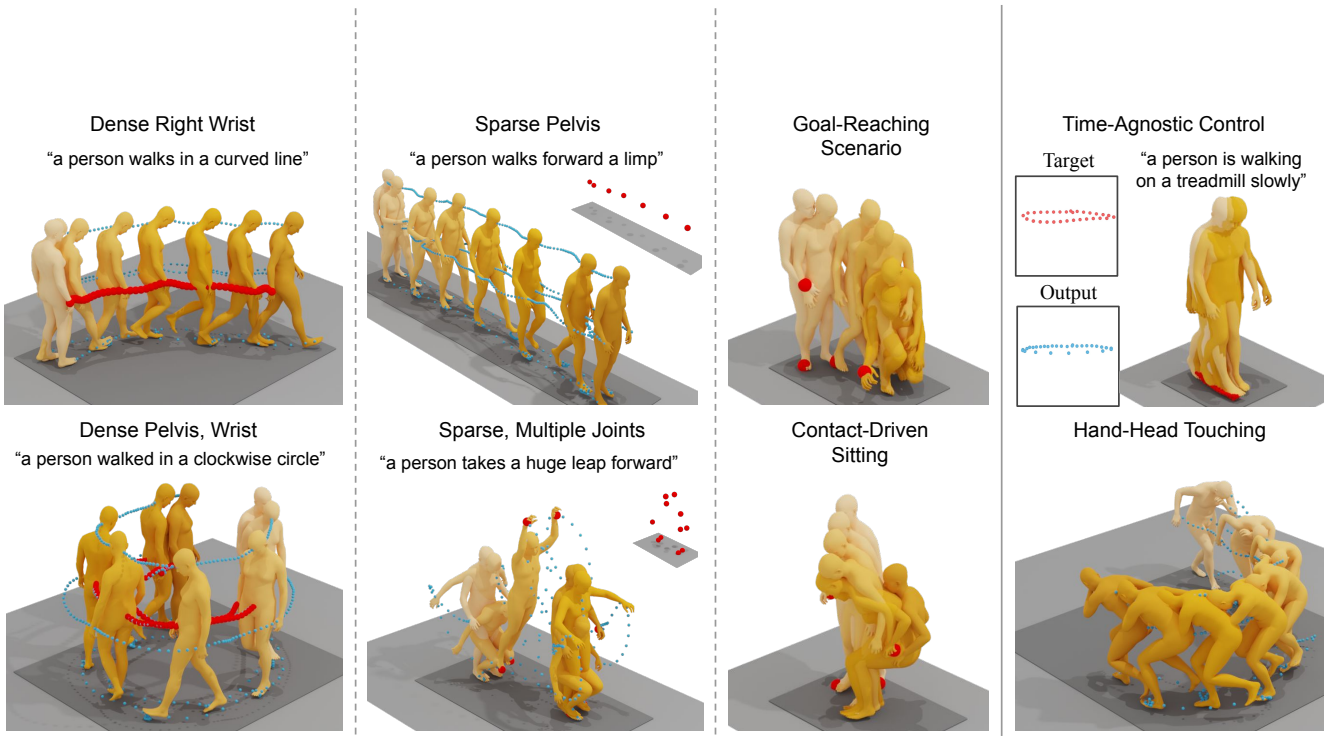
This ICCV paper is the Open Access version, provided by the Computer Vision Foundation.
Except for this watermark, it is identical to the accepted version;
the final published version of the proceedings is available on IEEE Xplore.

Motion Synthesis with Sparse and Flexible Keyjoint Control

Inwoo Hwang[1] Jinseok Bae[1] Donggeun Lim[1] Young Min Kim[1,2†]



(a) Dense Signal (b) Sparse Signal (c) Goal-Driven (d) Implicit Control

Figure 1. We enable a wide range of practical and controllable motion generation with high quality and precision. Our approach synthesizes natural human motion from *explicit* signals, including (a) *dense* signals with *multiple* joints, (b) *sparse* signals, and (c) *goal*-driven scenarios. Additionally, we generate motion from (d) *implicit* control signals defined via objective functions, including *time-agnostic* control.

