


A SUPPLEMENTARY MATERIAL

In addition to the subsequent appendix sections, we provide the following as supplementary material for this work:

- **Demo Videos:** We provide several demonstration videos showcasing visual interaction results generated using InterMask. These include an animation gallery with generated interactions for everyday actions, dance and combat; nuanced description demonstration to highlight the capability of InterMask to follow specific details in text; diversity demonstration with multiple generated interactions for each text prompt; comparison videos to compare the generated results of InterMask and InterGen (Liang et al., 2024); comparison videos for the ablation study on Inter-M Transformer; an animation gallery to show reaction generation results; some results on more complex prompts; some results showing longer 10 second interactions; and finally some failure cases and some results addressing the concerns on fluidity of the generated motions. The videos are shown in form of a webpage, provided as an html file.
- **Code Implementation:** We provide the open-source code implementation of our method to ensure reproducibility.

 github.com/gohar-malik/intermask

B MOTION VQ-VAE

Our VQ-VAE framework, illustrated in Figure 7, constructs a 2D motion token map to represent individual motion sequences in a discrete manner, while retaining both spatial and temporal dimensions. The encoder processes motion sequences represented as $\mathbf{m}_p \in \mathbb{R}^{N \times J \times d}$, where N is the number of poses, J is the number of joints, and d is the joint feature dimension. By employing 2D convolutional layers, the encoder effectively captures spatial and temporal dependencies within the motion data while progressively downsampling both dimensions. The downsampling process is achieved through strided 2D convolutions and ResNet blocks, which also use 2D convolutions along with dropout layers. This results in a latent representation of size $\tilde{\mathbf{t}}_p \in \mathbb{R}^{n \times j \times d'}$, where n and j are the downsampled temporal and spatial dimensions, and d' is the latent feature dimension.

The latent representation $\tilde{\mathbf{t}}_p$ is quantized using a learned codebook \mathcal{C} with $|\mathcal{C}|$ entries. Each feature vector \tilde{t}_i in $\tilde{\mathbf{t}}_p$ is replaced by the index of its nearest codebook entry, using the vector quantization process:

$$\mathbf{q}(\tilde{t}_i) = \arg \min_{c_k \in \mathcal{C}} \|\tilde{t}_i - c_k\|^2, \quad (9)$$

where c_k represents the codebook entries.

The resulting quantized representation is a 2D motion token map t_p , where each token encodes local spatio-temporal context. This design enables the preservation of both spatial and temporal dimensions in the motion data, enhancing the model’s ability to generate realistic and contextually accurate interactions.

C VQ-VAE GEOMETRIC LOSSES

Equation (10) shows the geometric losses used to train our Motion VQ-VAE to impose physical and geometric constraints on the reconstructed motion, in a data-driven way. The velocity loss \mathcal{L}_{vel} encourages the reconstructed motion sequences to obey the velocity of joints in the ground truth sequences and the bone length loss \mathcal{L}_{bl} the distance between adjacent joints. The foot contact loss \mathcal{L}_{fc} encourages the feet to have zero velocity whenever they are in contact with the ground.

