CoMo: Controllable Motion Generation through Language Guided Pose Code Editing Appendix

Yiming Huang[1], Weilin Wan[2], Yue Yang[1],
Chris Callison-Burch[1], Mark Yatskar[1], Lingjie Liu[1]

[1] University of Pennsylvania
[2] The University of Hong Kong

A Implementation Details

A.1 Model Architectures

We follow the setup in [4] for the decoder and transformer architectures. As Figure A1 illustrates, the Motion Decoder consists of 1D convolution layers, two residual blocks for upsampling, and ReLU activation layers. The codebook size is set to be 392×512. The Motion Generator uses a linear layer to project the K-hot pose code vectors before applying positional encoding and passing the sequence through a decoder-only transformer with causal self-attention blocks.

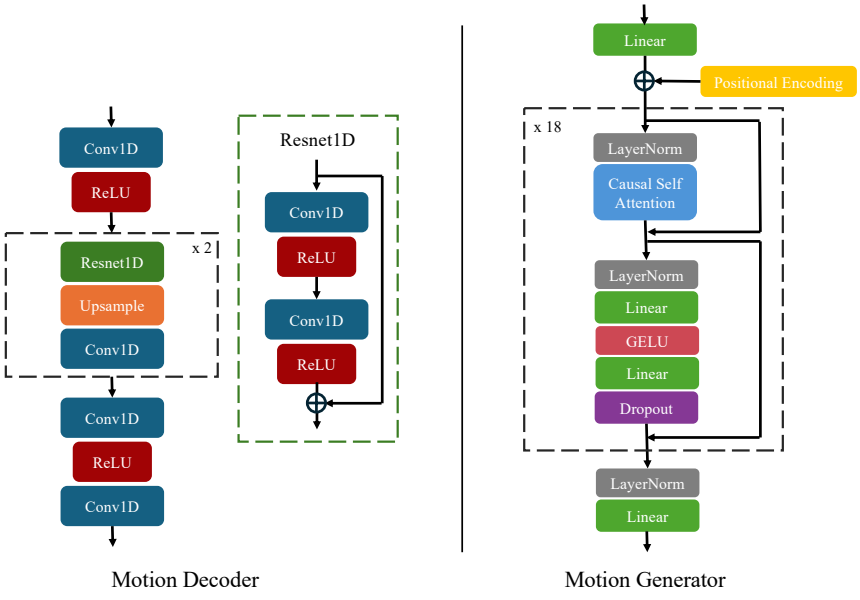


Fig. A1: Architectures of the Motion Decoder and Motion Generator of CoMo.

A.2 Pose Codebook

We follow the overall computation method of pose codes designated by [2] and adjust the heuristics to accommodate the task of motion generation. We select 70 pose code categories consisting of 392 pose codes. For each pose code category, a corresponding type of heuristic threshold is applied to define the pose codes within each category (e.g., "L-knee angle" will have a set of angle thresholds). Table A2 shows the 7 different types of heuristic thresholds used. Amongst the 70 pose code categories, 4 use the angle threshold, 18 use the distance threshold, 31 use relative position thresholds (6 along the x-axis, 16 along the y-axis, 9 along the z-axis), 13 use the relative orientation threshold, and 4 use the ground-contact threshold. Table A1 shows a list of the 70 pose code categories grouped according to the type of threshold they use. The semantics of pose codes are defined by combining the pose code category names, which indicate the joints involved, with the joint states described by the threshold conditions.

Mutual Exclusivity Although we frame our optimization objective concerning independent Bernoulli variables, certain pose codes can be mutually exclusive. For example, the concepts "L-arm below torso" and "L-arm above torso" are contradictory and should not co-occur. The encoded motion sequence is structured as K-hot, representing K pose code categories, each corresponding to a subset of mutually exclusive codes. During training, mutual exclusivity is ensured within the ground truth representation. During inference, we enforce this mutual exclusivity for each of the K categories by activating the pose code with the highest log-likelihood within its corresponding category. We can also generate diverse motion sequences by sampling from the predicted distributions of the K categories provided by the transformer model (see Figure C6).