Abstract

Creating expressive character animations is labor-intensive, requiring intricate manual adjustment of animators across space and time. Previous works on controllable motion generation often rely on a predefined set of dense spatio-temporal specifications (e.g., dense pelvis trajectories with exact per-frame timing), limiting practicality for animators. To process high-level intent and intuitive control in diverse scenarios, we propose a practical controllable motions synthesis framework that respects sparse and flexible keyjoint signals. Our approach employs a decomposed diffusion-based motion synthesis framework that first synthesizes keyjoint movements from sparse input control signals and then synthesizes full-body motion based on the completed keyjoint trajectories. The low-dimensional keyjoint movements can easily adapt to various control signal types, such as end-effector position for diverse goal-driven motion synthesis, or incorporate functional constraints on a subset of keyjoints. Additionally, we introduce a time-agnostic control formulation, eliminating the need for frame-specific timing annotations and enhancing control flexibility. Then, the shared second stage can synthesize a natural whole-body motion that precisely satisfies the task requirement from dense keyjoint movements. We demonstrate the effectiveness of sparse and flexible keyjoint control through comprehensive experiments on diverse datasets and scenarios. Project page: http://inwoohwang.me/SFControl

†Corresponding author

1. Introduction

Motion generation with precise control is critical for the interactivity and realism of character animations in the gaming, virtual reality, and robotics industries. Traditional animation workflows are labor intensive, requiring animators to specify detailed spatial and temporal constraints manually. While recent advances synthesize motions that abide by high-level intent signals (e.g., text or audio), scene-aware or interactive tasks may require fine-grained and precise positional control. In response, emerging methods generate motion guided by spatial control signals. However, their required inputs are dense frame-wise locations with exact timing, which are challenging to specify manually, severely limiting the practicality of real-world applications.

Instead, we introduce a high-quality motion synthesis framework that abides by sparse and flexible intuitive user control. Our framework is inspired by the observation that human motion can be intuitively described by the low-dimensional keyjoints movements, such as the end effectors (hands or feet) or body location (root or head). Our decomposed framework consists of two stages. The first stage synthesizes keyjoint movements, which precisely adhere to the highly sparse control inputs. Since the first stage manages low-dimensional data compared to full-body motion, it is highly effectively adapted to respect a wide range of sparse signals. The subsequent second stage then completes the natural full-body motion from the obtained dense keyjoint trajectories from the first stage.

While it is challenging to generate high-dimensional full-body motion that achieves both naturalness of motion and fidelity to the explicit control signal, the decomposition effectively disentangles controllability from motion quality. Both stages employ diffusion models and use an imputation strategy while training to allow explicit control [10, 12, 23]. In the first stage, the imputed information is randomly selected keyjoint control signals, which earns the ability to adapt to different sets of joints for sparse control signals. The imputation for the second stage is the complete keyjoint trajectories, and the second stage can learn to generate a high-quality full-body motion with complete keyjoint signal. Both stages can condition on the text descriptions or action labels to maintain the natural semantic intent of the overall motion sequence.

We can flexibly define a desired objective function on a subset of keyjoints, achieving user-friendly control. For example, the animator can only specify joint trajectories without time information, which could not be processed with previous approaches. We can allow time-agnostic control on the keyjoint trajectory by formulating a differentiable optimization objective on the spatial trajectory regardless of the time stamp. Additionally, we can efficiently tackle goal-driven motion generation, where the goal pose serves as a highly sparse spatial constraint. We demonstrate that our generated motion can perform diverse tasks with different sets of goal specifications in a unified setup without dataset-specific training: reaching target hand positions [2], climbing with rock constraints [71], and sitting with hand control [78].

Our contributions can be summarized as follows: (1) We propose a decomposed, diffusion-based framework for controllable motion synthesis that accurately follows sparse control signals while maintaining high-quality motion. (2) Our method effectively respects highly sparse control signals by concentrating on low-dimensional keyjoints movements at first stage within decomposed framework (3) We formulate diverse keyjoint constraints that enhance user controllability, such as time-agnostic control, goal-driven motion synthesis, and various constraint functions.

2. Related Works

Human Motion Generation with Semantic Input Recent advances in data-driven human motion synthesis have enabled the generation of highly realistic and diverse human motions. Modern generative modeling techniques—such as generative adversarial networks (GANs) [69], variational autoencoders (VAEs) [17], token-based models [19], and diffusion models [62]—have successfully captured the complexity of human motion dynamics. Many works focused on synthesizing motions that adapt to high-level semantic conditions (e.g., action categories [9, 16, 25, 49, 62], text descriptions [5, 17–20, 44, 50, 51, 62, 73, 74, 76], audio [1, 15, 37, 42, 53, 57, 63**?**]) and surrounding environments (e.g., interacting objects [4, 7, 14, 36, 40, 41, 61, 70, 79] or scenes [8, 26, 27, 29, 30, 38, 43, 60, 66, 67, 72, 80]). These methods bridge the gap between user intent and complex motion generation, enabling intuitive applications for animation.