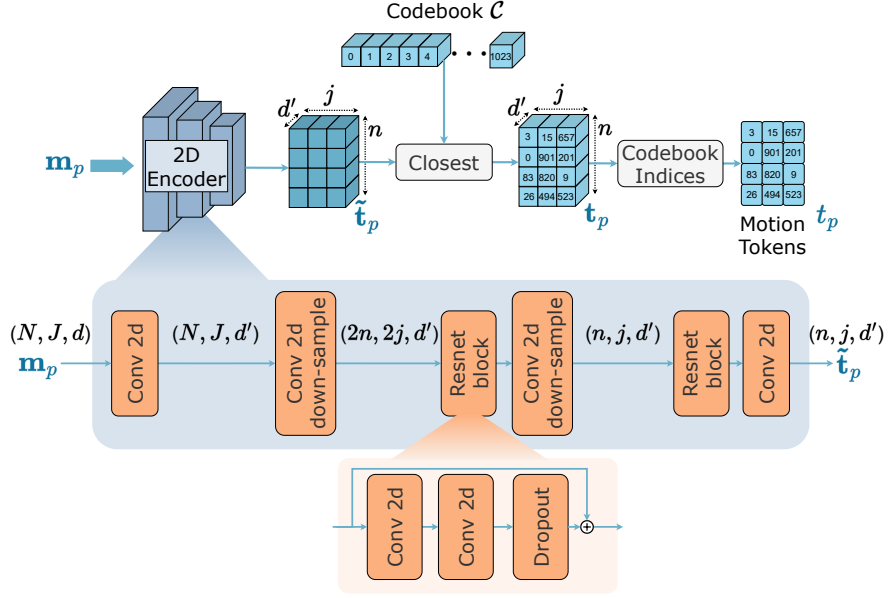


Figure 7: Detailed illustration of the 2d discrete motion token map construction. The 2d encoder, consisting of 2d convolutional layers, downsamples the input motion from (N, J, d) to (n, j, d') . The downsampled representation is then quantized by replacing each vector with the index of its closest vector in the learned codebook.

p_r	Interaction Generation		Reaction Generation	
	FID ↓	R Prec (Top1) ↑	FID ↓	R Prec (Top1) ↑
0.7	5.214	0.447	2.850	0.476
0.8	<u>5.154</u>	<u>0.449</u>	<u>2.991</u>	<u>0.462</u>
0.9	5.152	0.450	3.368	0.416

Table 4: Interaction Generation and Reaction Generation results of different values of p_r . **Bold** face indicates the best result, while underscore refers to the second best.

$$\begin{aligned}
\mathcal{L}_{vel} &= \frac{1}{N-1} \sum_{i_n=1}^N \|(m_{i_n+1} - m_{i_n}) - (\hat{m}_{i_n+1} - \hat{m}_{i_n})\|_1 \\
\mathcal{L}_{fc} &= \frac{1}{N-1} \sum_{i_n=1}^N \|(\hat{m}_{i_n+1} - \hat{m}_{i_n}) \cdot f_{i_n}\|_1 \\
\mathcal{L}_{bl} &= \frac{1}{N-1} \sum_{i_n=1}^N \|B(m_{i_n}) - B(\hat{m}_{i_n})\|_1
\end{aligned} \tag{10}$$

Here, m_{i_n} represents the ground pose, \hat{m}_{i_n} represents the reconstructed pose at time step i_n , N represents the total time steps in the sequence, $f_{i_n} \in \{0, 1\}$ represents the binary foot contact label for the heel and toe joints for each pose m_{i_n} , and $B(\cdot)$ denotes the bone lengths joining adjacent joints.

D TWO-STAGE TOKEN MASKING

Here we provide more details on the masking strategy (section 3.2.1) used in training the Inter-M Transformer. As illustrated in Figure 8, the two-stage masking technique begins with random