Sequence Corruption. To mitigate the discrepancy between training and testing, we randomly replace the tokens within each subset during training while maintaining mutual exclusivity. Additionally, we perform random masking on the fine-grained, part-specific text conditions to facilitate effective text-based control.

A.3 Fine-grained Keyword Setup

Figure A2 presents the prompt template used to interact with GPT-4 to generate fine-grained keywords that enhance the text descriptions for the motion sequences. For each motion sample, we generate 5 different sets of fine-grained keywords using the original text description as input. During training, the keywords paired with the motion samples are chosen randomly from these five sets. An example of a text description from the HumanML3D training set [3] and the corresponding generated fine-grained keywords is presented in Table A3. The generated keywords expand the original text description to include additional details that depict the motion of specific body parts. More examples of text descriptions, generated fine-grained keywords, and the corresponding generated motion sequence are shown in Figure C6.

Table A1: Pose Code Categories. A total of 70 categories of pose codes are used, grouped according to their threshold type. We denote Left and Right as 'L' and 'R' respectively. The axes considered for relative positions are noted in parentheses.

Angle	Distance	Relative Position	Relative Orientation	Ground-contact
L-knee	L-elbow vs R-elbow	L-shoulder vs R-shoulder (YZ)	L-hip vs L-knee	L-knee
R-knee	L-hand vs R-hand	L-elbow vs R-elbow (YZ)	R-hip vs R-knee	R-knee
L-elbow	L-knee vs R-knee	L-hand vs R-hand (XYZ)	L-knee vs L-ankle	L-foot
R-elbow	L-foot vs R-foot	neck vs pelvis (XZ)	R-knee vs R-ankle	R-foot
	L-hand vs L-shoulder	L-ankle vs neck (Y)	L-shoulder vs L-elbow	
	L-hand vs R-shoulder	R-ankle vs neck (Y)	R-shoulder vs R-elbow	
	R-hand vs L-shoulder	L-hip vs L-knee (Y)	L-elbow vs L-wrist	
	R-hand vs R-shoulder	R-hip vs R-knee (Y)	R-elbow vs R-wrist	
	L-hand vs R-elbow	L-hand vs L-shoulder (XY)	pelvis vs L-shoulder	
	R-hand vs L-elbow	R-hand vs R-shoulder (XY)	pelvis vs R-shoulder	
	L-hand vs L-knee	L-foot vs L-hip (XY)	pelvis vs neck	
	L-hand vs R-knee	R-foot vs R-hip (XY)	L-hand vs R-hand	
	R-hand vs L-knee	L-wrist vs neck (Y)	L-foot vs R-foot	
	R-hand vs R-knee	R-wrist vs neck (Y)		
	L-hand vs L-foot	L-hand vs L-hip (Y)		
	L-hand vs R-foot	R-hand vs R-hip (Y)		
	R-hand vs L-foot	L-hand vs torso (Z)		
	R-hand vs R-foot	R-hand vs torso (Z)		
		L-foot vs torso (Z)		
		R-foot vs torso (Z)		
		L-knee vs R-knee (YZ)		

A.4 Metrics.

For quantitative evaluation, we obtain motion and text feature vectors using the motion feature extractor and text feature extractor pretrained in [3] and calculate the following metrics:

FID evaluates the quality of generated motion by computing the difference in the mean and covariance of the motion features.

R-Precision measures the consistency between text and generated motion. The ground truth text description and a set of mismatched text descriptions are selected for each generated motion to form a pool. The descriptions in the pool are ranked based on the Euclidean distance between their text feature and the generated motion feature, with smaller distances ranking higher. R-precision (Top-k) calculates the average probability of the ground truth text description ranking within the Top-k candidates.

MM-Dist computes the average Euclidean distance between a text feature and the corresponding generated motion feature over N randomly generated pairs.

Diversity computes the average Euclidean distance between pairs of generated motion features over M randomly generated pairs and indicates the variance of generated motion.