Motion Synthesis with Full-Body Pose Input In addition to high-quality motion generation from high-level intents (e.g., text prompts), allowing spatially precise control is critical for practical applications, particularly in interactive animation workflows. A foundational task in this domain is motion in-betweening [3, 6, 13, 21, 22, 22, 28, 34, 35, 47, 54, 77], which synthesizes plausible transitions between keyframes defined as full-body information. Early works explored interpolation using splines such as Bézier curves [39, 46, 55], but these methods often lacked the flexibility to model complex human motion. Recent neural approaches have improved robustness: Qin et al. [54] proposed a two-stage deterministic network for motion completion, while CondMDI [10] leveraged generative capabilities of diffusion models and extended their abilities to incorporate text instructions.

Motion Synthesis with Joint-Level Control Beyond keyframe interpolation, recent methods focus on joint-level spatial control for advanced motion synthesis. Prior-MDM [56] pioneered joint-aware generation by fine-tuning a pre-trained motion diffusion model MDM [62]. GMD [56] introduces classifier-free guidance to diffusion models to enable root joint trajectory control, and OmniControl [68] generalized the control capability to arbitrary joints via ControlNet [75] modules with spatial guidance at inference. MotionLCM [11] further incorporated ControlNet into a motion latent space with a consistency model. Other approaches explored latent space optimization: TLControl [64] trained part-based discrete latent codes and performed inference-time code optimization, while ControlMM [52] exploited spatial guidance into a masked motion model via token optimization. DNO [31] optimized diffusion noise via a differentiable objective function defined through global joint position. A recent concurrent work, Studer *et al*. [59] introduced a factorized diffusion approach with Bézier curve parameterization, though it omits text conditioning.

Despite these advancements, most existing methods rely on dense control signals (e.g., full-sequence pelvis trajectories), and the performance degrades under sparse or partial joint constraints. They also assume strict temporal alignment between control signals and generated motions, limiting applicability to real-world scenarios with asynchronous or ambiguous timing. Addressing these gaps remains essential for practical deployment in interactive animation systems.

3. Motion Synthesis with Keyjoint Control

Given joint-level control signals, our goal is to generate natural and precise full-body character motion while satisfying the input constraints. We define keyjoints set $\mathcal{J}$, as the end-effectors (hands and feet) and critical body locations (pelvis (root) and head), which collectively capture the essential dynamics of full-body human motion. The control signals are either the *explicit* positions with corresponding timing inputs or an *implicit* function of a set of keyjoints.

Inspired by the observation that human motion can be effectively described through keyjoint dynamics, we propose a decomposed diffusion framework for controllable motion synthesis. The pipeline operates in two stages: we first synthesize keyjoint trajectories that precisely satisfy the input control signal (Sec. 3.1), and subsequently synthesize full-body motion conditioned on these trajectories (Sec. 3.2). The keyjoint movements $\mathbf{C}$ serve as a low-dimensional intermediate representation that contains the essential dynamics of human motion $\mathbf{X}$, thereby decomposing the motion synthesis problem into simplified stages. The overall process is illustrated in Figure 2.

Motion Representation The keyjoint trajectory representation is denoted as $\mathbf{C} = \{\mathbf{c}^n\}_{n=1}^N$, where each $\mathbf{c}^n \in \mathbb{R}^d$ encodes the kinematic state of the n-th frame's keyjoint. Here, d denotes the dimensionality of the joint's feature representation, including the global positions of keyjoints $J \in \mathbb{R}^{6\times 3}$ and a root's y-axis rotation $\gamma \in \mathbb{R}^1$. Full-body motion is represented as $\mathbf{X} = \{\mathbf{x}^n\}_{n=1}^N$, where each $\mathbf{x}^n \in \mathbb{R}^D$ represents a full-body pose with dimensionality $D \gg d$. We adopt the HumanML3D [17] representation and convert the root information into global coordinate system. This conversion aligns with the methodology in [32], where the feature dimension is $D = 263$.

During the first stage, we synthesize the movement of the keyjoint set $\mathcal{J}$ in the global coordinate. The resulting keyjoint trajectory $\mathbf{C} = \{\mathbf{c}^n\}_{n=1}^N$ precisely reflects joint-level constraints, while providing essential and sufficient information for predicting full-body movement. We then convert $\mathbf{C}$ to be in the coordinates relative to the root to align with the full-body representation. In the next stage, we synthesize the complete full-body motion $\mathbf{X} = \{\mathbf{x}^n\}_{n=1}^N$.

