ReMoS: 3D Motion-Conditioned Reaction Synthesis for Two-Person Interactions

Anindita Ghosh[1,2], Rishabh Dabral[2], Vladislav Golyanik[2], Christian Theobalt[2], and Philipp Slusallek[1,2]

[1] German Research Center for Artificial Intelligence (DFKI)
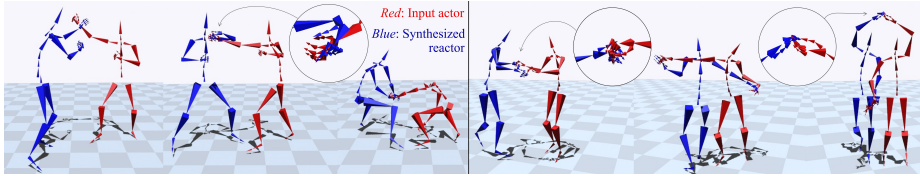[2] Max-Planck Institute for Informatics (MPII)
[3] Saarland Informatics Campus

Abstract. Current approaches for 3D human motion synthesis generate high-quality animations of digital humans performing a wide variety of actions and gestures. However, a notable technological gap exists in addressing the complex dynamics of multi-human interactions within this paradigm. In this work, we present ReMoS, a denoising diffusion-based model that synthesizes full-body *reactive motion* of a person in a two-person interaction scenario. Given the motion of one person, we employ a combined spatio-temporal cross-attention mechanism to synthesize the reactive body and hand motion of the second person, thereby completing the interactions between the two. We demonstrate ReMoS across challenging two-person scenarios such as pair-dancing, Ninjutsu, kickboxing, and acrobatics, where one person's movements have complex and diverse influences on the other. We also contribute the ReMoCap dataset for two-person interactions containing full-body and finger motions. We evaluate ReMoS through multiple quantitative metrics, qualitative visualizations, and a user study, and also indicate usability in interactive motion editing applications. More details are available on the project page: https://vcai.mpi-inf.mpg.de/projects/remos.

Keywords: Reactive Motion Synthesis · Denoising Diffusion Model

1 Introduction

Digital 3D character motion synthesis has emerged as the next frontier for animation pipelines [50], particularly through denoising diffusion probabilistic models (DDPMs) [61]. While methods for generating character motion for various tasks such as text or music conditioned motion synthesis [3,4,18,22,23,34,49,80], face and gesture synthesis [5,6,25,47,48,73], human-scene interaction [19,71,82,84] exist, synthesizing interactions *between* humans is relatively under-explored. Modeling such human-human interactions is essential for designing generative 3D human motion synthesis frameworks supporting the complex physical and social interplay of two interacting persons [66]. It offers new capabilities for character animation tools and software, with applications in commercial and entertainment media [26], interactive mixed and augmented reality [14], and social robotics [76].

Fig. 1: Visualizations of reactive 3D motion sequences synthesized with the proposed ReMoS approach. We synthesize the 3D full-body motion of the reactor (*blue*) conditioned only on the 3D motion of the actor (*red*), thereby completing the interactions between the two (Ninjutsu practice on the *left* and Lindy Hop dancing on the *right*). The synthesized hand interactions are enlarged and highlighted with *circles*.

Toward this goal, we focus on the task of modeling reactive motions. We aim to automatically generate realistic, temporally-aligned reactive motions of a responding person given the continuous motion of a guiding person (see Fig. 1). This alleviates a common overhead for animators, enabling them to design an *acting* character and automatically obtain meaningful motions for a *reacting* character. However, automating such a process poses two key challenges. First, two-person interaction synthesis increases the dimensionality of the already challenging problem of single-person motion synthesis. In our case, the generated reactive motions must align with the conditional signals provided in the form of 3D motion sequences from the actor. Second, doing so in a generative setting without any cues from text prompts or action labels (to obviate additional supervision and data needs) requires a careful trade-off between generating diverse motions and adhering to the narrow manifold of plausible reactions.

While existing methods for synthesizing interactions [10, 20] and two-person motions [42, 54, 65] are good at generating plausible motions, they rely on additional annotations, such as action labels or text prompts, to specify the motion. Action labels typically depict actions at a high level, such as "high-five" or "salsa", but do not capture the fine-grained synchronization with the actor's motions. Textual descriptions offer nuance for the motions, but collecting accurate textual annotations is considerably cumbersome, particularly for contact-heavy or fast-paced two-person motions, such as dancing and exercising. Further, performing generative tasks through textual prompting remains challenging for non-experts [79]. Apart from the data, contact-based motion generation also needs to consider hand-based interactions to improve realism. This entails the need for frameworks capable of synthesizing hand movements with higher degrees of freedom, which are challenging to model but crucial for improving motion realism. Balancing focus between the broader *full-body reactive movements* and the finer *hand motions* involves addressing motion at two significantly different scales — a challenge exacerbated by the high degrees of freedom of the skeletons.

With this objective in mind, we present ReMoS, a novel approach for Reactive Motion Synthesis with full-body articulations. Inspired by the advancements of DDPMs in 3D human motion synthesis [50, 54, 65], we develop a DDPM framework (Sec. 3.1) with a cascaded, two-stage generation strategy tailored to our

problem setting (Fig. 2). In the first stage of our diffusion model, we generate the reactive motion for the reactor's body joints conditioned on the actor's body joints. In the second stage, we use the synthesized body motions as an additional parameter to generate masking for appropriate hand motions (Sec. 3.2). We propose a combined spatio-temporal cross-attention (CoST-XA) mapping between the actors' and the reactors' body motion embeddings, that learns the inter-dependencies in their motions without needing additional annotations. Further, to synthesize hand interactions, we introduce a hand-interaction-aware cross-attention (H-XA) mechanism to ensure the relevant hand joints react to the actors' motions, thus allowing the network to distill localized hand interaction features. We ensure accurate coordination between the actor and the reactor by using contact-based reaction loss (Sec. 3.3) and an inference-time guidance function (Sec. 3.4) that improves the plausibility of the body and hand interactions. To explore reaction synthesis in complex and diverse two-person scenarios, we contribute the ReMoCap dataset consisting of full-body and finger motion sequences for fast-paced swing dance of Lindy Hop [63], and the martial art technique of Ninjutsu [12] (Sec. 4). In summary, our technical contributions are:

- **ReMoS.** A novel method for reactive 3D human motion synthesis using a cascaded diffusion framework to generate full-body and hand motions. Our framework generates fine-grained reactive motions for complex and dynamic contact-based interactions and derives the reactions directly from the actor's motions, without requiring explicit label or text annotations.
- **Interaction-Based Attentions.** A combined spatio-temporal cross attention (CoST-XA) mechanism to enforce coherence between the body movements of the actor and the reactor, and a hand-interaction-aware cross-attention (H-XA) mechanism to enforce the appropriate hand-based interactions between the two characters.
- **ReMoCap.** A new dataset for two-person interactions under complex scenarios of Lindy Hop dancing and Ninjutsu. The dataset consists of $\sim 275.7K$ motion frames with multiview RGB videos and 3D full-body motion capture of the two interacting persons. It further includes finger-level articulations.
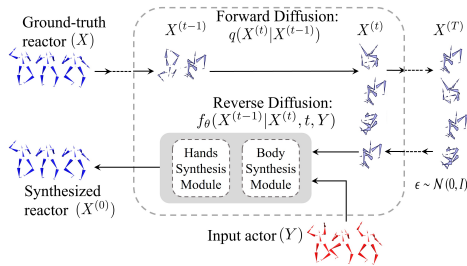
We evaluate our approach in different scenarios, including Lindy Hop dancing, Ninjutsu, Acrobatics [24] and Kickboxing [55], and report state-of-the-art performance (Sec. 5.3). We also report a user study on comparing the visual quality of ReMoS compared to the ground truth and the baseline methods (Sec. 5.4).

2   Related Work

Multi-Person 3D Motion Synthesis. Synthesizing close interactions between two or multiple persons is a challenging task in animation. Early works in multi-person interaction are based on motion graphs [59], interaction patches [58], momentum-based inverse kinematics and motion blending [37], and topologically-based pose representations [28, 29], to name a few. The increasing availability of interaction datasets [16, 39, 43, 53, 55, 78] has led to a rise in data-driven approaches for synthesizing digital partners or opponents in multi-person inter-

action settings. Mousas [46] uses a hidden Markov model to control the dance motions of a digital partner in an immersive setup. Ahuja et al. [1] introduce a residual-attention model to generate body pose in a conversation setting conditioned on two audio signals and the opposite person's body pose. Starke et al. [64] propose a phase function-based network that learns asynchronous movements of each bone and its interaction with external objects. Guo et al. [24] propose the Extreme Pose Interaction dataset with a two-stream network with cross-interaction attention modules for forecasting pose sequences of two interacting characters. GAN-based models [20,45] generate motions for an interacting person conditioned on an input character and class labels depicting the type of reaction performed. These methods leverage the daily interaction datasets such as SBU Kinect [78] and 2C [55]. InterFormer [11] uses an interaction transformer with both spatial and temporal attention to generate reactive motions given some initial seed poses of both characters. It can synthesize sparse-level interactions based on the motions of the K3HI [32] and the DuetDance [39] datasets. Concurrent to our work, Duolando [60] uses an off-policy reinforcement learning model to predict tokenized motion for a leader and follower conditioned on music. The aforementioned methods either depend on seed motion as input or require additional conditions to drive the motion. In contrast, ReMoS focuses on synthesizing well-synchronized full-body and hand motions for the reactor conditioned only on the actor's 3D motions, without using any additional labels or prompts.