MModality determines the diversity of generated motion for the same text condition. For each text description, several motions are generated, and the average

Threshold Type	Semantics	Threshold Condition
Angle	bent to almost 10 degrees	$x \leq 10$
	bent to almost 20 degrees	$10 < x \leq< 20$
	bent to almost 30 degrees	$20 < x \leq 30$
	bent to almost 40 degrees	$30 < x \leq 40$
	bent to almost 50 degrees	$40 < x \leq 50$
	bent to almost 60 degrees	$50 < x \leq 60$
	bent to almost 70 degrees	$60 < x \leq 70$
	bent to almost 80 degrees	$70 < x \leq 80$
	bent to almost 90 degrees	$80 < X \leq 90$
	bent to almost 100 degrees	$90 < x \leq 100$
	bent to almost 110 degrees	$100 < x \leq 110$
	bent to almost 120 degrees	$110 < x \leq 120$
	bent to almost 130 degrees	$120 < x \leq 130$
	bent to almost 140 degrees	$130 < x \leq 140$
	bent to almost 150 degrees	$140 < x \leq 150$
	bent to almost 160 degrees	$150 < x \leq 160$
	bent to almost 170 degrees	$160 < x \leq 170$
	straight	$x > 170$
Distance	very close	$x \leq 0.1$
	slightly close	$0.1 < X \leq 0.2$
	close	$0.2 < x \leq 0.3$
	almost shoulder width apart	$0.3 < x \leq 0.4$
	shoulder with apart	$0.4 < x \leq 0.5$
	almost spread	$0.5 < x \leq 0.6$
	spread	$0.6 < x \leq 0.7$
	slightly wide	$0.7 < x \leq 0.8$
	wide	$0.8 < x \leq< 0.9$
	very wide	$x > 0.9$
Relative Position along X axis	at the right of	$x \leq -0.15$
	ignored	$-0.15 < x \leq 0.15$
	at the left of	$x > 0.15$
Relative Position along Y axis	below	$x \leq -0.15$
	ignored	$-0.15 < x \leq 0.15$
	above	$x > 0.15$
Relative Position along Z axis	behind	$x \leq -0.15$
	ignored	$-0.15 < x \leq 0.15$
	in front of	$x > 0.15$
Relative Orientation	vertical	$x \leq 10$
	ignored	$10 < x \leq 80$
	horizontal	$x \leq 80$
Ground-contact	on the ground	$x \leq 0.1$
	ground-ignored	$x > 0.1$

Table A2: Pose Code Threshold Conditions. For each pose code category, a corresponding threshold type is applied to specify the semantics of pose codes within that category. x represents the input value. Angles are represented in degrees, distances/relative positions/ground contact are represented in meters, and relative orientation is represented by the cosine similarity between unit vectors along the y-axis.

```
Given a text description of a motion: {details}. Enrich the description of the full motion by
summarizing in detail the shape and speed for each of the body parts in {body_parts} that is
required to achieve the given motion in natural language. The output should be in json format
with {body_parts} as keys, and one short motion attribute as values. Key-value format example:
"head":"head is upright". Do not output anything else.
```

```
Given a text description of a motion: Please help me to describe the mood that is required
to achieve the human motion described as: '{details}' using one short motion attribute. Do not
output anything else.
```

Fig. A2: Prompt template for generating fine-grained keywords.

Table A3: An example of generated fine-grained keywords on HumanML3D.

Text		a man stumbles to his right
Fine-grained keywords	Head	The man's head tilts slightly forward and turns to the right, following the direction of the stumble
	Torso	The man's torso leans to the right, as if losing balance.
	L-Arm	The man's left arm swings outward to the left, in an instinctive attempt to regain balance
	R-Arm	The man's right arm tucks in towards his body as he stumbles to the right
	L-Hand	The man's left hand is open, ready to grasp anything in the vicinity for support if needed
	R-Hand	The man's right hand clenches slightly, moving in unison with the right arm
	L-Leg	The man's left leg is firmly planted, acting as the pivot for the stumble
	R-Leg	The man's right leg lifts and steps awkwardly to the right, causing the stumble
	L-Feet	The man's left foot remains grounded, providing the only source of stability during the stumble
	R-Feet	The man's right foot lands unevenly on the ground, leading to the stumble
	Mood	Unsteady

distance between the generated motion features is computed. This value is then averaged across all text descriptions.

