2023 2nd International Conference on Artificial Intelligence, Human-Computer Interaction and Robotics (AIHCIR) | 979-8-3503-6036-3/23/$31.00 ©2023 IEEE | DOI: 10.1109/AIHCIR61661.2023.00028

TDT: Two-stream decoupled Transformer for text-driven human animation generation

1st JiaShuang Zhou
School of Computer Science and Artificial Intelligence
Wuhan Textile University
Wuhan, Hubei, China
2115363027@mail.wtu.edu.cn

2nd Xiaoqin Du*
School of Computer Science and Artificial Intelligence
Wuhan Textile University
Wuhan, Hubei, China
xiaoqindu@wtu.edu.cn

3rd Yifan Lv
School of Computer Science and Artificial Intelligence
Wuhan Textile University
Wuhan, Hubei, China
2115363017@mail.wtu.edu.cn

4th Yongqi Liu
School of Computer Science and Artificial Intelligence
Wuhan Textile University
Wuhan, Hubei, China
2115363016@mail.wtu.edu.cn

Abstract—**The automatic generation of high-quality human animations in a controllable and natural manner has always been a goal pursued by experts in the field of animation. Text-driven human animation generation offers user-friendly and versatile application scenarios. In this paper, we propose a two-stream decoupled Transformer network, TDT, for extracting motion semantic information from the text modality and generating human animations based on it. Our contributions are: (1) The text is highly decoupled in its description of trajectories and motions, utilizing a two-stream Transformer network to extract trajectory features and motion features separately. (2) We employ vector orthogonalization to constrain the motion feature vector and the trajectory feature vector, promoting the decoupling of the motion manifold and the trajectory manifold. (3) We introduce a local loss function, the mean-variance reconstruction loss, as a complement to the global loss function, further facilitating feature fusion between the two modalities. Experimental results demonstrate that our proposed model outperforms state-of-the-art text-driven human animation generation models in terms of objective evaluation metrics. Moreover, according to visualization results, the generated motions exhibit greater consistency with the semantics described in the input text by the user and are closer to the ground truth data.**

Keywords-Motion synthesis; Multimodal; Expert mixture network; Orthogonal loss; Joint embedding space

I. Introduction

The technique of automatic generation of realistic and diverse human animation has been an important area of study in the field of animation. Over the past 20 years, experts in this field have devoted a significant amount of time and effort to research modeling human motions. However, synthesizing realistic and controllable motion sequences remains an extremely challenging task.

Motion generation methods can be broadly categorized into two types: (1) unconstrained generation, which models the entire space of possible motions, and (2) conditioned generation, which aims to achieve controllability by using speech, music, images, or text (natural language) as conditioning inputs. Cudeiro[1] used speech as input to create facial animation. Guofei Sun[2] used music as input to generate human dance animation, while Kanji[3] used photographs to create 2D character animation. In this work, we specifically focus on the latter, using text as input. This choice is motivated by the fact that text input contains rich semantics and can more precisely describe our desired motion.

In this paper, our goal is to decouple the feature (latent) space to enhance the controllability of generated motion without deteriorating the information gap caused by ELBO decomposition. To achieve this, we propose using a latent space orthogonal loss, which enforces orthogonality between the trajectory feature vector and the motion feature vector. This approach helps decouple the latent space. Additionally, utilizing the latent space orthogonal loss increases inter-class distance and reduces intra-class distance to some extent, resulting in a clustering effect among samples in the dataset. This advantage is beneficial for constructing a joint embedding space for multimodal features.

There are various methods for multimodal feature fusion, such as feature stitching[7], [8], vector multiplication [9], [10], multi-modality cross-modal attention [11], [12], and pre-trained models[13], [14] among others. However, only a joint embedding space allows for the generation of data by inputting unimodal data, which corresponds to input text to generate human motion with consistent information in our experiment. A crucial aspect in constructing the joint embedding space is how to measure the distance between multimodal sample pairs.