Diffusion Based 3D Motion Synthesis. Denoising diffusion models [61] have recently demonstrated their high potential in generative human motion modeling, specifically in single-person conditional motion synthesis. Conditional single-person motion generation has been performed using diffusion-based approaches for co-speech gestures [2, 86], audio-driven motion [13, 68, 85], and text-driven motion [54,67,77,81] synthesis. Diffusion-based techniques have also been extensively used for conditional synthesis for human-object interactions [38, 40, 41, 74], hand-object interactions [44, 75], and human-scene interactions [33]. Guided motion diffusion models such as GMD [35] and TraceAndPace [52] incorporate spatial constraints on motion trajectories, to guide the motion towards a goal at inference time. OmniControl [72] proposes spatial and realism guidance to control any joint for diffusion-based human motion generation. For two-person interaction synthesis, RAIG [65] proposes a diffusion-based, role-aware approach for two-person interactions, given separate textual descriptions for each person. BiGraphDiff [10] uses graph transformer denoising diffusion to learn two-person interaction conditioned on action labels. ComMDM [54] uses a communication block between two MDMs [67] to coordinate two-person interaction generation. ContactGen [21] presents a diffusion-based contact prediction module that adaptively estimates potential contact regions between two humans according to the interaction label. Liang et al. [42] propose InterGen, a diffusion-based model to synthesize two-person interactions given text descriptions as input conditions. They also propose the InterHuman dataset that includes diverse two-person interaction scenarios with rich text annotations. Inspired by these recent approaches, we design our proposed method ReMoS as a DDPM-based

Fig. 2: ReMoS Overview. Given the motion of the actor (*bottom-middle*, in *red*), we synthesize a plausible motion for the reactor (*bottom-left*, in *blue*). We achieve this using a denoising diffusion-based probabilistic model (*center*) trained on reactive motion sequences (*top-left*, in *blue*).

3D motion-conditioned reaction synthesis method, consisting of a cascaded diffusion model with a combined spatio-temporal cross-attention mechanism. This allows ReMoS to implicitly learn the fine-grained synchronization between two interacting persons from only one of the person's motions, without any need for additional prompts or labels, and synthesize the corresponding motions of the second person. ReMoS further synthesizes plausible hand motions for the reactor to incorporate realistic hand-based interactions.

3   Reactive Motion Synthesis

We consider a digital, two-person interaction setting where the motions of one character, the *actor*, are known, and we need to synthesize the motions of the other character, the *reactor*. ReMoS aims to synthesize the synchronized reactive motions of the reactor in such a setting. We denote the reactor's motion as $X = \{X_B, X_H\}$, where $X_B \in \mathbb{R}^{N \times J_B \times 3}$ denotes the 3D full-body joint positions with J_B body joints, $X_H \in \mathbb{R}^{N \times J_H \times 3}$ denotes the 3D finger joint positions with J_H finger joints across both hands, and N denotes the number of frames in the motion sequence. Likewise, we denote the actor's motion as $Y = \{Y_B, Y_H\}$. Our goal is to model the conditional probability distribution $P(X|Y)$ from which we can sample plausible reactive motions. We model this distribution using conditional denoising diffusion probabilistic models (DDPMs), owing to their unique strengths in capturing temporal coherence of motion data and handling complex motion dynamics [13, 67, 68]. We elaborate our methodology below.

3.1   DDPM for Reactive Motion Synthesis

The diffusion process consists of two steps: forward or *destructive* diffusion, and reverse or *denoising* diffusion. In the forward process, we progressively corrupt a clean reactive motion sequence X by adding Gaussian noise ϵ to it for T steps. With sufficiently small noise and large T, we can get $X^{(T)} \sim \mathcal{N}(0, I)$ following closed-form formulation of Ho et al. [30]:

$$X^{(t)} = \sqrt{\bar{\alpha}_t} X^{(0)} + \sqrt{1 - \bar{\alpha}_t} \epsilon, \qquad \epsilon \sim \mathcal{N}(0, I), \tag{1}$$

where $\bar{\alpha}_t$ controls the rate of diffusion and $t \in [0, T]$. Reversing this diffusion process allows for sampling novel motion sequences from a multivariate Gaussian

distribution $p\left(X^{(T)}\right) \sim \mathcal{N}\left(0, I\right)$ as

$$p\left(X^{(0)}\right) = p\left(X^{(T)}\right) \prod_{t=1}^{T} p\left(X^{(t-1)}|X^{(t)}\right). \tag{2}$$

We approximate the computationally intractable term $p\left(X^{(t-1)}|X^{(t)}\right)$ with a learnable function $f_\theta\left(X^{(t)}, t\right)$, and optimize θ to encode the space of human reactive motions. Our reactive motion X is also conditioned on the actor's motion Y. Therefore, we modify the learnable function as $X^{(t-1)} = f_\theta\left(X^{(t)}, t, Y\right)$. Following recent works [51,67], we estimate the original motion $X^{(0)} = f_\theta\left(X^{(t)}, t, Y\right)$ from our diffusion model by iterating through all t denoising steps during inference.

3.2 ReMoS Framework

To generate fine-grained reactions with appropriate hand motions, ReMoS decodes the reactive motion of X in a cascaded fashion (Fig. 3). It first estimates the full-body joints X_B, and then the hand joints X_H as
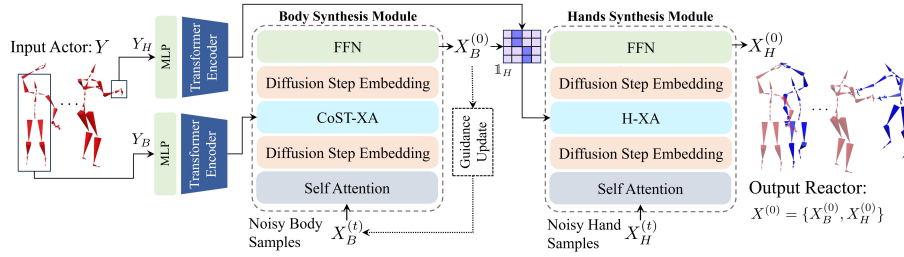
$$X_B^{(0)} = f_{\theta_B}\left(X_B^{(t)}, t, Y_B\right), \tag{3}$$

$$X_H^{(0)} = f_{\theta_H}\left(X_H^{(t)}, t, Y_H, \mathbb{1}_{H_A}\left(Y_B\right) \mathbb{1}_{H_R}\left(X_B^{(0)}\right)\right), \quad \text{and} \tag{4}$$

$$X^{(0)} = \left\{X_B^{(0)}, X_H^{(0)}\right\}, \tag{5}$$

where $\mathbb{1}_{H_A}$ and $\mathbb{1}_{H_R}$ are binary, spatio-temporal hand-interaction mask functions that determine which hand joints of the reactor and actor interact at any given frame. Our cascaded framework comes from the observation that full-body articulations occur at significantly different scales than hand articulations. Since the body motions affect the hand articulations, our cascaded framework feeds the generated body motions as an additional condition to the hand generation module (Fig. 3). We also train the body and the hand generation modules separately, with additional hand-interaction-aware attention mapping for the hand joints. Previous work [19] used such disjoint training strategy on conditional motion synthesis for single persons. To better accommodate for two-person interactions in our case, we benefit from using a cascaded diffusion strategy [31].

Body Synthesis Module. In the first stage of our cascaded framework, we diffuse the reactor's body motion, $X_B^{(t)}$, at each diffusion step t, and feed it into a transformer decoder [69] block. To condition the reactor's body motion on the actor's body motion Y_B, we introduce a combined spatio-temporal cross-attention (CoST-XA) mapping. CoST-XA combines the spatial and temporal features of the actor's and the reactor's motions through an attention matrix to simultaneously learn the inter-dependencies between each pair of reactor and actor body joints $(x_{n,j}, y_{n,j})$, $x_{n,j} \in X_B^{(t)}$ and $y_{n,j} \in Y_B$, respectively, at each frame. Denoting the query from the reactor's motion features as $Q_B \in \mathbb{R}^{NJ_B \times 3}$

Fig. 3: ReMoS Framework. Given the full-body sequence of the actor (*left*, in *red*), we input noisy body and hand samples (*from below*) in a cascaded fashion. We synthesize the body samples first, and use them for hand-interaction-aware attention masking (*top-center*) to synthesize the denoised hand samples (*top-right*). The full-body reactive motion is a concatenation of the denoised body and hand samples (*right*, in *blue*).

and the key-value pair from the actor's motion features as $K_B \in \mathbb{R}^{NJ_B \times 3}$ and $V_B \in \mathbb{R}^{NJ_B \times 3}$, respectively, we formulate an attention matrix of dimensions $NJ_B \times NJ_B$. In contrast, most previous approaches [42,65] use cross-attention only on the temporal features of the motion using an attention matrix of shape $N \times N$. InterFormer [11] sequentially uses spatial and temporal cross-attention modules instead of combining them, which may result in a partial loss of information on fine-grained, inter-person interactions across time, especially in the absence of additional annotations (prompts or action labels) to condition the reactions. Our proposed cross-attention module, which we define as

$$\text{CoST-XA} = \text{softmax}\left(\frac{Q_B K_B^T}{\sqrt{d_{K_B}}}\right) V_B, \tag{6}$$

crucially considers combinations of spatial and temporal interaction features between different body segments of the actor and the reactor to efficiently synchronize their motions. We simultaneously project the diffusion timestep t at each denoising step into the transformer block after the attention blocks and use a final layer of fully-connected network with SiLU activations [15] and batch normalization [7] to generate the body motions. The first stage output is a vector $X_B^{(0)} \in \mathbb{R}^{N \times J_B \times 3}$ representing the reactor's clean, synthesized body motion.

Hands Synthesis Module. The second stage of our cascaded framework synthesizes the reactor's hand joints. We input noisy samples $X_H^{(t)}$ into a similar transformer decoder block conditioned on the actor's hand motions Y_H. Here, we require the attention weights to be high for the hands used in the hand-based interactions and low for the passive hands. We explicitly enforce this using a spatio-temporal hand-interaction-aware cross-attention mapping (H-XA) to encourage learning localized interaction features. For H-XA to work, we introduce binary hand-interaction masks $\mathbb{1}_{H_A}$ and $\mathbb{1}_{H_R} \in \mathbb{R}^{N \times J_H}$. Its entries are calculated by thresholding the distances of the actor's and reactor's wrist joints in Y_B and

$X_B^{(0)}$, respectively, to the nearest body joints of the other person. The active entries of $\mathbb{1}_{H_A}$ and $\mathbb{1}_{H_R}$ determine which of the actor's and reactor's hands are sufficiently close to the body of the other person (and hence interacting). We then project the encoded Y_H features into each decoder layer using H-XA, as

$$\text{H-XA} = \text{softmax}\left(\frac{(\mathbb{1}_{H_R} \odot Q_H)(\mathbb{1}_{H_A} \odot K_H)^T}{\sqrt{d_{K_H}}}\right) V_H, \tag{7}$$

where K_H and V_H are the hand motion features from the actor, Q_H are the hand motion features from the reactor with d_{K_H} channels, and $\odot$ denotes element-wise product for masking the query and key values. We project the diffusion timestep t at each denoising step into the transformer block and use fully-connected networks with SiLU activations to generate the clean, synthesized reactor hand motions $X_H^{(0)} \in \mathbb{R}^{N \times J_H \times 3}$. At the end, to obtain the full reactive motion $X^{(0)}$, we concatenate $\left\{X_B^{(0)}, X_H^{(0)}\right\}$ as in Eqn. 5.