A.5 Motion Editing Prompts

Figure A3 and Figure A4 present the prompt templates used to interact with GPT-4 for motion editing on pose codes. Pose code semantics are provided as a table to provide context for the LLM to interpret encoded motion sequences.

A.6 Inference Time

We follow the metric in [1] and calculate the Average Inference Time per Sentence (AITS) of our approach and T2M-GPT [4], which has a similar architecture

```
Motion is represented by a set of joint states, defined as follows:
Table 1 Joint State Meanings (Key: Joint State Index, Value: Joint State Meaning): {table1}
Given the edit instruction: {edit}
Return a semi-colon separated sequence of the ids of the joint states you will need to examine
in order to determine the starting and ending frame of a motion sequence that will be affected
by the edit instruction.
Format example: 0;1;5;9. Do not reply anything else.
```

```
You will be provided with a text description of the motion, a motion code sequence and a motion
edit instruction. You are be required to determine the starting and ending frame of the sequence
that will be affected by the edit. Here is what you need to know about the encoding of the motion
sequences:
The motion is represented a number of time frames, each time frame contains a set of joint states,
each joint state contains a code value. The definitions are:
Table 1 Joint State Meanings (Key: Joint State Index, Value: Joint State Meaning): {table1}
Table 2 Code Meaning (Key: Code ID, Value: Code Meaning): {table2}
Rules: smaller angles indicates more bending.
The motion code sequence is: {codes}
The total number of time frames is {length}
The text description is: {details}
The edit instruction is: {edit}
Return the starting index and ending index of the segment that is affected by the edit, separated
by semi-colon, if the edit affects the overall movement, select the entire sequence. Format
example: 0;19. Do not reply anything else.
```

Fig. A3: Prompt template for identifying the frames for editing.

```
Motion is represented by a set of joint states, defined as follows:
Table 1 Joint State Meanings (Key: Joint State Index, Value: Joint State Meaning): {table1}
Given the edit instruction: {edit}
Return a semi-colon separated sequence of the ids of the joint states you may be affected by the
edit instruction. Format example: 0;1;5;9. Do not reply anything else.
```

```
You will be provided with a text description of the motion, a motion code sequence for a given
joint state and a motion edit instruction. You will be required to determine how to modify the
codes within the provided sequence accordingly.
Here is what you need to know about the encoding of the motion sequences: The motion is
represented as a list of joint states of length T, T is the number time frames. Each joint
state contains a code value. The usable codes are defined as follows:
Table 1 Usable Code Meaning (Key: Code ID, Value: Code Meaning): {table2}
Rules: smaller angles indicates more bending.
You are given this motion code sequence for the joint state {joint}, it has already been sliced
to keep only the segment you will need to edit: {codes}.
The text description of the overall motion sequence is: {details}.
The edit instruction is: {edit}
Return the edited motion only as a sequence of integer code ids of length {length} separated by
semi-colons, only use code ids in the provided table. If no edit needs to be made, return the
original sequence. Format example: 1;2;3;4. Do not reply anything else. No explanation needed.
```

Fig. A4: Prompt template for identifying the body parts/joints for editing (above) and the prompt for executing the edits (below).

setup on the test set of HumanML3D [3]. AITS corresponds to the time cost in seconds for a model to generate one motion sequence, excluding the time costs for model and dataset loading. For our model, the time cost would cover both the generation of the pose code sequence and the decoding of pose codes

into the final motion sequence. As shown in Table A4, despite requiring longer sequence generation due to the addition of fine-grained keywords, our approach achieves new motion editing capabilities while maintaining competitive motion generation performance with marginal inference time increment.

Table A4: Comparison of the Average Inference Time per Sentence costs on HumanML3D Test Set [3] with NVIDIA RTXA6000 GPU.