There are two types of animation generation methods: autoregressive and non-autoregressive. The autoregressive technique involves generating subsequent animation frames based on the previous frame, allowing for the incorporation of relevant information for prediction. However, it requires data from the preceding frame as input, which can lead to issues such as slow decoding and error accumulation. Autoregressive approaches have been employed by Ghosh[4], Henter[15], and Ahn[16] for motion sequence generation.

On the other hand, the non-autoregressive technique directly generates complete and continuous motions based on semantic information. It offers faster decoding times and produces motions with better continuity and consistency compared to the autoregressive model. In the methodology proposed by Petrovich[6] and Temos[18], the Transformer model is employed as a general module for generating non-autoregressive motions. We also utilize the Transformer model in our approach to generate non-autoregressive motions.

979-8-3503-6036-3/23/$31.00 ©2023 IEEE
DOI 10.1109/AIHCIR61661.2023.00028

In this study, we propose a two-stream decoupled Transformer network (TDT) for generating text-driven human animation. Our contributions are as follows:

- The description of trajectories and motions in the text is essentially decoupled, which leads us to consider using asymmetric coding to encode trajectories and motions. Specifically, we utilize a two-stream Transformer network to extract motion features and trajectory features separately.
- We introduce an orthogonal loss that enforces orthogonality between the motion feature vector and the trajectory feature vector. This loss promotes the decoupling between the motion latent space and the trajectory latent space.
- We present a local mean-variance reconstruction loss to address the limitation of the original KL divergence, which only assesses the global loss between two distributions but not the local loss between text-motion pairs. The local loss also helps align the motion embedding with the text embedding.

Through both quantitative and qualitative experimental comparisons, our model outperforms the baseline model proposed by Ghosh[4]. Please refer to TABLE I. and Figure 3. for detailed comparisons.

II. Our Method

A. Network structure

An end-to-end two-stream decoupled Transformer network is trained for text-driven human skeleton animation synthesis. The network consists of four components: motion encoder, text encoder, motion decoder, and discriminator, as shown in Figure 1. The model learns the joint embedding encoding of text and motion and generates human skeleton animation that corresponds to the semantics of the input text.

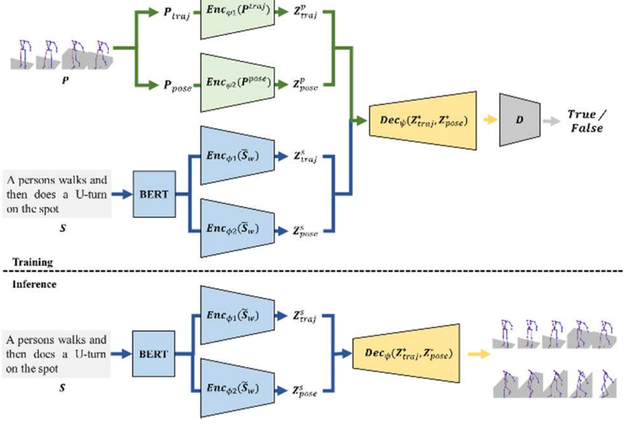

P_{traj}
$Enc_{\varphi1}(P^{traj})$
Z^p_{traj}
P
P_{pose}
$Enc_{\varphi2}(P^{pose})$
Z^p_{pose}
$Dec_\psi(Z^*_{traj}, Z^*_{pose})$
D
True / False
$Enc_{\phi1}(\tilde{S}_w)$
Z^s_{traj}
A persons walks and then does a U-turn on the spot
BERT
S
$Enc_{\phi2}(\tilde{S}_w)$
Z^s_{pose}
Training
Inference
A persons walks and then does a U-turn on the spot
BERT
S
$Enc_{\phi1}(\tilde{S}_w)$
Z^s_{traj}
$Enc_{\phi2}(\tilde{S}_w)$
Z^s_{pose}
$Dec_\psi(Z^*_{traj}, Z^*_{pose})$

Figure 1. The network is made up of four components: a motion encoder, a text encoder, a motion decoder, and a discriminator.

