TEXT TO MOTION ENHANCED BY VIDEO-BASED KEYPOINTS

Tzu-Hsuan WU
Graduate Institute of Networking and Multimedia
National Taiwan University
No. 1, Sec. 4, Roosevelt Rd., Taipei 106319, Taiwan (R.O.C.)
r13944046@ntu.edu.tw

June 4, 2025

ABSTRACT

Recent advances in deep learning—particularly large language models, diffusion models, and vision transformers—have driven rapid progress in human motion generation. While text-to-motion frameworks excel at producing diverse behaviors from natural language, they often lack precise control over motion dynamics. Conversely, video-based reconstruction yields accurate motion but incurs high computational cost and limited generalization. To bridge this gap, we propose a video-augmented text-to-motion generation framework built upon AttT2M, a lightweight transformer-based model. Our approach first extracts 2D joint keypoints from monocular video using YOLOv8m-pose in the COCO format and encodes them into compact joint-vector representations. We then inject these video-derived features into AttT2M via two dedicated cross-attention blocks—one placed before and one after its original text-embedding attention—thereby enriching motion tokens with spatiotemporal cues. A tunable parameter $\alpha \in [\,0,\ 1]$ controls the balance between video-enhanced and text-only features, enabling users to trade off fidelity for diversity. Experiments on HumanML3D demonstrate that our method improves output stability and allows fine-grained style control, while maintaining low latency suitable for real-time applications.

1 Introduction

As deep learning techniques, such as large language models (LLMs), diffusion model, and vision transformers (ViTs), become increasingly prevalent, their applications have gone viral and are being leveraged across a wide range of fields. Human motion generation is also a growing research interest led by this trend, and gradually adopted into the production of animation, game, and even avatars in meta verse. Text-to-motion generation is widely used in human motion synthesis, capitalizing on the natural language processing capabilities of LLMs to extract correlated vectors in latent space to generate 3D human motion models. Owing to multi-modal learning models like CLIP [1] and GANs [2], we can strongly connect the semantics of natural language with a variety of media inputs, and achieve robust zero-shot predictions through large-scale pretraining. Another intuitive way to generate human motion models relies on videos of real people and uses deep learning models to reconstruct their motion. This approach is similar to conventional motion capture, which collects physical features from the human body in the real world through specific devices. However, 3D human motion reconstruction focuses on simple visual input—usually monocular RGB video—and aims to reduce the high cost of professional apparatus. It may allow flexibility in recording across various scenes, such as mountains, pools, or drama stages, due to the portability of the camera.

Despite the significant improvements enabled by deep learning networks, both approaches have their own limitations that remain to be overcome. Text-to-motion generation offers a wide range of human motions, yet it struggles to concisely control the generated output. This limitation can be critical in certain contexts, particularly where precise motion is required — such as game character interactions or CG animations in films. For example, the prompt "a man jumps straight to the left" may produce a jumping motion, but parameters like velocity, angle, distance, and even the character's emotion remain beyond the scope of text prompts and the user's control. On the other hand, video-based motion reconstruction rarely loses control. The reconstructed motion is built by features in videos, leaving no ambiguity for

neural networks during motion generation. Though research on lightweight models continues, visual input—especially sequential input like video—remains bulkier than text input and imposes a heavy load on computational resources.

To address such drawbacks, we adopt an eclectic resolution by importing an additional video-based keypoint features into text-to-motion frameworks. In our implementation, we opt for **AttT2M** [3] as our fundamental model for its lightweight and good property of temporal cross-attention with semantic labels. We first use **YoloV8** [4] to extract the spatial data of body joints in videos, adhering to Common Objects in Context (COCO) 17 keypoint format. Subsequently, we inject the keypoint features into the Text-driven Motion Generation Module in **AttT2M**, applying two cross-attention between the frame-wise motion embedding and keypoint features, positioned respectively before and after the original text-based attention module. In this way, we expect that our text-to-motion generation method could be more specified in certain styles or behaviors in supplementary videos, alleviating the problem of instability of output.

2 Related Work

Text-to-Motion Generation derives from the domain of human motion synthesis, has surged in recent years due to the rapid advancement of artificial intelligence tools, including transformer and LLMs. To enhance the quality of generated motions and the stability, researchers are still seeking better methods to connect the semantic features with motion features, comprising mixture of experts (MOE), motion diffusion model (MDM), improvements to encoders and decoders, and the use of prompt-based techniques.