Methods	AITS (s) ↓
Ours	0.62
T2M-GPT [4]	0.35

B Human Evaluation

Figure B5 shows a screenshot of our annotation interface. Users are provided with the source description and motion, the edit instruction, and the two edited motions from two different systems. We use the following prompt template to obtain updated descriptions for the baselines to generate edited motions:

```
Given a source motion description: {details} and an edit instruction: {edit}, provide an updated
motion description that describes the motion after applying the edit. The updated description
should be similar in style as {examples}. Do not reply anything else.
```

10 randomly chosen text descriptions from the training set are provided as examples to guide the LLM to generate updated text descriptions in a similar style to the dataset annotations. Examples of the updated text descriptions used by the baselines are shown in Table B5.

Table B5: Example updated descriptions for baseline motion editing.

Text Description	Edit Instruction	Updated Description
A person lowers their arms, and then moves them back up to shoulder height	Keep both knees deeply bent	A person, with both knees deeply bent, lowers their arms and then raises them back up to shoulder height
The person bent down and dodge something towards the left	Bend down slower	The person slowly bent down and dodged something towards the left
A person walks forward and then appears to bump into something, then continues walking forward	Make the bump more dramatic	A person walks forward, then suddenly collides with a large unseen obstacle with a significant impact, recoiling notably before resuming their forward motion
A person beginning to run in a straight line	Raise left hand at the end	A person begins to run in a straight line and raises their left hand near the end of the run

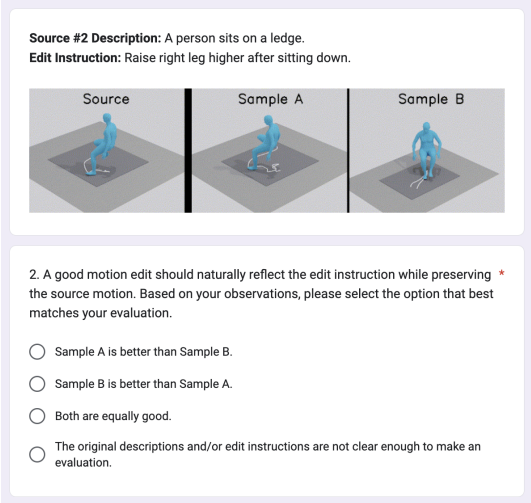


Fig. B5: An example of our annotation interface in our user study. The option "Both are equally good" indicates the case with no clear advantage over the baselines, viewed as a negative evaluation during analysis.

A total of 54 graduate students participated in this user study. The Fleiss' kappa measurement for annotation agreement is 0.4, indicating moderate agreement among the raters.

C Additional Qualitative Examples

C.1 Motion Generation Examples

In C6, we generate three motion samples for each text description under the same text condition with fine-grained keywords from the HumanML3D test set. The results demonstrate the diversity of the generated motion and consistency between the motion and text conditions.

C.2 Motion Editing Examples

In C7, we present qualitative examples for iterative motion editing, where a motion sequence is first generated from a text condition and then is edited iteratively by two edit instructions. The results demonstrate the capability of our approach for continuously interpreting and editing motion sequences, enabling both effective motion edits and the preservation of useful motion characteristics from previous iterations.

Failure Cases: As shown in C8, the semantics of pose codes focus on local kinematic attributes, which provides helpful context for LLMs to edit local joint states. However, for edits that require global changes in emotion or speed, the