The motion data $P = [P_0, \dots, P_{T-1}]$ is a sequence that includes T poses. Each pose $P_t \in \mathbb{R}^{J\times3}$ at time step t consists of J nodes and each joint point includes (x, y, z) coordinates. The first node represents the root node, and the set of root nodes represents a trajectory. In this paper, the raw data are manually divided into motion data P_t^{pose} and trajectory data P_t^{traj}.

$$P_t^{pose} = P_{\{t\}[1:,:]} \in \mathbb{R}^{(J-1)\times3}$$
$$P_t^{traj} \in \mathbb{R}^6 \tag{1}$$

In the above equation, $P_t[1:,:]$ represents the 3D coordinates (x, y, z) of the other joints excluding the root node, indicating that the coordinates of the root node are not included. Translation: P_t^{traj} selects the 3D coordinates and 3 rotation information of the root node at time t as the trajectory.

After separating the trajectory data from the motion data, the model passes them through the trajectory feature extraction network and the motion feature extraction network, respectively, to obtain the trajectory features Z^p_{traj} and the motion features Z^p_{motion}:

$$Z^p_{traj} = Enc_{\varphi1}(P^{traj}) \tag{2}$$
$$Z^p_{pose} = Enc_{\varphi2}(P^{\{pose\}})$$

where $\varphi1$ represents the parameters of the trajectory encoder, and $\varphi2$ represents the parameters of the motion encoder.

The text $S = [S_1, \dots, S_W]$ represents a sequence of W words, which can be encoded by a pre-trained model to obtain the text embedding vector $\tilde{S}$. The $\tilde{S}_w \in R^K$ denotes the embedding vector obtained after embedding the w-th word.

The text embedding vector is then mapped to the text latent space, resulting in the following two feature vectors:

$$Z^s_{traj} = Enc_{\phi1(\tilde{S})} \tag{3}$$
$$Z^s_{pose} = Enc_{\phi2(\tilde{S})}$$

where $Z^s_{traj}, Z^s_{pose} \in \mathbb{R}^{\mathrm{h}}$ (in this paper, we set $h = 256$), represent the trajectory encoder and motion encoder features extracted from the text, respectively. $\phi1$ and $\phi2$ denote the parameters of the trajectory encoder and motion encoder, respectively. To encourage these two manifolds to carry similar information and converge closely in the latent space, we apply suitable loss functions to constrain them.

During the decoding phase, the motion decoder samples from the motion and text manifolds to generate the skeleton animation for each time step T in a single pass, using the Transformer as the generator in a non-autoregressive manner. Unlike the autoregressive generation approach, this method does not require the initial state or the previous frame as input. It effectively avoids the error accumulation issue associated with autoregressive models, resulting in generated motions that are more consistent with the semantic description provided by the input text. The formulation is as follows:

$$\hat{P}^p = Dec_\psi\left(Z^p_{pose}, Z^p_{traj}\right) \tag{4}$$
$$\hat{P}^s = Dec_\psi\left(Z^s_{pose}, Z^s_{traj}\right)$$

In the above equation, $\hat{P} \in \mathbb{R}^{T\times J\times3}$ represents the motion sequence generated through the decoding process. $\hat{P}^p$ denotes the motion sequence generated using motion encoding, and $\hat{P}^s$ denotes the motion sequence generated using text encoding. ψ denotes the parameters of the decoder. The similarity between $\hat{P}^p$ and $\hat{P}^s$ is enforced using our loss function. Finally, $\hat{P} = \hat{P}^s$ is considered as the final prediction of the network, i.e., the generated human skeleton animation sequence(s).