For instance, **SPORT** [5] consists of three primary components—a body-part phase autoencoder, a body-part content encoder, and a diffusion-based decoder—and leverages their modular structure to combine their respective strengths, enhancing overall performance. The body-part phase autoencoder extracts the periodic features of human motion and is improved by separately learning different periodic features from individual body parts through the design of **BP-PAE**. Meanwhile, the body-part content encoder fuses semantic features from text using a CLIP model and GANs. It also adopts the Mixture-of-Experts (MoE) approach, training multiple network branches simultaneously to mitigate bias from a single expert.

Spatio-Temporal Motion Collage (STMC) [6] opts to a prompt-based method, which specifies a multi-track timeline of multiple prompts and stitches the independent intervals by SINC [7] (Spatial Stitching) and DiffCollage [8] (Temporal Stitching). This multi-track timeline prompt helps the Motion Diffusion Model (MDM) decompose content into clear time intervals and better understand complex text descriptions. **STMC** provides multiple time tracks for different body parts to implement specific actions. For example, a prompt like "walking straight" can be placed on the leg track. As long as creators follow such method to place correct prompts on a desired time spot, corresponding actions will be generated and composed together, which increasing raises the controllability against the conventional solutions.

In contrast, the Vector-Quantized Variational Autoencoder (VQVAE) is more popular in lightweight frameworks and requires relatively low computational resources compared to diffusion-based generation models. **AttT2M** [3] highlights a cross-attention mechanism between motion and text, which helps the word-level embedding with the latent code in motion space. At the first stage, a discrete latent motion representation is distilled by VQVAE, and passed to the second stage as an attention query. They use a two-layered attention design, which comprises a cross-attention with word-level features and a global motion-sentence conditional self-attention. The two-layered attention helps to learn the text-motion correlation, and send the predicted code back to the matching decoder for previous VQVAE to generate motion output.

Video-based human motion reconstruction has made rapid progress in recent years and leads to high-quality motion models. However, it remains limited by an inevitable reliance on computationally expensive optimization pipelines. **World-grounded Humans with Accurate 3D Motion (WHAM)** [9] selects an uni-directional RNN encoder and decoder instead of the common methods, which handle temporal features by windows with a fixed time duration. In **WHAM**, the authors introduce three types of inputs: image features, 2D keypoints—detected using ViTPose [10]—and camera parameters such as angular velocity coming from SLAM. The image features are integrated with 2D keypoints after feature extraction from separated encoders, and used to estimate motion pose through the paired RNN decoder. SLAM or gyroscope data is used to estimate the rough global root orientation and its velocity, helping calibrate the motion and direction in global coordinates.

3 Methodology

In our video-augmented method, we build upon the core architecture of **AttT2M** by integrating an additional cross-attention mechanism to learn auxiliary features from video input. Unlike text, video contributes quantifiable numerical data—such as pixel vectors—that offer more stability and less ambiguity than semantic cues. We, thus, introduce

video-derived representations based on simplified joint vectors extracted using **YOLOv8**'s pose estimation. The joint features only work as supplementary rather than dominated features that they were in the conventional 3D motion reconstruction, which enables to minimize computational demanding of videos and training overhead.

3.1 Joints Extraction through Yolov8

It is an efficient approach to concisely represent human motion using joint vectors, which capture detailed information about each movement. Position, velocity, and rotation are commonly used in this data format. A Other factors such as gait, ground contact, and motion direction [5, 11] are considered in various frameworks as well. **YOLOv8** provides an off-the-shelf API for extracting 2D joint coordinates based on the **COCO** format, consisting of 17 keypoints. **YOLOv8**'s high portability make it easy to integrate with existing frameworks, and offer an additional channel to receive the video supplementary fine-tuning generated motion. Moreover, it presents a good compatibility to video input during inference, allowing a robust text-to-motion generation having video as reference.In this case, users can constrain motion generation by desired video examples. The influence of the video can be adjusted using a condition parameter α, ranging from 0 to 1 to control the strength of video augmentation. This allows the model to preserve higher diversity in the primary text-to-motion generation when needed. We use the **YOLOv8m-pose** model to predict human body joints and pre-generate the results into JSON files to reduce training latency. The data consist of features extracted from each frame in the following format: $[\ frames,\ joints\ (17),\ 2D\ cordinates(x,\ y)\ ]$. we denote the data as **Joint vectors** and pass them to video-enhanced block, as shown in figure 1.