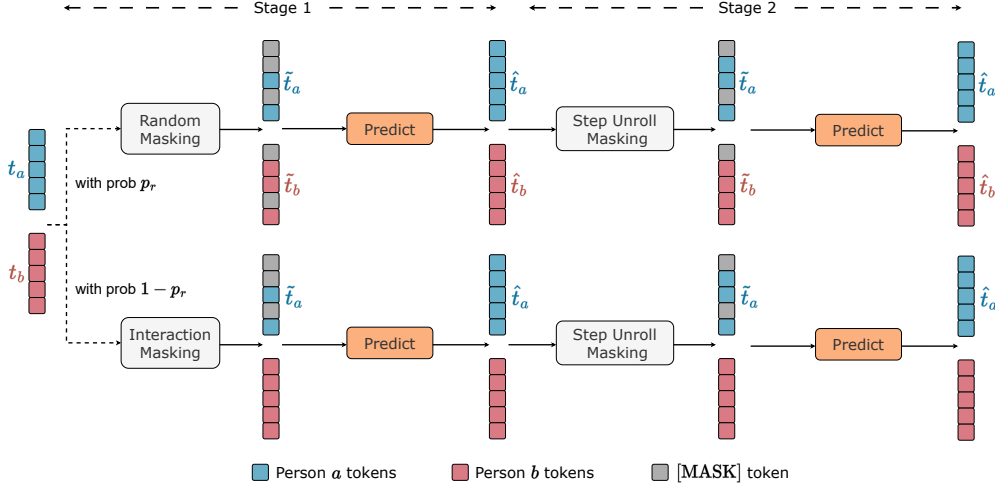


Figure 8: Illustration of the two-stage masking technique used during training of the Inter-M Transformer. For stage 1, we either apply Random Masking with a probability of p_r or Interaction Masking with probability $1 - p_r$. In stage 2, we apply the Step Unroll Masking on the predicted tokens from stage 1.

masking or interaction masking in the first stage. Random masking teaches the model to predict random tokens from both individuals. Whereas Interaction masking, where only one individual’s tokens are masked, promotes learning inter-person dependencies critical for coherent interactions and to improve performance in the reaction generation task. The masking strategy alternates between these two methods based on a probability parameter p_r , with random masking applied p_r of the time and interaction masking $(1 - p_r)$. To evaluate the effect of p_r , we test different values 0.7, 0.8, 0.9 and find that $p_r = 0.8$ offers the best balance between interaction and reaction generation, as shown in Table 4. In the second stage, step unroll masking is applied which retains some of the predicted tokens from stage 1, remasks the remaining tokens and predicts them again. This is employed to incorporate the inference-time progressive refinement of tokens in the training process.

E IMPLEMENTATION DETAILS

Our models are implemented using PyTorch, with details of the model architecture, training, and inference provided below. Key hyperparameters are summarized in the accompanying tables.

E.1 MODEL ARCHITECTURE

The Motion VQ-VAE employs 2D convolutional residual blocks for both the encoder and decoder. The temporal downsampling factor is $n/N = 1/4$ for both datasets, while the spatial downsampling is dataset-specific: $j/J = 5/22$ for InterHuman and $j/J = 5/56$ for InterX. Strided convolutions are used for downsampling in the encoder, while the decoder restores dimensions via upsampling and convolutional layers. The latent representation in VQ-VAE has a dimension $d' = 512$, and the codebook size $|C| = 1024$.

For the Inter-M transformer, we use $\mathbf{L} = 6$ transformer blocks, each with 6 attention heads. The transformer embedding dimension is $\tilde{d} = 384$.

E.2 TRAINING DETAILS

The Motion VQ-VAE is trained for 50 epochs with a batch size of 512. The learning rate is initialized at 0.0002 and decays via a multistep learning rate schedule, reducing by a factor of 0.1 after 70% and 85% of the iterations. A linear warm-up is applied for the first quarter of the iterations. The

Parameter	Value	Description
d'	512	Latent space dimension of VQ-VAE
$ \mathcal{C} $	1024	Codebook size (number of entries)
n/N	1/4	Temporal downsampling factor for both datasets
j/J (InterHuman)	5/22	Spatial downsampling for InterHuman dataset
j/J (InterX)	5/56	Spatial downsampling for InterX dataset
\mathbf{L}	6	Number of transformer blocks
Attention heads	6	Number of attention heads per block
\tilde{d}	384	Transformer embedding dimension
CLIP version	ViT-L/14@336px	Version of CLIP used for text in transformer

Table 5: Motion VQ-VAE and Inter-M Transformer **Model Parameters**

commitment loss factor β is 0.02, and the geometric losses for velocity, foot contact, and bone length are weighted differently across the datasets.

The Inter-M transformer is trained for 500 epochs with a batch size of 52, following a similar multi-step learning rate decay but with a decay factor of 1/3 after 50%, 70%, and 85% of the iterations. The condition drop probability is 0.1 to allow for flexibility in training with or without text conditioning.