B. Loss Function

In this paper, we utilize the smooth L_1 loss as the distance metric during model training. This loss function offers more stability compared to the standard L_1 loss and is less sensitive to outliers than the L_2 loss. Specifically, the smooth L_1 loss is differentiable around $x = 0$ for $x \in \mathbb{R}$. The following losses are employed in the training process:

1) ***Action reconstruction loss***: This loss aims to minimize the discrepancy between the real motion P and the generated motion $\hat{P}^P$, $\hat{P}^S$.

$$L_{\{R\}} = L_1\left(P, \hat{P}^P\right) + L_1\left(P, \hat{P}^S\right) \qquad (5)$$

2) ***Reconstruction loss:*** This loss measures the distance between two latent variables.

$$L_E = L_1\left(Z_{traj}^P, Z_{traj}^S\right) + L_1\left(Z_{pose}^P, Z_{pose}^S\right) \qquad (6)$$

3) ***Motion speed reconstruction loss:*** This loss represents the disparity between the velocity of the real motion and the velocity of the generated motion.

$$L_{\{V\}} = L_1\left(P_{vel}, \hat{P}_{vel}^P\right) + L_1\left(P_{vel}, \hat{P}_{vel}^S\right) \qquad (7)$$

Here, $P_{vel}(t) = P(t+1) – P(t), \hat{P}_{vel}^P$ denotes the reconstructed motion speed, and $\hat{P}_{vel}^S$ denotes the speed of the motion reconstructed from the input text.

4) ***KL loss:*** For text-driven motion generation, it is important to consider the following KL divergences for better synthesis results:

$$\begin{aligned} L_{KL} &= D_{KL}\left(Z_{pose}^P \parallel Z_{pose}^S\right) + D_{KL}\left(Z_{pose}^S \parallel Z_{pose}^P\right) \\ &+ D_{KL}\left(Z_{pose}^p \parallel \mathcal{N}(0, I)\right) + D_{KL}\left(\mathcal{N}(0, I) \parallel Z_{pose}^p\right) \\ &+ D_{KL}\left(Z_{traj}^p \parallel Z_{traj}^s\right) + D_{KL}\left(Z_{traj}^s \parallel Z_{traj}^p\right) \\ &+ D_{KL}\left(Z_{traj}^p \parallel \mathcal{N}(0, I)\right) + D_{KL}\left(\mathcal{N}(0, I) \parallel Z_{traj}^p\right) \qquad (8) \end{aligned}$$

$D_{KL}\left(Z_{pose}^p \parallel \mathcal{N}(0, I)\right)$ and $D_{KL}\left(Z_{traj}^p \parallel \mathcal{N}(0, I)\right)$ represent the generated motion latent spaces and trajectory latent spaces, respectively, mapped to the standard Gaussian distribution from motion. $D_{KL}\left(Z_{pose}^s \parallel \mathcal{N}(0, I)\right)$ and $D_{KL}\left(Z_{traj}^s \parallel \mathcal{N}(0, I)\right)$ denote the generated motion latent spaces and trajectory latent spaces, respectively, mapped to the standard Gaussian distribution by the input text. While KL divergence measures the distance between two distributions, considering only the global alignment of the distributions is not sufficient. Semantic local alignment of the text with the motion is also crucial.

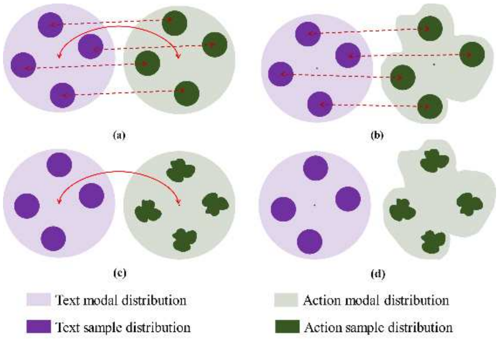


Figure 2. The mean-variance reconstruction loss is a novel metric function that addresses the challenge of achieving both global alignment and local alignment. Figure (a) illustrates the effectiveness of using both L_{MV} and L_{KL} losses. With this combination, not only can the distribution of the text modality align with the distribution of the motion modality, but also the alignment between the latent variables of the text-motion pairs can be achieved. Figure (b) demonstrates that without L_{MV} , only local alignment is achieved between the two modalities, lacking global alignment. Similarly, Figure (c) shows that without L_{MV} , only global alignment is achieved, lacking local alignment. In both cases, the generated motions do not conform to the semantic information. Figure (d) indicates that in the absence of both L_{MV} and L_{KL} losses, the model collapses posteriorly.