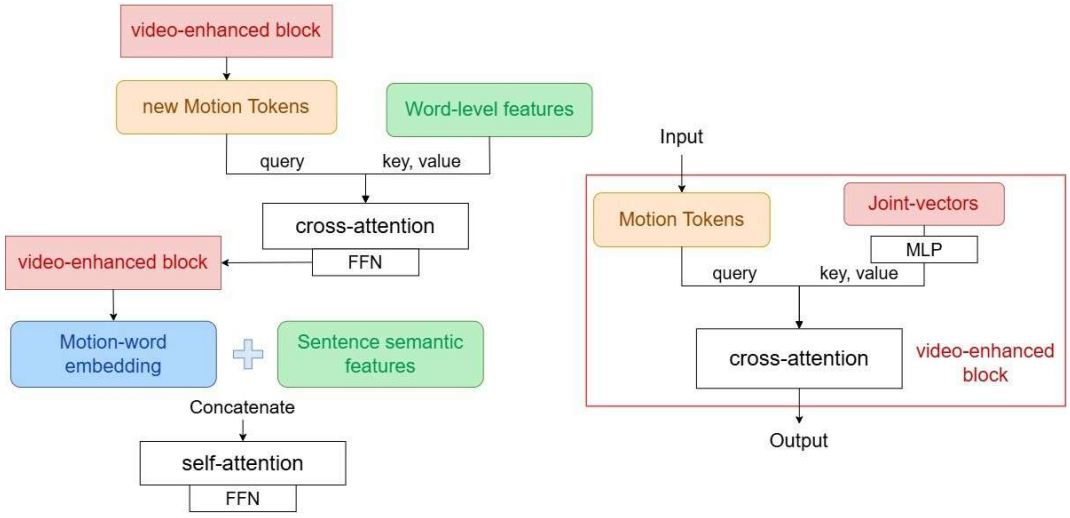

Figure 1: We adopt the **AttT2M** architecture and leverage cross-attention mechanisms to integrate auxiliary features within motion tokens. Two video-enhanced blocks are inserted in **AttT2M**'s original design-demonstrated in the left side of the above picture-to enrich the encoded representations.

3.2 Joint-Motion Cross-attention

In the **AttT2M**, Zhong et al. explored a text and motion cross-modal relationship learning. The word-level features and sentence-level semantic features are first extracted by CLIP. The word-level features, denoted as $[num\ of\ words,\ embedding\ token]$, are padded to a length of 70 if the number of keywords is fewer than 70, or truncated to 70 otherwise, to ensure consistent batch dimensions during training. Similarly, we have to align the frames of our joint vector, thus we have to truncate the too long video frames despite a trade-off missing completeness of behavior continuum. To match the average video length in **HumanML3D** dataset, we ultimately chose 180 as the maximum frame length, which is relatively short and corresponds to approximately 4–5 seconds. As a result, our framework's ability to generate motion is limited to short clips.

In detail, We squeeze the joint vectors into a one-dimensional representation on each frame: $[frames,\ joints\ (17),\ 2D\ cordinates(x,\ y)\ ] \rightarrow [frames,\ 34]$. Subsequently, the squeezed joint vectors are projected into into the same dimensional embedding space with motion tokens and word-level features by an MLP. This is subsequently complemented by pose embeddings, which help the attention mechanism recognize the sequential differences between frames. The processed joint vectors function similarly to word-level features, serving as key–value pairs to estimate the correspondence with the queried motion. We refer to this iterative step as the video-enhanced

block in our implementation shown in figure 1. During training, we freeze all parameters from the original AttT2M architecture, relying on its satisfying performance from pretraining, and update gradients of the MLP and newly introduced cross-attention layers only.

To verify the impact brought by our design, we employ a temperature parameter α to control the output from the video-enhanced block during testing. We use α to determine the proportion of enhanced features and video-free features illustrtaed as below.

$$\textbf{new Motion Tokens} = \alpha * \textbf{Video-Enhanced Output} + (1 - \alpha) * \textbf{Motion Tokens} \tag{1}$$

Here, α ranges from 0 to 1, serving as a gate for the video-based enhancement. When $\alpha = 0$, the new motion tokens remain identical to those generated by the original AttT2M, effectively disabling the enhancement. In contrast, when $\alpha = 1$, the motion tokens are entirely replaced by the output of the video-enhanced block. The effect of varying α is further analyzed in the ablation study in Section 5.1.