3.3   Losses and Training Details

We train ReMoS to minimize a weighted sum of three loss terms, the reconstruction loss $\mathcal{L}_c$, the reaction loss $\mathcal{L}_r$, and the kinematic loss $\mathcal{L}_k$, as

$$\mathcal{L} = \lambda_c \mathcal{L}_c + \lambda_r \mathcal{L}_r + \lambda_k \mathcal{L}_k, \tag{8}$$

where λ_c, λ_r and λ_k are scalar weights to balance the individual losses.

Reconstruction Loss $\mathcal{L}_c$. This is the standard diffusion data term that minimizes the ℓ_2-distance between the ground truth and the synthesized reactive motion, as $\mathcal{L}_c = \left\|X - X^{(0)}\right\|_2$. While the reconstruction loss provides the vital data term to drive the training, it does not enforce interaction synchronization between the actor and the reactor, or any kinematic constraints on the motion.
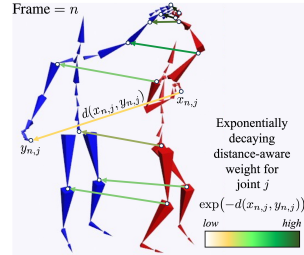
Reaction loss $\mathcal{L}_r$. We introduce the reaction loss to ensure accurate timing and spatial positioning of the reactor's motion with respect to the actor's motion. We calculate the ground-truth Euclidean distance $d(\cdot, \cdot)$ between each joint of the actor and the reactor at each frame and minimize the deviations from these distances for the actor and the synthesized reactor's motions. We note that this distance term also implicitly constrains the reactor's joints in frames that are less relevant for interactions, *i.e.*, where the reactor's joints are far from the actor's joints. Further, the Euclidean distances for their hand joints in frames with no hand-based contact become drastically high. To mitigate these concerns, we use an exponentially decaying distance-aware weight $\exp\left(-d\left(x_{n,j}, y_{n,j}\right)\right)$ at each frame n to focus more on the reactor's joints that are closer to the actor, and therefore more relevant for interaction (see Fig. 4). Thus, we get our reaction loss term as

$$\mathcal{L}_r = \frac{1}{NJ} \sum_{n=1}^{N} \sum_{j=1}^{J} \exp\left(-d\left(x_{n,j}, y_{n,j}\right)\right) \cdot \left|d\left(x_{n,j}, y_{n,j}\right) - d\left(x_{n,j}^{(0)}, y_{n,j}\right)\right|. \tag{9}$$

Table 1: Dataset Comparisons. Comparing ReMoCap with existing multi-person interaction datasets.

Dataset (Chronologically)	Motion Frames	Duration (hours)	Multi-view RGB Videos	Finger Articulation
SBU [78]	$\sim$ 7K	0.13	✗	✗
NTU-26 [43]	$\sim$ 22K	0.47	✓	✗
2C [55]	$\sim$ 13K	0.06	✗	✗
DanceDB [53]	$\sim$ 1.44M	4.00	✗	✗
DuetDance [39]	$\sim$ 196K	1.09	✗	✗
CHI3D [16]	$\sim$ 486K	2.70	✓	✗
ExPI [24]	$\sim$ 30K	0.33	✓	✗
InterHuman [42]	$\sim$ 107M	6.56	✗	✗
DD100 [60]	$\sim$ 200K	1.92	✗	✓
ReMoCap (ours)	$\sim$ 275.7K	2.04	✓	✓

Fig. 4: Visualization of Distance Aware Reaction Loss. We use an exponentially decaying distance-aware reaction loss to focus more on the reactor's joints that are closer to the actor.

Kinematic Loss $\mathcal{L}_k$. To maintain the kinematic plausibility of the generated motions, we follow existing literature on regularizers for bone length consistency, foot contacts, and temporal consistency [56,57,68]. Our kinematic loss term is a weighted sum of the joint velocity loss $\mathcal{L}_{vel}$, the joint acceleration loss $\mathcal{L}_{acc}$, the bone length consistency loss $\mathcal{L}_{bone}$, and the foot sliding loss $\mathcal{L}_{foot}$, as

$$\mathcal{L}_k = \lambda_v \mathcal{L}_{vel} + \lambda_a \mathcal{L}_{acc} + \lambda_b \mathcal{L}_{bone} + \lambda_f \mathcal{L}_{foot}, \tag{10}$$

where λ_v, λ_a, λ_b and λ_f are scalar weights (more details in the appendix).

3.4 Inference Time Spatial Guidance

While our method synthesizes plausible reactive motions, it can sometimes spatially misalign the reactor's body to the actor's, especially at the arm joints, for fast-paced, contact-heavy interactions. This, in turn, affects finger-joint synthesis in the second stage of our cascaded diffusion. To improve spatial alignment, we leverage guidance functions [35, 52] that can provide gradients to nudge the sampling process towards a certain direction. We design a guidance function $G\left(\phi, \hat{\phi}\right) \in \mathbb{R}^{N \times J_A \times 3}$, which minimizes the distance between the J_A arm joints of the actor (ϕ) and the J_A arm joints of the synthesized reactor ($\hat{\phi}$). Specifically, we re-apply our interaction masks $\mathbb{1}_{H_A}$ and $\mathbb{1}_{H_R}$ to determine which of the reactor's hands are more likely to be interacting with the actor, and minimize the distances between the arm joints of the corresponding sides as $G = \arg\min_{\hat{\phi}} \left(\left\| \mathbb{1}_{H_A} \odot \phi - \mathbb{1}_{H_R} \odot \hat{\phi} \right\| \right)$. We then plug G into the denoising pipeline of our body diffusion module, as

$$X_B^{(0)} = X_B^{(0)} - \gamma \nabla_{X_B^{(0)}} G\left(\phi, \hat{\phi}\right), \qquad \gamma = \text{ guidance scale.} \tag{11}$$

4   ReMoCap Dataset

We propose the ReMoCap dataset to facilitate research on contact-based two-person interactions with finger-level articulations. It contains complex two-person interactions in two separate scenarios: the fast-paced, swing-style Lindy Hop dancing, and the martial arts of Ninjutsu. We consider these two types of motions for studying physical interactions for two main reasons. First, these motions encompass a significantly *diverse* spectrum of motions in terms of intensity, speed, and style. Second, choreographed motion generation is developing as an active field of research and we envision this dataset to facilitate research in interactive applications such as performing in virtual reality [9] and remote tutoring [53].

Data Collection. We invited 4 trained Lindy Hop dancers and 5 trained Ninjutsu artists to perform diverse interactions in a multi-view capture studio. We tracked their motions using a commercially available markerless, multi-view motion-capture system [8], which tracks 93 degrees-of-freedom driving 69 body joints. Capturing motions in a markerless setting enables the performers to move uninhibited, while also making the data suitable for training monocular or multi-view motion capture methods under severe inter-person occlusions. We also capture RGB videos from 120 camera views for each sequence. The dataset includes 3D skeleton poses with full-body and finger annotations, foreground-background segmentation masks, and 3D surface reconstructions of the subjects. We capture sequences of multiple lengths totaling ∼275.7K frames (2.04 hours) of motion data from each view at a frame rate of 50 fps for the Lindy Hop and 25 fps for the Ninjutsu. Out of all the frames, around 150K frames have hand-based interactions between the two characters where the closest distance between the finger joints of the actor and the reactor is within 50 mm. We present more dataset statistics in the appendix.

Dataset Comparison. Table 1 shows a comprehensive comparison of ReMo-Cap with existing multi-person interaction datasets. Existing datasets consisting of two-person interactions, such as SBU [78], K3HI [32], NTU-26 [43] and 2C [55], are limited in size and motion capture quality. They typically feature simple actions, such as handshakes, punching, pushing and kicking, with weak interactions, and do not capture hand motions. The recent ExPI [24] dataset captures Lindy Hop aerial sequences to model more complex interactions, and the Inter-Human [42] dataset features both daily motions (*e.g.*, passing objects, greeting, communicating) and professional motions (*e.g.*, Taekwondo, Latin dance, boxing). However, these datasets do not provide finger-level motion capture data, which is a key requirement to intricately model inter-human activities. Two-person dance datasets, such as DuetDance [39] and DanceDB [53], also lack the hand motion data needed for modeling interactions. Only the concurrent work, Duolando [60]* proposes the DD100 dataset with strong interactions between two dancers and provides hand motion data.

---

*unpublished at the time of our paper submission

5   Experiments and Results

We conduct comprehensive experiments to evaluate ReMoS on multiple two-person datasets covering a wide range of interaction scenarios. We perform training and evaluation on multiple large-scale datasets, including 2C [55], ExPI [24], and our proposed ReMoCap. We provide the implementation details, report quantitative and qualitative comparisons on standard evaluation metrics and through a user study, report ablations, and show how to apply ReMoS as a motion editing tool. We provide details on the dataset preparation in the appendix.