5) ***Mean-Variance loss:*** The mean-variance reconstruction encourages the two Gaussian distributions to be as possible as close to each other.

$$L_{MV} = \sum_{n=1}^{k} \left\|\tilde{Z}_\mu^P - \tilde{Z}_\mu^P\right\|^2 + \left\|\tilde{Z}_\sigma^P - \tilde{Z}_\sigma^P\right\|^2 \qquad (9)$$

Here, $\tilde{Z}_\mu^P$ and $\tilde{Z}_\sigma^P$ represent the mean and variance, respectively. 0 provides a depiction of the relationship between global loss and local loss.

6) ***Latent space orthogonal loss:*** To decouple the motion features and trajectory features, we propose the use of the latent space orthogonal loss in this paper. The orthogonality loss enforces vertical orthogonality between the two feature vectors, ensuring that the features extracted by the motion feature extractor and the trajectory feature extractor are decoupled from each other. This decoupling allows for more effective feature extraction and enhances the controllability of the generated motion.

$$L_{\{O\}} = \sum_{n=1}^{k} \left\|Z_{pose}^p \odot Z_{traj}^p\right\|^2 + \left\|Z_{pose}^S \odot Z_{traj}^S\right\|^2 \quad (10)$$

The symbol $\odot$ in the above equation represents the Hadamard product, which computes the element-wise multiplication of two vectors.

7) ***Discriminant loss:*** To enhance the generative capability of the generator and improve the quality of motion generation, a binary cross-entropy discriminator, denoted as D, is utilized. The discriminator is trained to distinguish between real and generated motions. The corresponding loss functions for the discriminator and the generator are defined as follows:

$$\begin{gathered} L_D = L_2\left(D\left(\hat{P}, 0\right)\right) + L_2\left(D\left(\hat{P}, 1\right)\right) \qquad (11) \\ L_G = L_2\left(D\left(\hat{P}, 0\right)\right) \end{gathered}$$

In the above equations, L2 denotes the binary cross-entropy loss. The generator refers to the two-stream decoupled Transformer network proposed in this paper.

The overall loss function is a weighted sum of the above loss functions.

$$\begin{gathered} min_{Enc,Dec}\,(\lambda_R L_R + \lambda_E L_E + \lambda_V L_V + \lambda_{KL} L_{KL} \\ + \lambda_{MV} L_{MV} + \lambda_O L_O + \lambda_G L_G\,) \\ \min_D(\lambda_G L_D) \qquad (12) \end{gathered}$$

The experiment sets $\lambda_R = \lambda_V = 1,\backslash\lambda_E = 0.1, \lambda_{KL} = \lambda_{MV} = \lambda_O = \lambda_G = 0.001$.

III. Experimental Results

A. Dataset

Similar to Ghosh [4], this paper utilizes the KIT motion-language dataset. The dataset consists of 3,911 recordings, totaling 11.23 hours of human skeletal animations, and includes 6,353 motions with their corresponding text descriptions. It is important to note that there is a one-to-many relationship between motions and text descriptions. Additionally, 899 motions in the dataset do not have text descriptions. To ensure the stability of model training, these samples are not used in the training process. In the experiments, we randomly split the dataset into training, validation, and testing sets in a 6:2:2 ratio. This splitting method is consistent with the baseline [4].