4 Dataset

A longstanding issue exists in the compatibility of joint formats and SMPL features. SMPL parameters encompass various aspects of human body modeling, each with its own dimensionality. The parameters include shape, pose, global translation, and rotation, and may vary across different datasets. The SMPL model employs a joint regressor to map mesh vertices to joint positions, while the varied joints set makes the problem more chaotic. Different applications and datasets utilize varying joint configurations. For instance, some employ a 17-joint set aligned with COCO keypoints, while others use a 22-joint set compatible with Human3.6 M. This variation necessitates different regressors, leading to inconsistent joint outputs during implementations.

To fine-tune the pre-trained model of AttT2M, we apply the same dataset, **HumanML3D** [12], as before to keep the consistency in the format of motion ground truth. **HumanML3D** is built on the **AMASS** [13] dataset, adding further semantic labels to construct a text-motion paired dataset. It generates abundant mirrored motions by flipping motions along the y-axis, but these are mostly restricted to low-level actions such as walking, jumping, and lifting objects. More abstract motion descriptions are often ignored—for example, opening a door or entering a room. Moreover, **HumanML3D** visualizes the motion data from **AMASS** dataset and renders it into SMPL model animations, which aligns well with our need for additional video input.

5 Experiments

In this section, we compare our video-enhanced design with the original AttT2M architecture, and present an evaluation on the *Controllability* and *Stability*. Here, we define the two key attributes, *Controllability* and *Stability*, to represent the robustness of the entire system. *Controllability* indicates how closely the generated motion matches the ground truth. It is a crucial metric, as we try to reduce the ambiguity inherent in natural language and improve the precision on reconstructing a desired motion based on users' intent. For instance, prompting "a man is dancing" might produce a waltz-style motion, which could significantly differ from the intended hip-hop dance. *Stability* refers to the ability to reproduce the specific motions without too much distortion or variation. This is especially important in practical application, where repetitive actions are widely used in animation and video games. Though there is an inevitable trade-off between the *Stability* and *Diversity* on motion generation, we introduce the temperature parameter α to balance the two and mitigate the potential degradation in output quality.

5.1 Experiment Design

Methods	FID↓	Diversity	R Precision↑		
			Top-1:	Top-2:	Top-3:
AttT2M	0.21917	9.5458	0.5452	0.7174	0.8132
Ours	0.16787	0.97741	0.6599	0.8309	0.9118

Table 1: Comparison between our method and the original AttT2M architecture. We evaluate both models on the testing set of HumanML3D dataset.

To prevent unnecessary modifications and reduce the risk of codebook collapse, we retain AttT2M's training configuration as much as possible during the fine-tuning of our video-enhanced block. Nevertheless, we adjust the batch size from 128 to 8 and reduce the total training iterations from 300,000 to 250,000 to accommodate our computational resources. The learning rate is set to 0.0001, with a decay rate of 0.02 every 100,000 iterations. As for the optimizer, we use AdamW, apply a dropout rate of 0.1, and set the probability of random masking during encoder training to 0.5. We ultimately save the model having the best FID score, which is selected from the periodic evaluation for every 5,000 iterations during training. And the comparison of training results between our fine-tuned video-enhanced model and AttT2M's pre-trained model is demonstrated in the table 1

Aside from the quantitative metrics, we then render several human motion models through our fine-tuned model to inspect what differences the video-enhanced blocks lead to. The temperature parameter α is leveraged to control the dependencies on the video input, forming a stepwise approach that allows us to delve deeper and observe such progressive improvements. We construct two experimental groups with $\alpha = 1$ and 0.5 to examine both a full and an eclectic integration of the video input. On the other hand, the control group shares the same output structure as AttT2M, with $\alpha = 0$. That is, the video-enhanced component is effectively disabled, leaving the text-to-motion design unchanged.

5.2 Results

We focus on two aspects in our analysis in this section, *Controllability* and *Stability*. As the Figure 2 shows, A significant distinction between the non-video-enhanced models and the video-enhanced ones. Without the video data—from which the joint vectors are extracted—the motion generation becomes noticeably less predictable. Although the generated output exhibits some typical characteristics of ballet dancing with respect to the prompt, it remains imprecise and deviates from the ground-truth video. In contrast, the experimental groups presents a high correspondence toward the input video. In the framewise decomposition of generated motion, The sequential actions of ballet dancing can be seen, and then follows with a twirling as we expect.