5.1   Implementation Details

As a pre-processing step, we normalize the two-person body poses by translating the actor's motion Y and the reactor's motion X together, such that the root joint of Y is at the global origin for all N frames. We then compute the relative 3D joint coordinates of X_B and Y_B w.r.t. the root joint of Y_B. For each hand, we compute the relative 3D joint coordinates of X_H and Y_H w.r.t. the corresponding wrist joints of X_H and Y_H. Using normalized inter-person, root-relative joint positions benefits the stability and convergence of our model. ReMoS uses $T = 500$ diffusion steps where $\bar{\alpha}$ changes linearly from 0.0002 to 0.02. We train for about 64K iterations on both sequences of ReMoCap using Adam [36] with a base learning rate of 10^{-5} and a batch size of 64. We decay the learning rate using a Step LR scheduler with a step size of 5 epochs and a decay rate of 0.99. We use $d = 256$ for our latent embedding representation and use 6 layers in our transformer decoder with 4 heads for calculating the attention. The training takes around 8 and 11 hours on an NVIDIA RTX A4000 GPU for Lindy Hop and Ninjutsu, respectively. The inference time is $\sim24s$ to generate 50 frames of full-body and finger motions ($12.5s$ for body synthesis and then $11.5s$ for hand synthesis). We set $\lambda_c, \lambda_r, \lambda_v = 10.0$, $\lambda_a, \lambda_k, \lambda_b = 1.0$ and $\lambda_f = 20.0$ (only applied after 100 training epochs) as the loss term weights in Eqns. 8 and 10. We set the guidance scale $\gamma = 10^{-3}$ in Eqn. 11.

5.2   Baselines and Ablated Versions

For baselines, we choose the closest motion synthesis methods in a two-person setting, namely, MixNMatch [20], InterFormer [11], ComMDM [54], Role-Aware Interaction Generation (RAIG) [65], and InterGen [42]. InterFormer [11] was originally trained in a reactive motion synthesis setting without additional annotations, such as action labels or text descriptions, and we maintain this setup. For the other methods, we mask out their input text/label embeddings to comply with our annotation-free setting. We re-train these methods on ReMoCap with a thorough hyper-parameter search and report their best performances. We provide more training details in the appendix.

   We also compare our proposed ReMoS model with five of its ablated versions:
- **w/o diffusion.** Training ReMoS using a transformer encoder-decoder network without using DDPM strategy.

Table 2: Quantitative Evaluation on ReMoCap. We compare ReMoS with our baselines and ablated versions (Sec. 5.2) on the ReMoCap dataset. We evaluate these methods on metrics such as MPJPE, MPJVE, FID, and Diversity. ↓: lower is better, ↑: higher is better, →: values closer to GT are better. **Bold** indicates best.

Method	Lindy Hop					Ninjutsu				
	MPJPE (mm) ↓	MPJVE (mm) ↓	FID ↓ (body)	FID ↓ (hands)	Div →	MPJPE (mm) ↓	MPJVE (mm) ↓	FID ↓ (body)	FID ↓ (hands)	Div →
GT	-	-	-	-	7.57	-	-	-	-	10.51
InterFormer [11]	66.6	8.26	0.53	0.65	4.54	270.2	3.4	0.57	0.68	6.48
MixNMatch [20]	70.2	10.3	0.77	0.78	2.48	257.2	5.2	0.74	0.72	4.83
ComMDM [54]	59.4	4.41	0.32	0.53	7.48	201.2	4.1	0.34	0.58	9.98
RAIG [65]	71.2	4.32	0.47	0.63	8.45	199.1	5.1	0.21	0.63	10.11
InterGen [42]	62.6	3.92	0.30	0.61	7.21	172.6	3.9	0.32	0.57	9.98
ReMoS (ours)	**40.7**	**2.26**	**0.12**	**0.26**	**7.62**	**139.2**	**3.3**	**0.16**	**0.35**	**10.26**
w/o diffusion	72.5	4.91	0.58	0.74	4.04	224.5	4.1	0.52	0.64	6.06
w/o cascading	63.9	4.95	0.51	0.55	7.12	223.6	4.2	0.42	0.75	6.62
w/o CoST-XA	44.2	3.62	0.21	0.39	7.45	176.6	3.6	0.27	0.41	8.91
w/o reaction loss	44.6	3.51	0.22	0.38	7.31	144.6	3.7	0.23	0.39	8.99
w/o spatial guidance	41.9	2.34	**0.12**	**0.26**	**7.62**	139.4	3.4	**0.16**	**0.35**	10.26

- **w/o cascading.** Training ReMoS with an integrated transformer for all the joints instead of a cascaded strategy for body and hand generation.
- **w/o CoST-XA.** Using a spatial followed by a temporal cross-attention mapping instead of the combined spatio-temporal features for body synthesis.
- **w/o reaction loss.** Removing the reaction loss (Eqn. 9) in training.
- **w/o spatial guidance.** Removing spatial guidance G_A (Sec. 3.4) in inference.

5.3 Quantitative Evaluation

We evaluate ReMoS on standard evaluation metrics. We measure the temporal consistency deviation from the ground truth 3D pose following [57] to report the mean per-joint positional error (MPJPE) and the mean per-frame, per-joint velocity error (MPJVE), both in mm, on the synthesized motions. We measure the Fréchet Inception Distance (FID) [27] to compare the distribution gap between the embedding spaces of the generated and ground-truth motions. As part of our method focuses on generating hand motions, we measure the FID score of body and hands separately for ReMoCap. We also compute the latent variance of the generated motions (Diversity) [67, 81]. Table 2 reports the quantitative evaluation of ReMoS with its baselines and ablations on ReMoCap. ReMoS achieves state-of-the-art performance for both the Lindy Hop pair-dance setting and the Ninjutsu setting. We observe that methods using denoising diffusion have higher Diversity compared to transformer-based [11] or GAN-based [20] methods. This enforces the variability claims of denoising diffusion models. We note at least 20% improvement in the MPJPE and around 40% improvement in the FID score for the reactor's body motion when using the proposed CoST-XA mechanism, confirming its benefit in learning fine-grained, inter-person dependencies across time. We also note an almost 50% improvement in the FID scores of

Table 3: Quantitative Evaluation on the ExPI and 2C datasets. We compare ReMoS with state-of-the-art motion synthesis methods on the ExPI [24] and 2C datasets [55]. ↓: lower is better, ↑: higher is better, →: values closer to GT are better. **Bold** indicates best.

Method	ExPI				2C			
	MPJPE (mm) ↓	MPJVE (mm) ↓	FID ↓ (body)	Div →	MPJPE (mm) ↓	MPJVE (mm) ↓	FID ↓ (body)	Div →
GT	-	-	-	2.01	-	-	-	2.22
InterFormer [11]	99.1	3.56	0.42	1.31	90.7	5.11	0.52	1.45
MixNMatch [20]	122.4	5.56	0.48	1.18	62.4	6.01	0.47	1.24
ComMDM [54]	121.4	5.41	0.45	2.48	69.9	3.34	0.49	2.86
RAIG [65]	131.2	3.96	0.53	2.51	91.6	4.92	0.67	4.45
InterGen [42]	100.6	3.91	0.43	2.09	67.6	4.01	0.47	2.91
ReMoS (ours)	**97.9**	**3.52**	**0.41**	**1.98**	**59.1**	**3.33**	**0.34**	**2.07**

Table 4: User Study Results. Mean scores on a five-point Likert scale (scores $1 - 5$).

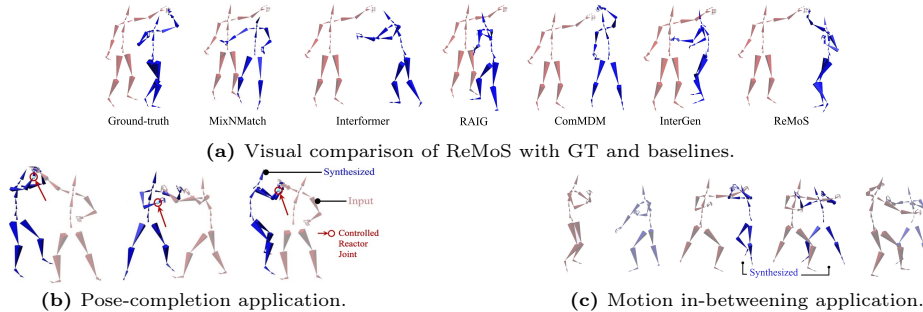| Method	Motion Quality ↑	Reaction Plausibility ↑
GT	4.86 ± 0.54	4.72 ± 0.56
InterFormer	2.52 ± 0.61	2.28 ± 0.57
MixNMatch	1.92 ± 0.71	2.18 ± 0.57
ComMDM	3.02 ± 0.47	3.12 ± 0.52
RAIG	2.83 ± 0.67	2.48 ± 0.65
InterGen	3.18 ± 0.57	3.19 ± 0.53
ReMoS	$\mathbf{3.79 \pm 0.55}$	$\mathbf{3.88 \pm 0.54}$

the reactor's hand motion when using the cascaded strategy and the reaction loss. Further, the spatial guidance function fine-tuning improves MPJPE and MPJVE. The MPJPE values are overall higher in Ninjutsu than in Lindy Hop as the trajectory of the reactor is more diverse for Ninjutsu. We also report the evaluation of ReMoS with its baselines on the ExPI [24] and the 2C [55] datasets in Table 3. These datasets do not provide hand motions, so we only evaluate the reactor's body motions, and report state-of-the-art performance of ReMoS.

5.4   User Study

We evaluate the visual quality of our captured ground truth data and the generated reactive motions through a user study. We show participants 26 different interaction sequences across the ground truth, our method, and its baselines. For each sequence, we randomly show them three methods side-by-side and ask them to rate the 3D motions they observe in terms of *(a)* the reactor's motion quality, *irrespective* of the actor's motion (Motion Quality), and *(b)* the plausibility of the reaction *given* the actor's motion (Reaction Plausibility). We ask the participants to rank these motions from '1' (worst score) to '5' (best score) in a five-point Likert scale. Table 4 reports the mean scores from the responses of 40 participants, excluding the responses that did not pass our validation checks. We notice the ground truth motions in ReMoCap achieve the highest score of around 96%, indicating that participants perceive our captured ground truth motions to be natural-looking, with realistic interactions. ReMoS has the second best ranking of around 78%, which is almost 21% higher than the baselines, showing that it is preferred more over the baselines.

5.5   Qualitative Results and Applications

Fig. 1 shows qualitative results of ReMoS on ReMoCap and highlights the synthesized hand interactions. We note that our actor and reactor are *interchangeable*, *i.e.*, depending on the character driving the interaction at a given time,

(a) Visual comparison of ReMoS with GT and baselines.