B. Evaluation Metric

In this paper, two evaluation metrics are used to assess the quality of the generated motion: Average Position Error (APE) and Average Variance Error (AVE). Similar to JL2P [5] and Ghosh [4], APE measures the average L_2 distance between the generated data $\widehat{H}$ and the real data H for j joints over N samples and F frames. A smaller APE indicates a higher similarity between the generated and real data.

$$\text{APE} = \frac{1}{NF}\sum_{n \in N}\sum_{f \in F}\left\|H_f[j] - \widehat{H}_f[j]\right\| \tag{13}$$

The second evaluation metric, AVE, calculates the average L_2 distance between the variance of the generated data and the variance of the real data for j joints. The variance $\sigma[j]$ represents the variation of joints within a sample over F frames. AVE measures the diversity of the generated motion.

$$\text{AVE} = \frac{1}{n}\sum_{n \in N}\left\|\delta[j] - \hat{\delta}[j]\right\| \tag{14}$$

C. Ablation Experiments

TABLE I. QUANTITATIVE RESULTS

	Method	root	global trajectory	mean w/o trajectory	mean with trajectory
APE	Lin [17]	7.78	4.52	26.64	25.63
	JL2P[5]	7.28	4.12	24.86	22.97
	Ghosh[4]	3.11	1.53	3.66	3.55
	ours	2.456	0.572	1.557	1.483
AVE	Lin17]	5.46	19	30.75	29.69
	JL2P[5]	4.7	18.55	30.96	29.58
	Ghosh[4]	0.43	0.52	0.49	0.48
	ours	0.339	0.137	0.334	0.327

For all ablation experiments, the results were averaged over three independent experiments with different randomized seeds.

1) ***Ablation Experiment 1:*** Two-stream decoupled Transformer network without mean-variance reconstruction loss (w/o MS). This experiment considers only the global loss in the training process without the local loss. The loss function includes L_R, L_E,L_V, L_{KL}, L_O, and L_G.

2) ***Ablation Experiment 2:*** Two-stream decoupled Transformer network without orthogonal decoupling (w/o OP). In this experiment, the loss function includes L_R, L_E, L_V, L_{KL}, L_{MS}, and L_G. without considering the decoupling of trajectory features from motion features.

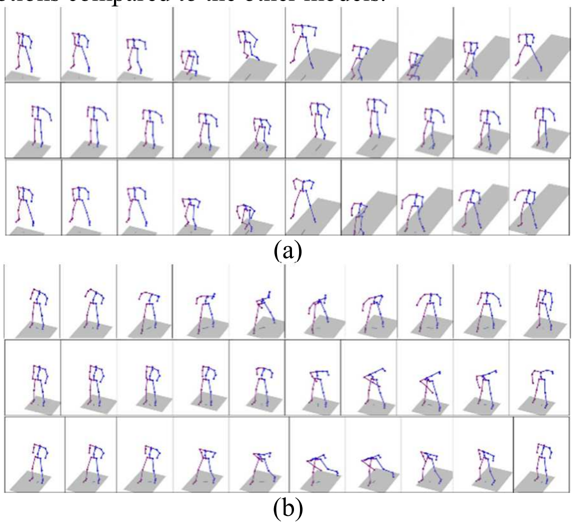
3) ***Ablation Experiment 3:*** Feature extraction without expert mixture network (w/o TS). In this experiment, a single motion encoder is used to represent the full body latent variables, and no motion data is separated from the trajectory data.

IV. EXPERIMENTAL RESULTS AND CONCLUSIONS

A. Quantitative results

We utilized the ADAM-W optimizer for optimization with a learning rate of 0.003. The minibatch size was set to 32, and a total of 300 rounds of training were conducted. The experiments were conducted on an Ubuntu 18.04.6 LTS platform with an Nvidia Tesla V100 GPU. Each experiment took approximately 8.6 hours to complete, and three independent runs were performed to obtain the average final results. It is worth noting that our experiments are sensitive to batch size, and using a batch size that is too large can lead to training failure. Unfortunately, we have not found a reasonable explanation for this phenomenon.