Figure 2: Comparison between motion generation results for different values of α. Using the prompts "the man is doing ballet, twirls then into a pose." We could find a significant *Controllability* improvement occur while $\alpha = 0.5$, validating the video-enhanced block helps shape the motion output by the auxiliary joint features.

To verify *Stability*, we visualize homogeneous results derived from identical outputs, testing their consistency under a given motion prompt. The three rows in Figure 3 correspond to the first through third outputs for the prompt "a person does breakdance." While there are slight variations, they all preserve the semantic content from text and closely adhere the constraint from joint vectors. A potential hazard exists to vandalize the diversity of motion generation while the temperature parameter α was too high, leading to an excessive constraints on output *Diversity*. Balancing the *Stability* and *Diversity* consumes a further research and examinations, which doesn't comprise in our current study.

References

[1] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision, 2021.

[2] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

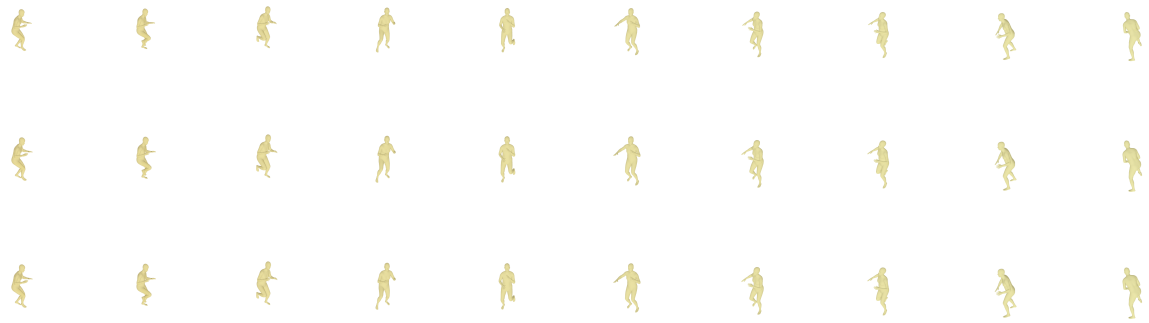

Figure 3: We generates three motions with the prompt "a person does breakdance" to validate the *Stability* of our design. As shown above, it is obvious to see the consistency among all of them while $\alpha = 1$, even in a frame scale.

[3] Chongyang Zhong, Lei Hu, Zihao Zhang, and Shihong Xia. Attt2m: Text-driven human motion generation with multi-perspective attention mechanism, 2023.

[4] Muhammad Yaseen. What is yolov8: An in-depth exploration of the internal features of the next-generation object detector, 2024.

[5] Bin Ji, Ye Pan, Zhimeng Liu, Shuai Tan, and Xiaokang Yang. Sport: From zero-shot prompts to real-time motion generation. *IEEE Transactions on Visualization and Computer Graphics*, pages 1–13, 2025.

[6] Mathis Petrovich, Or Litany, Umar Iqbal, Michael J. Black, Gül Varol, Xue Bin Peng, and Davis Rempe. Multi-track timeline control for text-driven 3d human motion generation, 2024.

[7] Nikos Athanasiou, Mathis Petrovich, Michael J. Black, and Gül Varol. Sinc: Spatial composition of 3d human motions for simultaneous action generation, 2024.

[8] Qinsheng Zhang, Jiaming Song, Xun Huang, Yongxin Chen, and Ming-Yu Liu. Diffcollage: Parallel generation of large content with diffusion models, 2023.

[9] Soyong Shin, Juyong Kim, Eni Halilaj, and Michael J. Black. Wham: Reconstructing world-grounded humans with accurate 3d motion. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 2070–2080, June 2024.

[10] Yufei Xu, Jing Zhang, Qiming Zhang, and Dacheng Tao. Vitpose: Simple vision transformer baselines for human pose estimation, 2022.

[11] Daniel Holden, Taku Komura, and Jun Saito. Phase-functioned neural networks for character control. *ACM Trans. Graph.*, 36(4), July 2017.

[12] Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. Generating diverse and natural 3d human motions from text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5152–5161, June 2022.

[13] Naureen Mahmood, Nima Ghorbani, Nikolaus F. Troje, Gerard Pons-Moll, and Michael J. Black. AMASS: Archive of motion capture as surface shapes. In *International Conference on Computer Vision*, pages 5442–5451, October 2019.