(b) Pose-completion application.   **(c)** Motion in-betweening application.

Fig. 5: Qualitative Results and Applications. We show some visual results and the application of ReMoS as a motion editing tool. **(a)** The reactor (in *blue*) synthesized by ReMoS has the most plausible alignment with the actor (in *red*) compared to the baselines. **(b)** We manually control the right-hand wrist joint of the reactor and let ReMoS synthesize the remaining body joints conditioned on the actor. **(c)** ReMoS synthesizes the reactor's motion in-between the start and end frames.

we can swap the actor and the reactor to produce the relevant reactive motions. Fig. 5a shows a visual comparison of ReMoS with the baselines for one frame of Lindy Hop motion. ReMoS synthesizes reaction with the most plausible alignment with the actor's motion. We provide detailed visual results in our supplementary video. Inspired by the applicability of diffusion-based methods for motion editing and controlling, we demonstrate how ReMoS can also be used as an interactive motion editing tool to control the reactor's motion as desired. In Figs. 5b and 5c, we show results from two different motion editing applications, namely, *pose completion with controlled joints* and *motion in-betweening*. We discuss more details in the appendix.

6   Conclusion

ReMoS brings 3D motion conditioned reaction synthesis to a qualitatively new level by generating diverse, well-synchronized reactions for complex movements and plausible hand motions for contact-based interactions. It outperforms the existing baselines both quantitatively and in a user study. We also highlight some practical applications of our model, such as pose completion and motion in-betweening, which can lead to the development of useful generative assistants for animators, designers, and creative artists. Even though we utilize joint positions as parameterization due to their ready availability, we can adapt our approach to accommodate mesh-level inter-person contacts by introducing an offset to the contact threshold between the two bodies, thus simulating skin-to-skin interactions. We believe there is scope for improvement in the inference speed, where approaches such as Pro-DDPM [17] and DDIM [62] have shown success. Future directions also involve scaling the problem towards multi-person motion prediction [70] and considering scene-aware interactions for the characters, which would further enhance immersive user experiences.

Acknowledgements

References

1. Ahuja, C., Ma, S., Morency, L.P., Sheikh, Y.: To react or not to react: End-to-end visual pose forecasting for personalized avatar during dyadic conversations. In: 2019 International conference on multimodal interaction (2019)
2. Ao, T., Zhang, Z., Liu, L.: GestureDiffuCLIP: Gesture diffusion model with clip latents. SIGGRAPH (2023)
3. Aristidou, A., Yiannakidis, A., Aberman, K., Cohen-Or, D., Shamir, A., Chrysanthou, Y.: Rhythm is a dancer: Music-driven motion synthesis with global structure. IEEE Transactions on Visualization and Computer Graphics (TVCG) (2022)
4. Athanasiou, N., Petrovich, M., Black, M.J., Varol, G.: Teach: Temporal action composition for 3d humans. In: 2022 International Conference on 3D Vision (3DV) (2022)
5. Bhattacharya, U., Childs, E., Rewkowski, N., Manocha, D.: Speech2affectivegestures: Synthesizing co-speech gestures with generative adversarial affective expression learning. In: Proceedings of the 29th ACM International Conference on Multimedia (2021)
6. Bhattacharya, U., Rewkowski, N., Banerjee, A., Guhan, P., Bera, A., Manocha, D.: Text2gestures: A transformer-based network for generating emotive body gestures for virtual agents. In: IEEE Conference on Virtual Reality and 3D User Interfaces (IEEE VR) (2021)
7. Bjorck, N., Gomes, C.P., Selman, B., Weinberger, K.Q.: Understanding batch normalization. In: Advances in Neural Information Processing Systems (2018)
8. https://captury.com (2023)
9. Chan, J.C., Leung, H., Tang, J.K., Komura, T.: A virtual reality dance training system using motion capture technology. IEEE transactions on learning technologies **4**(2) (2010)
10. Chopin, B., Tang, H., Daoudi, M.: Bipartite graph diffusion model for human interaction generation. In: Winter Conference on Applications of Computer Vision (WACV) (2024)
11. Chopin, B., Tang, H., Otberdout, N., Daoudi, M., Sebe, N.: Interaction transformer for human reaction generation. IEEE Transactions on Multimedia (2023)
12. Cummins, A.: In search of the ninja: the historical truth of ninjutsu. The History Press (2012)
13. Dabral, R., Mughal, M.H., Golyanik, V., Theobalt, C.: Mofusion: A framework for denoising-diffusion-based motion synthesis. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
14. Egges, A., Papagiannakis, G., Magnenat-Thalmann, N.: Presence and interaction in mixed reality environments. The Visual Computer (2007)

15. Elfwing, S., Uchibe, E., Doya, K.: Sigmoid-weighted linear units for neural network function approximation in reinforcement learning. Neural Networks (2018)
16. Fieraru, M., Zanfir, M., Oneata, E., Popa, A.I., Olaru, V., Sminchisescu, C.: Three-dimensional reconstruction of human interactions. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
17. Gandikota, R., Brown, N.: Pro-ddpm: Progressive growing of variable denoising diffusion probabilistic models for faster convergence. In: 33rd British Machine Vision Conference 2022, BMVC (2022)
18. Ghosh, A., Cheema, N., Oguz, C., Theobalt, C., Slusallek, P.: Text-based motion synthesis with a hierarchical two-stream rnn. In: ACM SIGGRAPH 2021 Posters, pp. 1–2 (2021)
19. Ghosh, A., Dabral, R., Golyanik, V., Theobalt, C., Slusallek, P.: IMoS: Intent-driven full-body motion synthesis for human-object interactions. In: Computer Graphics Forum. vol. 42. Wiley Online Library (2023)
20. Goel, A., Men, Q., Ho, E.S.L.: Interaction Mix and Match: Synthesizing Close Interaction using Conditional Hierarchical GAN with Multi-Hot Class Embedding. Computer Graphics Forum (2022)
21. Gu, D., Shim, J., Jang, J., Kang, C., Joo, K.: Contactgen: Contact-guided interactive 3d human generation for partners. AAAI (2024)
22. Guo, C., Zou, S., Zuo, X., Wang, S., Ji, W., Li, X., Cheng, L.: Generating diverse and natural 3d human motions from text. In: Conference on Computer Vision and Pattern Recognition (2022)
23. Guo, C., Zuo, X., Wang, S., Cheng, L.: Tm2t: Stochastic and tokenized modeling for the reciprocal generation of 3d human motions and texts. In: European Conference on Computer Vision (2022)
24. Guo, W., Bie, X., Alameda-Pineda, X., Moreno-Noguer, F.: Multi-person extreme motion prediction. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2022)
25. Habibie, I., Elgharib, M., Sarkar, K., Abdullah, A., Nyatsanga, S., Neff, M., Theobalt, C.: A motion matching-based framework for controllable gesture synthesis from speech. In: ACM SIGGRAPH Conference Proceedings (2022)
26. Hanser, E., Mc Kevitt, P., Lunney, T., Condell, J.: Scenemaker: Intelligent multi-modal visualisation of natural language scripts. In: Irish Conference on Artificial Intelligence and Cognitive Science. Springer (2009)
27. Heusel, M., Ramsauer, H., Unterthiner, T., Nessler, B., Hochreiter, S.: Gans trained by a two time-scale update rule converge to a local nash equilibrium. Advances in neural information processing systems **30** (2017)
28. Ho, E.S., Komura, T.: Planning tangling motions for humanoids. In: IEEE-RAS International Conference on Humanoid Robots (2007)
29. Ho, E.S., Komura, T.: Character motion synthesis by topology coordinates. In: Computer graphics forum. Wiley Online Library (2009)
30. Ho, J., Jain, A., Abbeel, P.: Denoising diffusion probabilistic models. Advances in Neural Information Processing Systems **33** (2020)
31. Ho, J., Saharia, C., Chan, W., Fleet, D.J., Norouzi, M., Salimans, T.: Cascaded diffusion models for high fidelity image generation. The Journal of Machine Learning Research **23**(1) (2022)
32. Hu, T., Zhu, X., Guo, W.: Two-person interaction recognition based on key poses. Journal of Computational Information Systems (2014)
33. Huang, S., Wang, Z., Li, P., Jia, B., Liu, T., Zhu, Y., Liang, W., Zhu, S.C.: Diffusion-based generation, optimization, and planning in 3d scenes. In: Proceed-

ings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023)

34. Huang, Y., Zhang, J., Liu, S., Bao, Q., Zeng, D., Chen, Z., Liu, W.: Genre-conditioned long-term 3d dance generation driven by music. In: International Conference on Acoustics, Speech and Signal Processing (ICASSP). IEEE (2022)

35. Karunratanakul, K., Preechakul, K., Suwajanakorn, S., Tang, S.: Guided motion diffusion for controllable human motion synthesis. In: International Conference on Computer Vision (ICCV) (2023)

36. Kingma, D.P., Ba, J.: Adam: A method for stochastic optimization. arXiv preprint arXiv:1412.6980 (2014)

37. Komura, T., Ho, E.S., Lau, R.W.: Animating reactive motion using momentum-based inverse kinematics. Computer Animation and Virtual Worlds (2005)

38. Kulkarni, N., Rempe, D., Genova, K., Kundu, A., Johnson, J., Fouhey, D., Guibas, L.: Nifty: Neural object interaction fields for guided human motion synthesis. arXiv preprint arXiv:2307.07511 (2023)

39. Kundu, J.N., Buckchash, H., Mandikal, P., Jamkhandi, A., Radhakrishnan, V.B., et al.: Cross-conditioned recurrent networks for long-term synthesis of inter-person human motion interactions. In: Winter Conference on Applications of Computer Vision (WACV) (2020)

40. Li, J., Clegg, A., Mottaghi, R., Wu, J., Puig, X., Liu, C.K.: Controllable human-object interaction synthesis. arXiv preprint arXiv:2312.03913 (2023)