TABLE I. presents the experimental results of our model on the KIT dataset. The data in the table demonstrates that our model outperforms the state-of-the-art methods in terms of the APE and AVE evaluation metrics for different human joint parts. Specifically, our proposed model achieves an APE of 2.456 for the root node and 3.11 for the TextToMotion model. For the other body parts, our model shows improvements of 62.6%, 57.5%, and 58.3% over the existing methods. The lower APE values indicate that the motions generated by our model are closer to the real motions compared to the other models.



(a)

(b)

Figure 3. The above four images show the comparison between the ground truth, the benchmark program [4], and our method. In each image, the first row represents the ground truth, the second row represents the results obtained by the benchmark program, and the third row represents the results obtained by our method.

B. Qualitative results

Figure 3. presents four images, where the first row represents the ground truth, the second row shows the experimental results obtained by Ghosh[4], and the third row displays the experimental results of our method. Figure 3. (a) illustrates the motion generated with the input text "A person walks forward, turns around, and takes a few steps forward again." Figure 3. (b) represents the motion generated with the input text "A person sinks to his knees, supported on his hands, and then assumes an upright posture."

In all of the above examples, the input texts contain at least 10 words and encompass various semantic descriptions. The motions generated by our model appear natural, align well with the text descriptions, and demonstrate high fidelity.

C. Ablation experimental results

In this paper, three ablation experiments are conducted to evaluate the effectiveness of different components: (1)

Two-stream decoupled Transformer network without mean-variance reconstruction loss (w/o MS), (2) Two-stream decoupled Transformer network without orthogonal decoupling (w/o OP), and (3) Feature extraction without expert mixture network (w/o TS). The experimental results are presented in TABLE II.

The evaluation metrics demonstrate that the model proposed in this paper performs well across most metrics, highlighting the significance of the three innovations proposed in this study and the overall motion generation framework. These results confirm the importance of incorporating mean-variance reconstruction loss, orthogonal decoupling, and expert mixture networks in achieving superior performance.

TABLE II. ABLATION EXPERIMENTAL RESULTS

	Method	root	global trajectory	mean w/o trajectory	mean with trajectory
APE	ours	2.456	0.572	1.557	1.483
	w/o MS	2.983	0.76	1.575	1.532
	w/o OP	3.1	0.645	1.653	1.581
	w/o TS	3.226	0.86	1.669	1.627
AVE	ours	0.339	0.137	0.334	0.327
	w/o MS	0.397	0.142	0.367	0.351
	w/o OP	0.414	0.144	0.376	0.357
	w/o TS	0.434	0.146	0.393	0.374

V. SUMMARY AND FUTURE WORK

In this paper, we proposed a two-stream decoupled Transformer network for text-driven human animation generation. The proposed model surpasses state-of-the-art methods in both qualitative and quantitative evaluations. Our contributions can be summarized as follows:

1) We introduced expert networks to extract motion and trajectory manifolds separately, enabling enhanced extraction of semantic features from motions and texts.

2) By employing orthogonal loss, we decoupled the trajectory features from the motion features, improving controllability in motion generation.

3) We incorporated mean-variance reconstruction loss to enhance the local alignment between text and motion latent spaces. This improved the model's ability to balance global and local alignment, resulting in higher fidelity in generated motions and prevention of model posterior collapse.

REFERENCES

[1] D. Cudeiro, T. Bolkart, C. Laidlaw, A. Ranjan, and M. J. Black, "Capture, learning, and synthesis of 3D speaking styles," in Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition, 2019, pp. 10 101–10 111.

[2] G. Sun, Y. Wong, Z. Cheng, M. S. Kankanhalli, W. Geng, and X. Li, "Dependence: music-to-dance motion choreography with adversarial learning," IEEE Transactions on Multimedia, vol. 23, pp. 497–509, 2020.