Parameter	Value	Description
VQ-VAE batch size	512	Number of samples per batch for VQ-VAE
Transformer batch size	52	Number of samples per batch for transformer
Initial learning rate	0.0002	Starting learning rate for both models
Learning rate decay	0.1 / 1/3	Decay factor for VQ-VAE / Transformer learning rate
β	0.02	Commitment loss factor for VQ-VAE
$\lambda_{vel}, \lambda_{fc}, \lambda_{bl}$ (InterHuman)	100, 500, 5	Geometric loss weights for InterHuman
$\lambda_{vel}, \lambda_{fc}, \lambda_{bl}$ (InterX)	100, 100, 5	Geometric loss weights for InterX dataset
Condition drop prob.	0.1	Drop probability for text conditioning during transformer training
p_r	0.8	Random Masking probability for stage 1 masking during training

Table 6: **Training Hyperparameters** for the Motion VQ-VAE and Inter-M Transformer

E.3 INFERENCE DETAILS

During inference, the number of iterations I is set to 20 for interaction generation and 12 for reaction generation. A classifier-free guidance (CFG) scale of 2 is applied, and the temperature is set to 1 to balance diversity and coherence in the generated results.

Parameter	Interaction Generation	Reaction Generation
Number of iterations	$I = 20$	$I_{react} = 12$
CFG scale		2
Temperature		1

Table 7: **Inference Hyperparameters** for Interaction and Reaction Generation

F DIVERSITY DEMONSTRATION

Our quantitative comparison (section 4.1) shows that InterMask prioritizes adherence to text over extreme diversity, while still being able to generate different distinct interactions for the same text prompt. Here, in Figure 9, we show 2 visual examples to demonstrate this. In both cases, the model remains consistent in generating interactions described in the text prompt while exhibiting distinct features in different samples. For the *dancing* case, first sample shows waving hands in the beginning followed by a synchronized forward step, while individuals face the same direction throughout. The second sample shows individuals facing each other in the beginning, followed by waving hands and

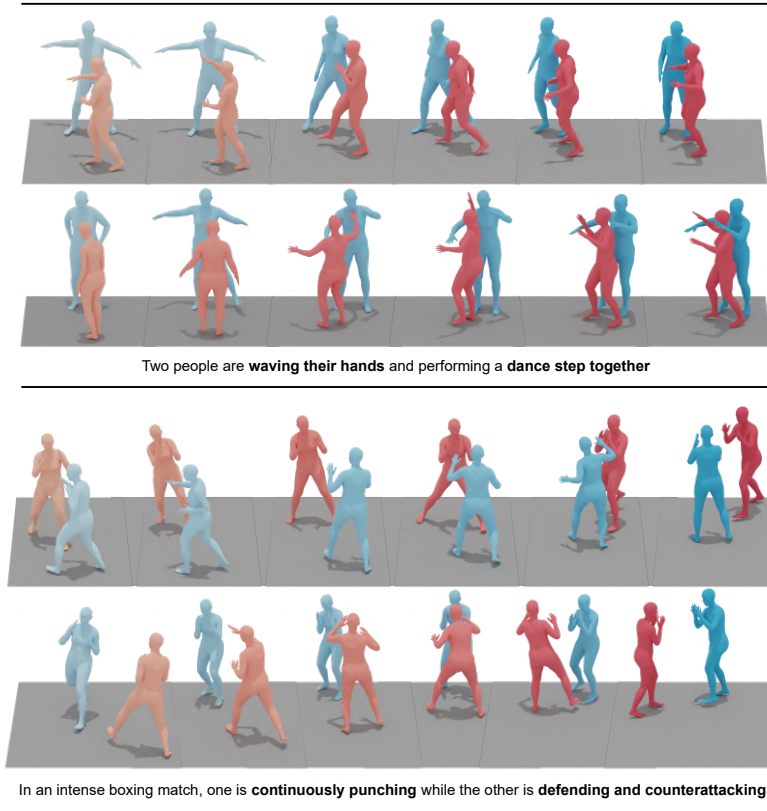


Figure 9: Diversity Demonstration of our method, where it generates two distinct interaction sequences for the same text prompt.

a synchronized spin. For the *boxing* case, first sample shows one individual punching three times and continuously moving forward, while the second sample shows them punching two times and retreating at the end.