41. Li, J., Wu, J., Liu, C.K.: Object motion guided human motion synthesis. ACM Transactions on Graphics (TOG) (2023)

42. Liang, H., Zhang, W., Li, W., Yu, J., Xu, L.: InterGen: Diffusion-based multi-human motion generation under complex interactions. International Journal for Computer Vision (IJCV) (2024)

43. Liu, J., Shahroudy, A., Perez, M., Wang, G., Duan, L.Y., Kot, A.C.: NTU RGB+D 120: A large-scale benchmark for 3d human activity understanding. IEEE transactions on pattern analysis and machine intelligence (2019)

44. Liu, X., Yi, L.: Geneoh diffusion: Towards generalizable hand-object interaction denoising via denoising diffusion. In: International Conference on Learning Representations (ICLR) (2024)

45. Men, Q., Shum, H.P., Ho, E.S., Leung, H.: Gan-based reactive motion synthesis with class-aware discriminators for human–human interaction. Computers & Graphics (2022)

46. Mousas, C.: Performance-driven dance motion control of a virtual partner character. In: IEEE Conference on Virtual Reality and 3D User Interfaces (VR) (2018)

47. Mughal, M.H., Dabral, R., Habibie, I., Donatelli, L., Habermann, M., Theobalt, C.: Convofusion: Multi-modal conversational diffusion for co-speech gesture synthesis. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2024)

48. Ng, E., Joo, H., Hu, L., Li, H., Darrell, T., Kanazawa, A., Ginosar, S.: Learning to listen: Modeling non-deterministic dyadic facial motion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2022)

49. Petrovich, M., Black, M.J., Varol, G.: TEMOS: Generating diverse human motions from textual descriptions. In: European Conference on Computer Vision (2022)

50. Po, R., Yifan, W., Golyanik, V., Aberman, K., Barron, J.T., Amit H. Bermano, E.R.C., Dekel, T., Holynski, A., Kanazawa, A., Liu, C.K., Liu, L., Mildenhall, B., Nießner, M., Ommer, B., Theobalt, C., Wonka, P., Wetzstein, G.: State of the art on diffusion models for visual computing. arXiv pre-prints (2023)

51. Ramesh, A., Dhariwal, P., Nichol, A., Chu, C., Chen, M.: Hierarchical text-conditional image generation with clip latents. In: International Conference on Virtual Reality (2022)
52. Rempe, D., Luo, Z., Bin Peng, X., Yuan, Y., Kitani, K., Kreis, K., Fidler, S., Litany, O.: Trace and pace: Controllable pedestrian animation via guided trajectory diffusion. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
53. Senecal, S., Nijdam, N.A., Aristidou, A., Magnenat-Thalmann, N.: Salsa dance learning evaluation and motion analysis in gamified virtual reality environment. Multimedia Tools and Applications (2020)
54. Shafir, Y., Tevet, G., Kapon, R., Bermano, A.H.: Human motion diffusion as a generative prior. In: International Conference on Learning Representations (ICLR) (2024)
55. Shen, Y., Yang, L., Ho, E.S.L., Shum, H.P.H.: Interaction-based human activity comparison. IEEE Transactions on Visualization and Computer Graphics (2020)
56. Shimada, S., Golyanik, V., Xu, W., Pérez, P., Theobalt, C.: Neural monocular 3d human motion capture with physical awareness. ACM Transactions on Graphics (ToG) (2021)
57. Shimada, S., Golyanik, V., Xu, W., Theobalt, C.: Physcap: Physically plausible monocular 3d motion capture in real time. ACM Transactions on Graphics (ToG) (2020)
58. Shum, H.P., Komura, T., Shiraishi, M., Yamazaki, S.: Interaction patches for multi-character animation. ACM transactions on graphics (TOG) **27**(5) (2008)
59. Shum, H.P., Komura, T., Yamazaki, S.: Simulating competitive interactions using singly captured motions. In: Proceedings of ACM symposium on Virtual reality software and technology (2007)
60. Siyao, L., Gu, T., Yang, Z., Lin, Z., Liu, Z., Ding, H., Yang, L., Loy, C.C.: Duolando: Follower gpt with off-policy reinforcement learning for dance accompaniment. In: International Conference on Learning Representations (ICLR) (2024)
61. Sohl-Dickstein, J., Weiss, E., Maheswaranathan, N., Ganguli, S.: Deep unsupervised learning using nonequilibrium thermodynamics. In: International Conference on Machine Learning (ICML) (2015)
62. Song, J., Meng, C., Ermon, S.: Denoising diffusion implicit models. arXiv preprint arXiv:2010.02502 (2020)
63. Spring, H.: Swing and the lindy hop: dance, venue, media, and tradition. American Music (1997)
64. Starke, S., Zhao, Y., Komura, T., Zaman, K.: Local motion phases for learning multi-contact character movements. ACM Transactions on Graphics (TOG) (2020)
65. Tanaka, M., Fujiwara, K.: Role-aware interaction generation from textual description. In: International Conference on Computer Vision (ICCV) (2023)
66. Tanke, J., Zhang, L., Zhao, A., Tang, C., Cai, Y., Wang, L., Wu, P.C., Gall, J., Keskin, C.: Social diffusion: Long-term multiple human motion anticipation. In: Proceedings of the IEEE/CVF International Conference on Computer Vision (2023)
67. Tevet, G., Raab, S., Gordon, B., Shafir, Y., Bermano, A.H., Cohen-Or, D.: Human motion diffusion model. arXiv preprint arXiv:2209.14916 (2022)
68. Tseng, J., Castellon, R., Liu, K.: Edge: Editable dance generation from music. In: Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
69. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. Advances in neural information processing systems (2017)

70. Wang, J., Xu, H., Narasimhan, M., Wang, X.: Multi-person 3d motion prediction with multi-range transformers. Advances in Neural Information Processing Systems (2021)
71. Wang, Z., Chen, Y., Jia, B., Li, P., Zhang, J., Zhang, J., Liu, T., Zhu, Y., Liang, W., Huang, S.: Move as you say, interact as you can: Language-guided human motion generation with scene affordance. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR) (2024)
72. Xie, Y., Jampani, V., Zhong, L., Sun, D., Jiang, H.: OmniControl: Control any joint at any time for human motion generation. In: International Conference on Learning Representations (ICLR) (2024)
73. Xing, J., Xia, M., Zhang, Y., Cun, X., Wang, J., Wong, T.T.: Codetalker: Speech-driven 3d facial animation with discrete motion prior. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023)
74. Xu, S., Li, Z., Wang, Y.X., Gui, L.Y.: InterDiff: Generating 3d human-object inter-actions with physics-informed diffusion. In: International Conference on Computer Vision (ICCV) (2023)
75. Ye, Y., Li, X., Gupta, A., De Mello, S., Birchfield, S., Song, J., Tulsiani, S., Liu, S.: Affordance diffusion: Synthesizing hand-object interactions. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023)
76. Yoon, Y., Ko, W.R., Jang, M., Lee, J., Kim, J., Lee, G.: Robots learn social skills: End-to-end learning of co-speech gesture generation for humanoid robots. In: International Conference on Robotics and Automation (ICRA). IEEE (2019)
77. Yuan, Y., Song, J., Iqbal, U., Vahdat, A., Kautz, J.: Physdiff: Physics-guided hu-man motion diffusion model. In: Proceedings of the IEEE/CVF International Con-ference on Computer Vision (2023)
78. Yun, K., Honorio, J., Chattopadhyay, D., Berg, T.L., Samaras, D.: Two-person interaction detection using body-pose features and multiple instance learning. In: IEEE computer society conference on computer vision and pattern recognition workshops (2012)
79. Zamfirescu-Pereira, J., Wong, R.Y., Hartmann, B., Yang, Q.: Why johnny can't prompt: How non-ai experts try (and fail) to design LLM prompts. In: Proceedings of Conference on Human Factors in Computing Systems (CHI) (2023)
80. Zhang, J., Zhang, Y., Cun, X., Zhang, Y., Zhao, H., Lu, H., Shen, X., Shan, Y.: T2m-gpt: Generating human motion from textual descriptions with discrete representations. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023)
81. Zhang, M., Cai, Z., Pan, L., Hong, F., Guo, X., Yang, L., Liu, Z.: Motiondif-fuse: Text-driven human motion generation with diffusion model. arXiv preprint arXiv:2208.15001 (2022)
82. Zhang, W., Dabral, R., Leimkühler, T., Golyanik, V., Habermann, M., Theobalt, C.: ROAM: Robust and object-aware motion generation using neural pose descrip-tors. In: International Conference on 3D Vision (3DV) (2024)
83. Zhang, W., Liu, Z., Zhou, L., Leung, H., Chan, A.B.: Martial arts, dancing and sports dataset: A challenging stereo and multi-view dataset for 3d human pose estimation. Image and Vision Computing **61** (2017)
84. Zhang, X., Bhatnagar, B.L., Starke, S., Guzov, V., Pons-Moll, G.: Couch: Towards controllable human-chair interactions. In: European Conference on Computer Vi-sion (2022)
85. Zhou, Z., Wang, B.: Ude: A unified driving engine for human motion generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023)

86. Zhu, L., Liu, X., Liu, X., Qian, R., Liu, Z., Yu, L.: Taming diffusion models for audio-driven co-speech gesture generation. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (2023)

ReMoS: 3D Motion-Conditioned Reaction Synthesis for Two-Person Interactions −Appendix−

Anindita Ghosh[1,2,3], Rishabh Dabral[2,3], Vladislav Golyanik[2,3], Christian Theobalt[2,3], and Philipp Slusallek[1,3]

[1] German Research Center for Artificial Intelligence (DFKI)
[2] Max-Planck Institute for Informatics (MPII)
[3] Saarland Informatics Campus

We provide additional details on the loss functions used for training ReMoS, more statistics on the ReMoCap dataset, and describe how the datasets and baselines are prepared for evaluation. We also show some additional results.