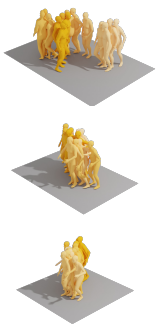
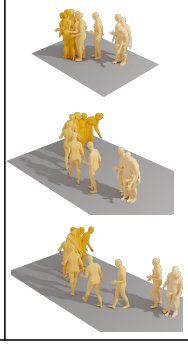
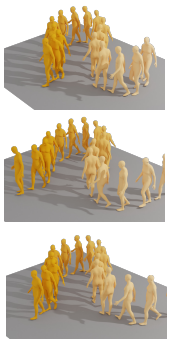
Text Description		a person walks forwards, sits.	figure appears to be fighting or dancing	person walks to pick something up then walks back to wipe something with it.	a person walks in a s shape
Fine-grained Keywords	Head	The head stays upright, looking forward with a steady pace	The head is held upright, with sharp, quick movements from side to side mimicking the rhythm of the motion	The head remains level and steady, turning slightly as the person looks at the object they need to pick up and the place where they need to wipe	The head remains upright, subtly swaying side to side in alignment with the torso
	Torso	The torso remains upright and slightly leans forward due to walking motion, then transitions to a seated position	Torso is slightly tilted forward, twisting and bending rhythmically to the right and left	The torso bends slightly forwards during the pickup and wiping actions, and remains upright during walking	The torso sways gently side to side, leading the movement and creating the 'S' shape
	L-Arm	The left arm swings back and forth in a natural rhythm with the walking motion, then rests on the left thigh when seated	The left arm is engaged in abrupt swings; at times it moves fast, then slows down, following the rhythm of the actions	The left arm swings naturally during walking, bends at elbow during pickup, and makes a forceful linear motion during the wiping action	The left arm swings in a complementary rhythm to the right leg's movement, bending at the elbow
	R-Arm	Similarly, the right arm swings in coordination with the left, opposite to the stride of the legs, then rests on the right thigh.	The right arm moves in similar sporadic swings as the left arm, as if throwing punches or performing dance moves	The right arm swings naturally during walking, is stationary during pickup with the left hand, and assists the left arm during the wiping action	The right arm swings in counterbalance to the left leg's movement, bending at the elbow
	L-Hand	The left hand remains relaxed, swinging in sync with the left arm, then rests on the left thigh	Left hand is either clenched in a fist or open, occasionally reaching out as if to touch or strike something	The left hand moves in sync with the left arm swing during walking, closes to grip the object during pickup, and makes a forceful scrubbing motion during the wiping action	The left hand relaxed, moving in rhythm with the arm swing but remaining relatively steady
	R-Hand	The right hand is also relaxed and follows the motion of the right arm, then rests on the right thigh	Right hand mimics the motion of the left, either clenched or open, depending on the rhythm and flow of the movements.	The right hand moves in sync with the right arm swing during walking, remains open during pickup with the left hand, and assists the left hand during the wiping action	The right hand is relaxed, following the motion of the right arm but remaining relatively steady
	L-Leg	The left leg alternates with the right in a forward stepping motion, then bends at the knee to assume a seated position	The left leg provides support and balance, bending at the knee and shifting weight when needed, moving in sync with the torso's movements	The left leg moves forward in a steady pace during walking, bends at the knee during pickup, and maintains balance during the wiping action	The left leg steps out to the left, then curves back in towards the right, creating one half of the 'S' shape.
	R-Leg	The right leg alternates with the left in a forward stepping motion, then also bends at the knee when sitting	The right leg moves in a similar manner to the left, stepping forward or backward in rhythm with the body's motion	The right leg alternates with the left leg in a steady walking motion, supports the body weight during pickup and provides balance during the wiping action	The right leg steps out to the right, then curves back in towards the left, creating the other half of the 'S' shape.
	L-Feet	The left foot steps forward alternately, heel touching the ground first, then rolls onto the toe in the walk. It rests flat when seated	Left foot is stabilizing the body during movements, with swift motions, constantly adjusting the balance	The left foot rolls from heel to toe during walking, adjusts for balance during pickup, and maintains a firm stance during the wiping action	The left foot leads and finishes each step, pivoting and adjusting to maintain balance during the curved walking motion
	R-Feet	The right foot also steps forward alternately, heel first, then rolls onto the toe. It too rests flat when sitting	Right foot mimics the movement of the left foot, used for both balance and propulsion, occasionally lifting off the ground	The right foot alternates with the left foot in a steady walking motion, supports body weight during pickup, and provides a pivot point during the wiping action	The right foot leads and finishes each step, pivoting and adjusting to maintain balance during the curved walking motion
	Mood	Determined	Aggressive elegance	Determined	Playful
Generated Motion Samples					

Fig. C6: Qualitative examples of diverse motion generation.