G ALTERNATIVE MODELING

In our ablation study (section 4.2), we explore a one-at-a-time modeling framework for interaction generation, referred to as *Alternative Modeling*, which contrasts with the collaborative modeling framework of InterMask. While collaborative modeling predicts the tokens of both individuals simultaneously, alternative modeling follows a sequential process, generating tokens for one individual at a time, conditioned on the thus far predicted tokens of the other.

During training, as shown in Figure 10(a), we randomly mask both individuals’ tokens $\{\tilde{t}_a, \tilde{t}_b\}$ and obtain their embeddings $\{e_a, e_b\}$ through the input process. Then, only the tokens of one individual are predicted (\hat{t}_a) by passing their embeddings through the transformer blocks, conditioned on the embeddings of the other individual using a cross-attention module. During inference (Figure 10(b)), both individuals’ tokens are initially fully masked $\{t_a(0), t_b(0)\}$. In the first iteration, the tokens of one individual are predicted, and these are remasked based on their confidence scores to obtain $t_a(1)$. The second individual’s tokens are then predicted in the next iteration, conditioned on the retained tokens from the first, to obtain $t_b(1)$. This alternation continues iteratively, progressively refining the tokens of both individuals. As shown in Table 3, the FID score for alternative modeling is 7.637 (compared to 5.154 for collaborative modeling), and the R-precision is 0.340 (compared to 0.449). These results indicate that while alternative modeling increases diversity, collaborative modeling produces high-quality and more realistic interactions, offering a better balance between diversity and interaction fidelity.

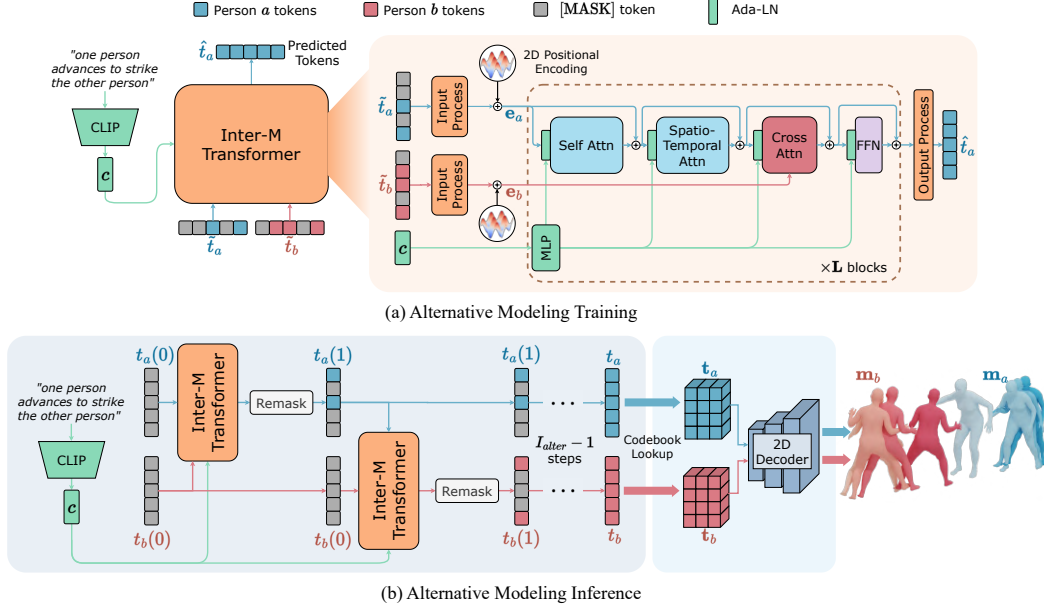


Figure 10: Overview of the **Alternative Modeling** approach, where we predict the tokens of one person at a time. (a) During training, only the embeddings of one individual e_a are updated in the transformer blocks, conditioned on the other individual’s embeddings e_b . (b) During inference, the process alternates between predicting and remasking the tokens of each individual, starting with both fully masked $\{t_a(0), t_b(0)\}$. This process continues for I_{alter} iterations for each individual.