A   Additional Details of Loss Functions

Kinematic Loss Terms. We describe the details of the velocity, acceleration, bone length and foot sliding losses loss terms from Eqn. 10 in the main paper. To improve the temporal consistency of the motion [68], we minimize the joint velocities and joint accelerations between two consecutive frames of the ground-truth reactive motions, X, and the synthesized reactive motions, $\hat{X}$, defined as

$$\mathcal{L}_{vel} = \frac{1}{N-1} \sum_{n=0}^{N-1} \left\| \left(X^{n+1} - X^n \right) - \left(\hat{X}^{n+1} - \hat{X}^n \right) \right\|_2^2, \tag{A.1}$$

$$\mathcal{L}_{acc} = \frac{1}{N-2} \sum_{n=0}^{N-2} \left\| \left(X^{n+2} - 2X^{n+1} + X^n \right) - \left(\hat{X}^{n+2} - 2\hat{X}^{n+1} + \hat{X}^n \right) \right\|_2^2, \tag{A.2}$$

where N is the total number of frames.

Additionally, we introduce a bone length consistency loss, $\mathcal{L}_{bone}$, to ensure that the synthesized reactor joint positions satisfy the skeleton consistency [42]. We define this loss as

$$\mathcal{L}_{bone} = \left\| \mathbf{B}\left(X \right) - \mathbf{B}\left(\hat{X} \right) \right\|_2^2, \tag{A.3}$$

where $\mathbf{B}$ represents the bone lengths in a pre-defined human body kinematic tree.

Further, foot sliding is a common artifact in motion synthesis [56, 57]. We constrain this by ensuring that the toe joint in contact with the ground plane has zero velocity. We use a binary foot contact loss [67, 68] on the foot joints of the synthesized pose to ensure that the output motion does not slide across the ground plane, defined as

$$\mathcal{L}_{foot} = \frac{1}{N-1} \sum_{n=0}^{N-1} \left\| \left(\hat{X}^{n+1} - \hat{X}^n \right) \cdot \hat{\mathbb{1}}_{foot}^n \right\|_2^2, \tag{A.4}$$

where $\hat{\mathbb{1}}_{foot}^{n} \in \{0, 1\}$ is the foot-ground contact indicator for the synthesized reactive motion $\hat{X}^{n}$ at each frame n.

B   ReMoCap Dataset Analysis

Our proposed ReMoCap dataset covers two types of motion, namely the Lindy Hop dance and the martial art technique of Ninjutsu (see Sec. 4 in the main paper).

Lindy Hop motion capture. The Lindy Hop part of the dataset consists of 8 dance sequences captured at 50 fps, each around 7.5 minutes long, resulting in around 174.2K motion frames. We had 4 trained dancers, 2 males (denoted A and B) and 2 females (denoted C and D), participate in the Lindy Hop motion capture. We pair the dancers as (A, C), (B, D), (A, D), and (B, C). Of these pairings, (A, D) contains dance sequences not performed by the other three pairs (in terms of twists and maneuvers). We also capture multiview RGB videos at 50 fps from 116 camera views for each sequence, which can benefit two-person pose reconstruction work in the future. We show samples from these videos in Fig. B.1. From these sequences, almost 145.2K frames have a hand-in-hand contact between the two dancers with a contact threshold of 50 mm between the finger joints of the two dancers. By increasing the contact threshold to 100 mm, the number of frames where the two dancers have contact increases to 147.5K.
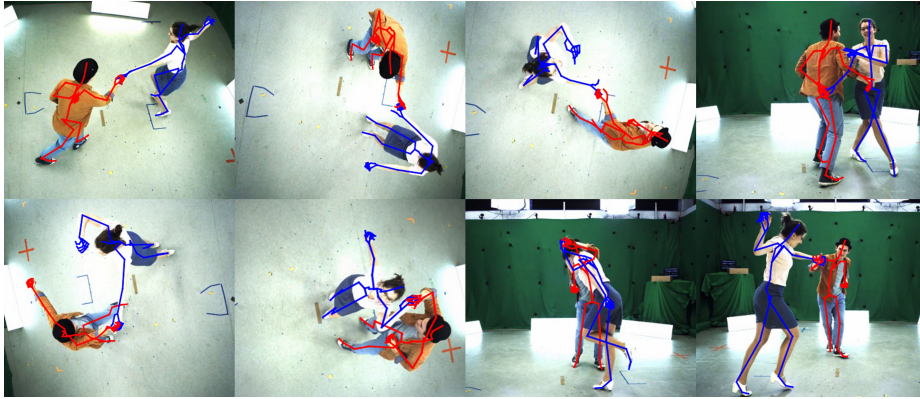
Ninjutsu motion capture. The Ninjutsu part of the dataset consists of 79 sequences each captured at 25 fps. The sequences vary in length with a total number of around 99.8K motion frames resulting in around 66.5 minutes of motion. We had 5 trained, male Ninjutsu artists participate in the Ninjutsu motion capture. We pair them in all possible combinations and ask them to perform different variations of motion. Along with the 3D pose, we also capture multiview RGB videos at 25 fps using 116 cameras. We show samples from these videos in Fig. B.2. From these sequences, almost 81K frames have contact-based interactions between the two performers, where the closest distance between any joints is 50 mm.

C   Dataset and Baseline Preparation

We discuss the preparation of the different datasets and the baseline methods for our evaluation purposes.

C.1   Dataset Preparation

Preparing the Lindy Hop data in ReMoCap. We split the dataset such that motions captured from dancer pairs (A, C), (B, D), and (B, C) are in our training set, and motions captured from pair (A, D) are in our test set. We

Fig. B.1:   Samples from the Lindy Hop motion capture for the ReMo-Cap dataset. We show multi-view RGB samples with corresponding 3D poses from our Lindy Hop motion capture performed by trained dancers. Lindy-hop requires coordination between the two dancers, while also allowing individual dancers the freedom to perform their own motions. This makes it suitable for testing our reactive motion synthesis approach.

downsample each motion sequence to 20 fps and filter the frames where the dancing partners have hand-to-hand contact between the actors' and reactors' finger joints. We represent each character using 27 body joints and 22 hand joints. We convert the 3D joint angle representations into joint positions using forward kinematics and then convert them to root local representations, as explained in Sec. 5.1 in the main paper. For training, we use a sequence length of 20 frames.

Preparing the Ninjutsu data in ReMoCap. We divide the whole dataset into roughly 3 : 1 train-test ratio and take 28 sequences of diverse attacking and maneuvering motions for testing, and the rest for training. We downsample each motion sequence to 10 fps, and filter out the frames where the pairs are more than 1 meter of each other. We represent each character using 27 body joints and 22 hand joints. We convert the 3D joint angle representations into joint positions using forward kinematics and then convert them to root local representations, as explained in Sec. 5.1 in the main paper. For training, we use a sequence length of 50 frames.

Preparing the Extreme Pose Interaction (ExPI) Dataset [24]. The ExPI dataset consists of 2 pairs of professionals performing acrobatics and Lindy Hop aerial sequences. It consists of 16 different acrobatic actions. Each couple consists of a leader and a follower. We aim to synthesize the motions of the followers as they react to the leaders' movements. We use the *common action split* proposed by the original authors [24], and split the dataset into train and test sets such that all the actions performed by (A, B) are in the train set and all the actions performed by (C, D) are in the test set. We represent each subject

Fig. B.2:   Samples from the Ninjutsu motion capture for the ReMo-Cap dataset. We show multi-view RGB samples with corresponding 3D poses from the Ninjutsu motion capture performed by trained artists. In contrast to existing martial arts datasets [55, 83], we include finger joint motion capture and moves of varying interaction complexity.

using 16 joints (omitting the *'lhead'* and *'rhead'* joints) and convert the global 3D joint positions given in the dataset to root relative joint representations, as explained in Sec. 5.1 in the main paper. Since the ExPI dataset does not have hand motions, we only train with body motions and forego the hand diffusion stage. We train ReMoS for about 20K iterations on the ExPI dataset using the Adam optimizer [36], with a base learning rate of 10^{-5} and a batch size of 32.

Preparing the Character-Character Dataset (2C) [55]. The 2C dataset consists of full-body motions of kickboxing actions performed by pairs of participants. The interactions include motions such as *kicking* and *punching*, with diverse reactions such as *avoiding* and *being hit*. We use the pose sequence of the leading character, who throws the punches and kicks, as the acting sequence for our model. We aim to synthesize the full body motion of the reacting character, who is blocking or avoiding the moves, as our output. Following the split of MixNMatch [20], we use a roughly 3 : 1 train-test ratio to train our method. Each character contains 25 joints and we convert the 3D joint angle representations into joint positions using forward kinematics and then convert them to root relative joint position representations, as explained in Sec. 5.1 in the main paper. Since the 2C dataset does not have hand motions, we only train with body motions and forego the hand diffusion stage. We train ReMoS for about 25K iterations on the 2C dataset using the Adam optimizer [36], with a base learning rate of 10^{-5} and a batch size of 16.

Preparing the InterHuman Dataset [42]. We report additional results on the InterHuman dataset in this appendix. It consists of human-human interactions for daily motions, such as passing objects, greeting, and communicating,

and professional activities, such as, Taekwondo, Latin dance, and boxing. It consists of a total of $7,779$ motions with 22 joints per person. We randomly select the pose sequence of one of the characters as the acting sequence for each motion to train our model. We aim to synthesize the full body motion of the corresponding other character in each motion as our output. We follow the split of InterGen [42] for our experiments. Since the InterHuman dataset does not have hand motions, we only train with body motions and forego the hand diffusion stage. We train ReMoS for about 45K iterations on the InterHuman dataset using the Adam optimizer [36], with a base learning rate of 10^{-5} and a batch size of 64.

C.2   Baseline Preparation

As we mention in Sec. 5.2 in the main paper, we use InterFormer [11], MixN-Match [20], ComMDM [54], RAIG [65] and InterGen [42] as baselines. We describe how we use each of these methods in our setting.

InterFormer [11]. InterFormer consists of a transformer network with temporal and spatial attentions. It takes an input acting sequence Y and encodes it with spatial and temporal self-attention. It also needs the initial pose of the reactor X and predicts the subsequent frames of the reactor in an autoregressive manner. It uses information from skeletal adjacency matrices and an interaction distance module that provides information on the interactions. We use the normalization technique mentioned in Sec. 5.1 in the main paper to normalize the actor's and the reactor's body poses. We train InterFormer on an NVIDIA RTX A4000 GPU for about 20K iterations for both the LindyHop and the Ninjutsu sets of ReMoCap, using the Adam optimizer [36] with a base learning rate of 10^{-5} and a batch size of 128. We use 207 dimensional latent embedding and 6 layers in the transformer decoder with 3 heads to calculate the attention.

