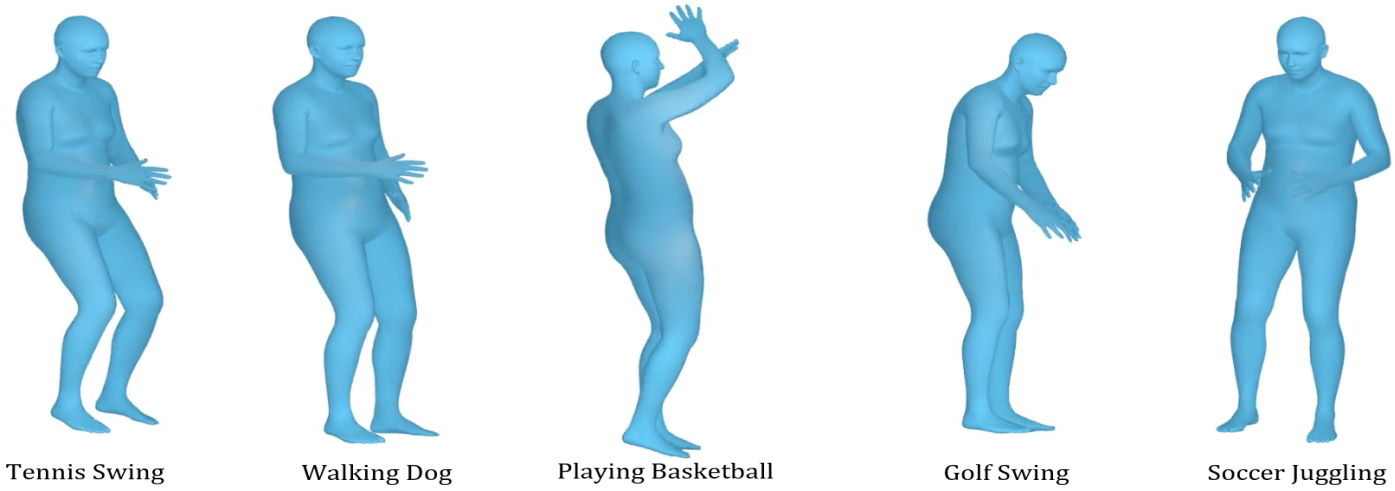


Figure 10: Motions generated from a real-world dataset.↑