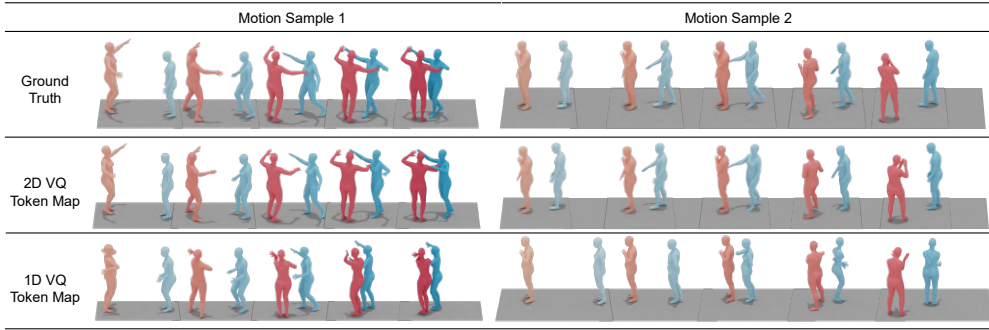


Figure 11: Qualitative results for the ablation study on Motion VQ-VAE to verify the proposed 2D token map.

H ABLATION STUDY QUALITATIVE RESULTS

In this section, we present qualitative results for our ablation studies to complement the quantitative findings discussed in section 4.2. Figure 11 shows two ground truth interaction sequences with their reconstructed samples from the 2D token map VQ-VAE and the baseline 1D token map VQ-VAE. As shown, the 1D VQ-VAE struggles to accurately reconstruct the spatial positions and orientations of the joints for both individuals, leading to incorrect positioning and orientation relative to each other. This results in not only unrealistic interactions but also bizarre individual poses. In contrast, the proposed 2D VQ-VAE provides highly accurate reconstructions at both the individual pose and the collective interaction level.

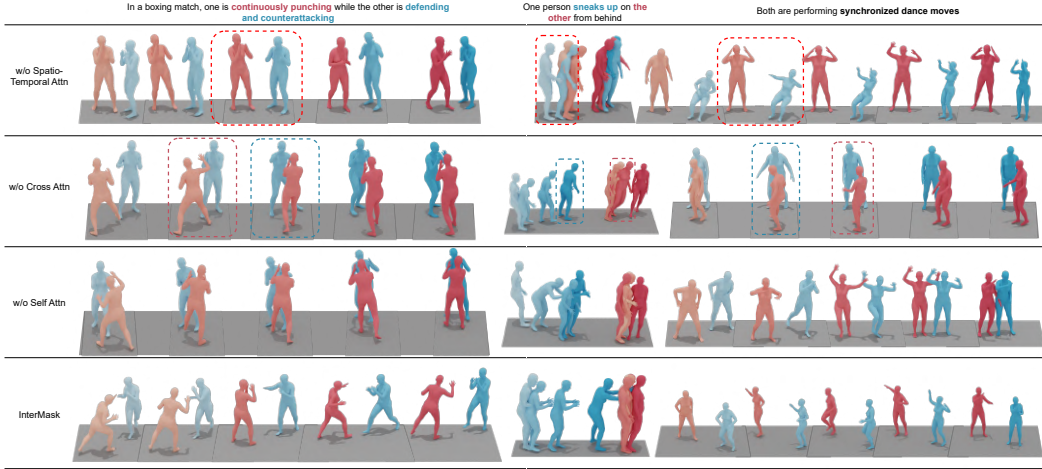


Figure 12: Qualitative results for the ablation study on Inter-M Transformer to verify contributions of the proposed Attention modules.