MixNMatch [20]. MixNMatch proposes an end-to-end framework to synthesize stylized reactive motion informed by multi-hot action labels. It operates in one of two settings, *interaction mixing* and *interaction matching*. In *interaction mixing*, it generates a reaction combining different classes of reactive styles according to the multi-label indicator. In *interaction matching*, it generates the reactive motion corresponding to the interaction type and the input motion. Our setting is similar to *interaction matching*, where we input the acting sequence into the model and synthesize the reactive sequence. We mask out the action label defining the interaction type from the input and train the reactor's motion X based on the actor's motion Y. We use the normalization technique mentioned in Sec. 5.1 in the main paper to normalize the actor's and the reactor's body poses. We train MixNMatch on an NVIDIA RTX A4000 GPU for about 3.6K iterations for both the LindyHop and the Ninjutsu sets of ReMoCap, using the Adam optimizer [36] with a base learning rate of 10^{-5} and a batch size of 16. We use 256 LSTM neurons for each spatial slice and 1,200 for the attention layer.

ComMDM [54]. ComMDM is proposed as a communication block between two MDMs [67] to coordinate interaction between two persons. It uses single-person motions from a pre-trained MDM as fixed priors, and a parallel composition with few-shot training that shows how two single-person motions coordinate for interactions. ComMDM is a single-layer transformer model that inputs the activations coming from the previous layer from the two MDM models, and learns to generate a correction term for the MDM models along with the initial pose of each person. ComMDM was originally trained for two motion tasks: *prefix completion* and *text-to-motion synthesis*. We follow the *prefix completion* setting of ComMDM which does not use textual annotations as a condition and was trained to complete 3 seconds of motion given a 1 second prefix. We train ComMDM on an NVIDIA RTX A4000 GPU for about 24K iterations for both the LindyHop and the Ninjutsu sets of ReMoCap, using the Adam optimizer [36] with a base learning rate of 10^{-5} and a batch size of 64. We use 256 dimensional latent embedding for the ComMDM block. During inference, we provide the full ground truth motion of actor Y into the first MDM module. Thus, the ComMDM block takes in the ground truth features from the first MDM module and the learned features from the second MDM module. In turn, the output of the second MDM module is the reactive motion X for our setting.

RAIG [65]. Role-Aware Interaction Generation (RAIG) is a diffusion-based model that learns two-person interactions by generating single-person motions for a designated role. The role is supplied in the form of textual annotations, which are translated into active and passive voices to ensure the text is consistent with each role. The model generates interactions with two transformers that share parameters, and a cross-attention module connecting them. The active and passive voice descriptions are proveded as inputs to the corresponding transformers responsible for generating the actor and the reactor. The transformers consist of cross-attention modules both for language and motion. To use RAIG as a baseline for our annotation-free setting, we mask out the cross-attention module for the language in both the transformers and train to generate two-person motions unconditionally. We normalize the interactions as described in the original paper [65]. We train RAIG on an NVIDIA RTX A4000 GPU for about 20K iterations for both the LindyHop and the Ninjutsu sets of ReMoCap, using the Adam optimizer [36] with a base learning rate of 2^{-4} and a batch size of 32. We use 512 dimensional latent embedding and 8 attention blocks. During inference, we freeze the transformer that learns the actor's motion Y. The other transformer generates the reactor's motion X, being influenced by the actor's ground truth motion.

InterGen [42]. InterGen is a diffusion-based approach that generates two-person motions from text prompts. It was originally trained by conditioning on rich textual annotations. It uses cooperative denoisers with novel weight-sharing and a mutual attention mechanism to improve interactions between two persons. To use it as a baseline in our annotation-free setting, we mask out the text embeddings from the model input, and train InterGen to generate two-person

Table D.1: Quantitative evaluation on the InterHuman dataset [42]. We compare ReMoS with state-of-the-art motion synthesis methods on the Inter-Human [42] dataset. ↓: lower is better, ↑: higher is better, →: values closer to GT are better. **Bold** indicates best.

Methods	MPJPE (mm) ↓	MPJVE (mm) ↓	FID (body) ↓	Div →	Multi-modality ↑
GT	–	–	–	7.74	–
ComMDM [54]	76.4	2.75	0.72	7.17	1.71 ± 0.5
RAIG [65]	83.2	2.76	0.67	7.26	2.01 ± 0.6
InterGen [42]	69.5	2.61	0.59	7.32	2.11 ± 0.6
ReMoS (ours)	**66.7**	**2.56**	**0.56**	**7.33**	**2.13 ± 0.3**

Table D.2: Trainable parameter counts.

Method	Params (full model)
InterFormer	$8.2M$
MixNMatch	**6.5M**
ComMDM	$22.2M$
RAIG	$81.2M$
InterGen	$170M$
ReMoS (ours)	$17.4M$

motions (both actor and reactor) unconditionally. We use the non-canonical motion representation proposed in the original paper [42]. During inference, we use the customization used in InterGen for *person-to-person generation.* We take a single-person motion (the actor's motion Y) as input, and freeze it during the forward diffusion process. The frozen weights from the first person propagate into the model, which then uses the ground truth actor's motions to reconstruct the second person's motion (the reactor's motion X). We train InterGen on an NVIDIA RTX A4000 GPU for about 30K iterations for both the LindyHop and the Ninjutsu sets of ReMoCap, using the Adam optimizer [36] with a base learning rate of 10^{-4}, a cosine LR scheduler, and a batch size of 64.

D   Additional Results

We provide additional results and the trainable parameter counts of all models. We further show how ReMoS can be used as a motion editing tool for character control applications.

D.1   Quantitative Evaluation on the InterHuman Dataset [42]

We report additional evaluation of ReMoS compared to its diffusion-based baselines on the InterHuman [42] dataset in Table D.1. We report performance on the standard evaluation metrics, including MPJPE, MPJVE, FID, Diversity and Multi-modality. For Multi-modality, we generate each sequence 5 times and report numbers with a 95% confidence interval. InterHuman dataset does not provide hand motions, so we only evaluate the reactors' body motions. ReMoS achieves state-of-the-art performance in the aforementioned metrics in the InterHuman dataset, highlighting the utility of our method for diverse forms of two-person interactions.

Table D.3: Quantitative evaluation on body joints. We compare the body synthesis module of ReMoS with state-of-the-art motion synthesis methods on body joints only. **Bold** indicates the best.

Methods	Lindy Hop (body only)				Ninjutsu (body only)			
	MPJPE ↓	MPJVE ↓	FID ↓	Div →	MPJPE ↓	MPJVE ↓	FID ↓	Div →
GT	-	-	-	7.62	-	-	-	11.5
MixNMatch	69.8	10.5	0.74	2.52	260.1	5.14	0.72	4.94
InterFormer	63.2	8.91	0.52	4.64	262.5	3.53	0.51	6.27
ComMDM	50.2	4.42	0.23	7.51	192.4	3.45	0.25	9.83
RAIG	68.5	4.01	0.26	9.02	188.3	4.25	0.19	10.14
InterGen	55.1	2.87	0.22	7.49	165.5	3.82	0.23	9.87
ReMoS (ours)	**40.2**	**2.21**	**0.12**	**7.52**	**137.2**	**3.19**	**0.16**	**10.26**

D.2   Trainable Parameters

We report the total number of trainable parameters of ReMoS as compared to the baseline methods. Table D.2 shows that ReMoS has lesser trainable parameters than the existing diffusion-based two-person synthesis models [42, 54, 65].

D.3   Comparison with baselines without hand motions.

We compare the body synthesis module of ReMoS with baselines trained only on the body joints (Table D.3). We report state-of-the-art performance for ReMoS even when finger joints are not included.

D.4   Motion Editing Applications of ReMoS

We describe how to use ReMoS as an interactive motion editing tool, providing control to animators for tasks such as *pose completion* and *motion in-betweening*. These are crucial applications that are possible due to the strong generative abilities of DDPMs. We provide visual results of these applications in our supplementary video.

Pose Completion with Controlled Joints. When an animator manually customizes some of the reactor's body joints to align with specific animation tasks, ReMoS can automatically synthesize the reactor's remaining body joints to complete the reactor's motion. We achieve this by providing the forward-diffused values of the controlled joints as the network input at each diffusion step. For example, to synthesize the motions of the remaining joints of the reactor's body given customized motions for some joints J_i and J_k, we set

$$X_B^{(0)} = f_{\theta_B}\left(X_B^{(t)}, t, Y_B, \mathbb{1}_{\{J_i, J_k\}}\right), \tag{D.1}$$

where $\mathbb{1}_{\{J_i, J_k\}}$ is a mask we use at each denoising step on all frames to ensure that the joints J_i and J_k are not denoised. Instead, we populate J_i and J_k with

the identical noise vectors as the ones used during forward diffusion, while introducing random noise to the rest of the joints throughout the sequence. Thus, animators can incorporate flexible spatial control over chosen joints while ReMoS synthesizes the remaining joints of the reactor to faithfully capture the interaction. In Fig. 5b in the main paper, we show the results of a pose-completion application where we manually control the right-hand wrist joint of the reactor and let ReMoS synthesize the remaining body joints conditioned on the actor.

Motion In-Betweening. Likewise, we can use the existing framework to perform motion in-betweening for the reactive sequence. We achieve this by providing some keyframes of the reactive motion and letting ReMoS synthesize the intermediate frames using a motion in-betweening routine. To synthesize the reactor's motion between two given keyframes N_a and N_b through reverse diffusion, we set

$$X_B^{(0)} = f_{\theta_B}\left(X_B^{(t)}, t, Y_B, \mathbb{1}_{\{N_a, N_b\}}\right), \tag{D.2}$$

where $\mathbb{1}_{\{N_a, N_b\}}$ is a mask we use at each denoising step to ensure that all joints at frames N_a and N_b are not denoised. Thus, ReMoS can fill in the motions between the two seed frames as shown in Fig. 5c in the main paper.