[3] J. Kanji and D. I. Levin, "Convolutional humanoid animation via deformation," arXiv preprint arXiv:1908.04338, 2019.

[4] A. Ghosh, N. Cheema, C. Oguz, C. Theobalt, and P. Slusallek, "Synthesis of compositional animations from textual descriptions," in Proceedings of the IEEE/CVF International Conference on computer vision, 2021, pp. 1396–1406.

[5] C. Ahuja and L.-P. Morency, "Language2pose: Natural language grounded pose forecasting," in 2019 International Conference on 3D Vision (3DV). IEEE, 2019, pp. 719–728.

[6] M. Petrovich, M. J. Black, and G. Varol, "Actionconditioned 3d human motion synthesis with transformer vae," in Proceedings of the IEEE/CVF International Conference on Computer Vision, 2021, pp. 10 985– 10 995.

[7] J. Dehesa, A. Vidler, C. Lutteroth, and J. Padget, "Towards data-driven sword fighting experiences in vr," in Extended Abstracts of the 2019 CHI Conference on Human Factors in Computing Systems, 2019, pp. 1–6.

[8] V. Pérez-Rosas, R. Mihalcea, and L.-P. Morency, "Utterance-level multimodal sentiment analysis," in Proceedings of the 51st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2013, pp. 973–982.

[9] A. Zadeh, M. Chen, S. Poria, E. Cambria, and L.- P. Morency, "Tensor fusion network for multimodal sentiment analysis," arXiv preprint arXiv:1707.07250, 2017.

[10] Z. Liu, Y. Shen, V. B. Lakshminarasimhan, P. P. Liang, A. Zadeh, and L.-P. Morency, "Efficient low-rank multimodal fusion with modality-specific factors," arXiv preprint arXiv:1806.00064, 2018.

[11] X. Wei, T. Zhang, Y. Li, Y. Zhang, and F. Wu, "Multimodality cross attention network for image and sentence matching," in Proceedings of the IEEE/CVF conference on computer vision and pattern recognition, 2020, pp. 10 941–10 950.

[12] Z. Shi, T. Zhang, X. Wei, F. Wu, and Y. Zhang, "Decoupled cross-modal phrase-attention network for image-sentence matching," IEEE Transactions on Image Processing, 2022.

[13] Y. Li, F. Liang, L. Zhao, Y. Cui, W. Ouyang, J. Shao, F. Yu, and J. Yan, "Supervision exists everywhere: A data efficient contrastive language-image pre-training paradigm," arXiv preprint arXiv:2110.05208, 2021.

[14] J. Lee, J. Kim, H. Shon, B. Kim, S. H. Kim, H. Lee, and J. Kim, "Uniclip: Unified framework for contrastive language-image pre-training," arXiv preprint arXiv:2209.13430, 2022.

[15] G. E. Henter, S. Alexanderson, and J. Beskow, "Moonglow: Probabilistic and controllable motion synthesis using normalizing flows," ACM Transactions on Graphics (TOG), vol. 39, no. 6, pp. 1–14, 2020.

[16] H. Ahn, T. Ha, Y. Choi, H. Yoo, and S. Oh, "Text2action: Generative adversarial synthesis from language to action," in 2018 IEEE International Conference on Robotics and Automation (ICRA). IEEE, 2018, pp. 5915–5920.

[17] A. S. Lin, L. Wu, and Q. H. R. J. M. Rodolfo Corona, Kevin Tai, "Generating animated videos of human activities from natural language descriptions," in Proceedings of the Visually Grounded Interaction and Language Workshop at NeurIPS 2018, December 2018.

[18] M. Petrovich, M. J. Black, and G. Varol, "Temos: Generating diverse human motions from textual descriptions," in Computer Vision–ECCV 2022: 17th European Conference, Tel Aviv, Israel, October 23–27, 2022, Proceedings, Part XXII. Springer, 2022, pp. 480–497.