Figure 12 illustrates the impact of different attention modules in our Inter-M transformer by comparing outputs by removing the spatio-temporal attention, cross-attention, and self-attention modules independently. We provide results for three distinct interaction scenarios: *boxing*, *sneaking*, and *synchronized dancing*. The spatio-temporal attention module emerges as a critical component for generating complex poses and ensuring spatial awareness in interactions. Without this module, the *boxing* scenario exhibits overly simplistic poses, such as timid *punching* and *blocking*, along with the individuals failing to face each other properly. In the *sneaking* scenario, the absence of spatio-temporal attention eliminates the essential spatial progression, as the sneaking individual does not gradually approach the other. Similarly, in the *dancing* scenario, the generated motions are reduced to basic hand raises, and one individual adopts an unnatural crouching pose. By contrast, the inclusion of spatio-temporal attention enables accurate spatial positioning and expressive, synchronized interactions. The cross-attention module seems vital for modeling inter-person dependencies, particularly in achieving accurate reaction timing. Without it, the response motions of the interacting individual appear either delayed or prematurely executed across all examples. For instance, in the *boxing* scenario, the reactive movements fail to synchronize with the initiating individual’s punches. In the *dancing* scenario, the lack of cross-attention results in poor synchronization, disrupting the fluidity of the interaction. Lastly, the self-attention module serves as a refinement mechanism, enhancing the overall quality and coherence of individual motions. Its removal introduces subtle inconsistencies, such as jerky transitions or less fluid movements, which slightly degrade the interaction’s realism. These observations collectively underscore the importance of each attention module in generating realistic, contextually accurate, and expressive interactions.

I REACTION GENERATION INFERENCE

InterMask does not require task-specific fine-tuning or architectural re-design for reaction generation, needing only minor adjustments to the inference process, as illustrated in Figure 13. The process begins by encoding the reference individual’s motion \mathbf{m}_b into tokens t_b using the VQ-VAE encoder. For the other individual, whose reaction is to be generated, we initialize with a fully masked token sequence, $t_a(0)$. Over the course of I_{react} iterations, the transformer progressively predicts and fills in the masked tokens, while the reference tokens remain unmasked throughout. At each iteration i_{react} , the least confident $\gamma\left(\frac{i_{\text{react}}}{I_{\text{react}}}\right) \cdot nj$ tokens are remasked and predicted again, following a cosine scheduling function $\gamma(\cdot)$. Once all tokens are generated, the final token sequence t_a is decoded back into motion \mathbf{m}_a using the VQ-VAE decoder. Since we drop the conditioning signal during some training passes, reaction generation functions effectively both with and without a text description, enabling the model to generate motions based solely on the reference motion or guided by additional textual instructions.

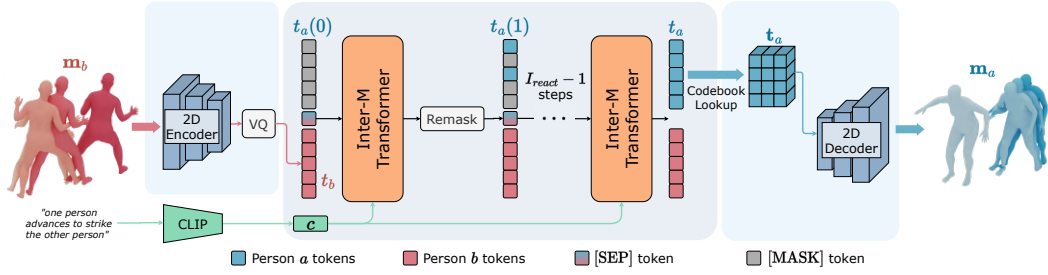


Figure 13: **Inference** process for the **Reaction Generation** task. The motion tokens of the reference individual t_b are obtained from the encoder and kept unmasked throughout. The second individual’s tokens are initially fully masked $t_a(0)$, and are predicted progressively over I_{react} iterations to obtain t_a , which is then decoded using the decoder.