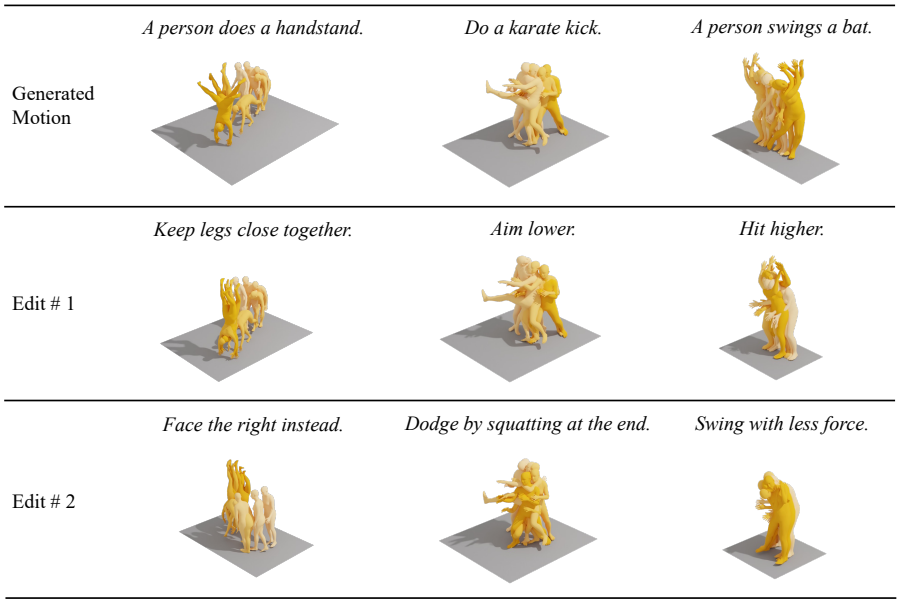


Fig. C7: Qualitative examples of iterative motion editing. The Motion Generator generates the initial motion sequence using the provided text description. The Motion Editor then iteratively edits the pose code sequence based on edit instructions with previous edits preserved.

LLM may struggle with interpreting how the global edit translates to fine-grained modifications of local attributes. In addition, for more complex motion sequences with faster movement, the LLM tends to choose a broader range of frames when determining which frames to edit, which may limit the precision of the edit being made. In such cases, the user may intervene and directly select the time frames they want to edit.

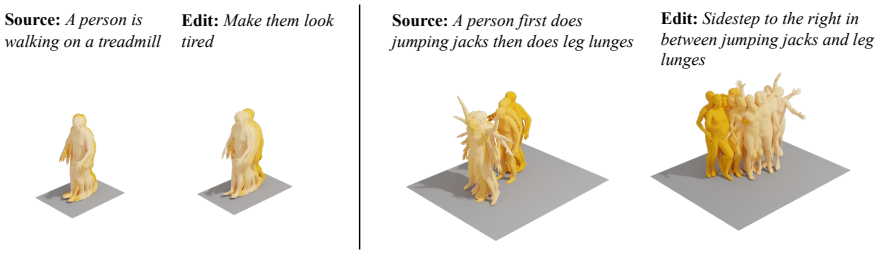


Fig. C8: Failure cases in motion editing. **Left:** The edited motion does not depict the target emotion adequately. **Right:** The edited motion mistakenly added the 'sidestep' near the start of the motion rather than in between the two exercises.

References

1. Chen, X., Jiang, B., Liu, W., Huang, Z., Fu, B., Chen, T., Yu, G.: Executing your commands via motion diffusion in latent space. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 18000–18010 (2023)
2. Delmas, G., Weinzaepfel, P., Lucas, T., Moreno-Noguer, F., Rogez, G.: Posescript: 3d human poses from natural language (2022)
3. Guo, C., Zou, S., Zuo, X., Wang, S., Ji, W., Li, X., Cheng, L.: Generating diverse and natural 3d human motions from text. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR). pp. 5152–5161 (June 2022)
4. 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: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2023)