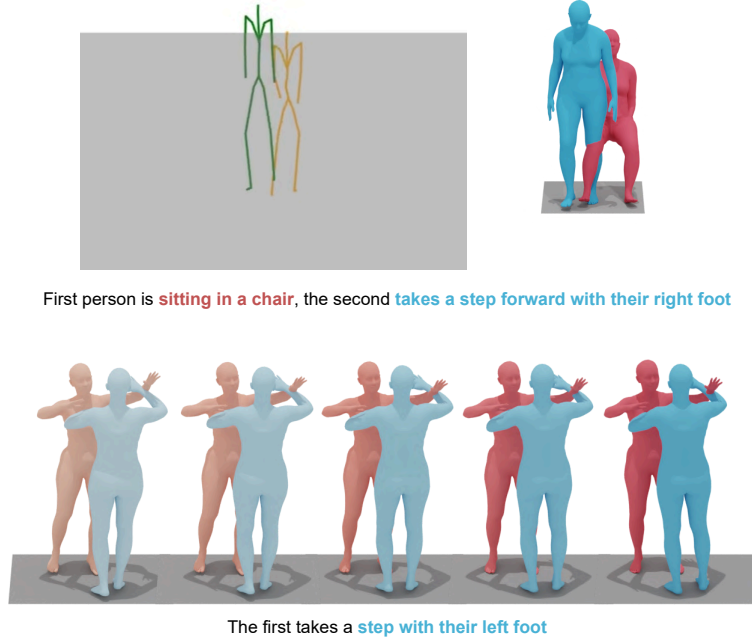


Figure 14: Examples of Limitations of our method. The first row shows body penetration when converted from output skeleton to SMPL mesh. The second row shows implicit bias towards dancing.

J FAILURE CASES

In Figure 14, we show visual examples of failure cases emerging from the limitations of our method, as described in section 5. In the first row, we show that when converting our output skeletons to SMPL (Loper et al., 2015) meshes for visualizations, the results can exhibit body penetration among the interacting individuals. One possible future solution to this problem is to include the mesh conversion in the training process and incorporate anti-penetration in the training loss. In the second row, we show that the model suffers from some implicit biases present in the dataset, where it assumes that the individuals are dancing without explicit mention in the text prompt.

K USER STUDY

We conducted the user study on the Amazon Mechanical Turk platform, where the interface presented to the users is shown in Figure 15. For each sample, users were provided with clear instructions to rate two animations—one generated by InterMask and the other by InterGen (Liang et al., 2024)—from the same text description, with the order of the animations randomized for each sample to avoid bias. Participants were asked to rate both animations on a scale of 1 to 5 for interaction quality and text adherence. Following the individual ratings, they were asked to select the better animation based on their overall impression. A total of 16 users evaluated a total 30 samples, with each sample being rated by three users. To ensure high-quality feedback, we filtered users to include only those with Amazon Mechanical Turk *master* status, with a task approval rating of over 97% and more than 1000 previously approved tasks.

For each text description (provided below), we provide **2 Human Interaction** animation results. Each Animation video is shown from two different point of views for better visibility.

You need to rate both interactions (1 and 2) on their **Quality** and **Adherence to Text**.

- 1) **Interaction Quality** - How natural and realistic are the poses of both people in the interaction? Does the action and its reaction makes sense? - (5: max quality, 1: min quality)

Animation 1:



Animation 2:



- 2) **Adherence to Text Description** - How closely does the interaction follow the provided text description? Do both characters follow the details in the text description? - (5: max adherence, 1: min adherence)

Animation 1:



Animation 2:



- 3) **Preference** - Overall which animation is **better** in your preference based on the above factors

- ☐ Animation 1
☒ Animation 2

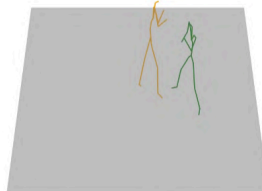
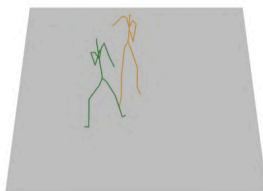
Text Description:

In an intense boxing match, one is continuously punching while the other is defending and counterattacking

Animation 1

Front View

Side View



Animation 2

Front View

Side View

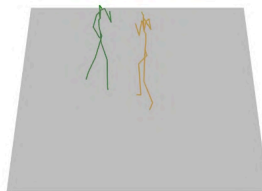
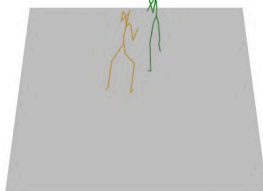


Figure 15: Interface of the **User Study** on Amazon Mechanical Turk.