3.1. Keyjoint Trajectory Model

We primarily train the diffusion model using a conditional data distribution $p(\mathbf{C}_0, l)$, where $\mathbf{C}_0$ is a keyjont trajectory and l denotes an additional condition (e.g., text or action label). During training a diffusion model, noise is progressively added to the data at timestep t within the total diffusion steps T. The noisy trajectory sample at timestep t is generated by $\mathbf{C}_t = \sqrt{\bar{\alpha}_t}\mathbf{C}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$, where $\bar{\alpha}_t$ is determined by diffusion noise scheduling and ϵ is noise sampled from i.i.d. Gaussian distribution.

The goal of this stage is to synthesize keyjoint movements $\mathbf{C}$ that satisfy the given keyjoint control signal $\mathbf{s}$. The keyjoint control signal $\mathbf{s}$ is either explicit *xyz* positions of selected keyjoints, indicated by a binary indicator $\mathbf{m_s} \in \{0,1\}^{N\times d}$, or an implicit objective function $\mathcal{F}(\cdot)$ defined over the keyjoint set $\mathcal{J}$. Below, we further elaborate the formulations incorporating explicit control signal (Sec. 3.1.1) and implicit control signal (Sec. 3.1.2).

3.1.1. Motion Generation with Explicit Keyjoint Control

The input for explicit control signal is provided by 3D *xyz* locations with exact time stamps. Therefore, we can employ imputation scheme to process sparse selections in both time and joint indices (Figure 2, top left).

During training, the denoising network incorporates an imputation strategy with a randomly generated mask, earning the flexibility to handle arbitrary control signals. We select a timestep t from the uniform distribution over $[1, T]$ and generate a random spatio-temporal mask $\mathbf{m_C} \in \{0,1\}^{N\times d}$. This mask $\mathbf{m_C}$ indicates which joints will be used for conditional motion synthesis. We then impute the noisy sample $\mathbf{C}_t$ with the corresponding values from the clean sample $\mathbf{C}_0$:

$$\mathbf{C}_t' = \mathbf{m_C} \circ \mathbf{C}_0 + (1 - \mathbf{m_C}) \circ \mathbf{C}_t,$$

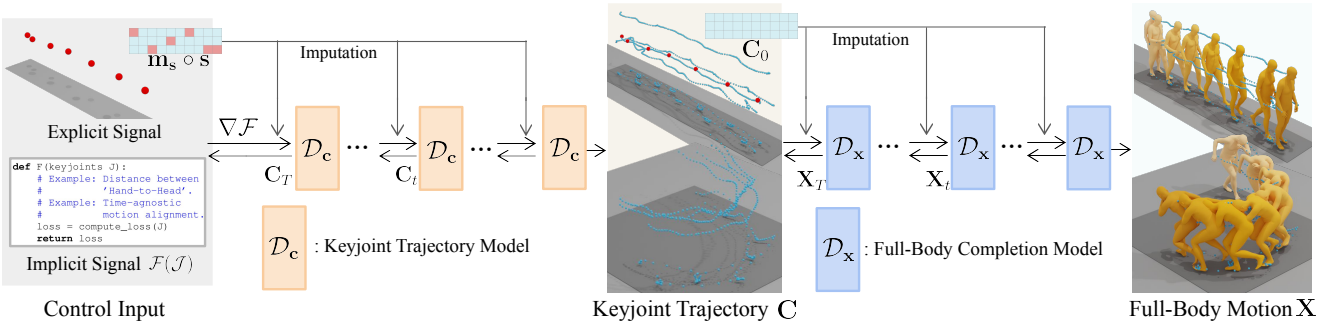


Figure 2. From control input, we first synthesize the keyjoint trajectory $\mathbf{C} \in R^{N\times d}$ as an intermediate representation. For *explicit* control, the signal $\mathbf{s}$ is combined with the binary indicator $\mathbf{m_s} \in \{0,1\}^{N\times d}$. We impute $\mathbf{m_s} \circ \mathbf{s}$ to the keyjoint trajectory and denoise it through keyjoint trajectory model $\mathcal{D}_\mathbf{c}$ at each diffusion step. For *implicit* control, an objective function $\mathcal{F}(\cdot)$ is defined over the keyjoint $\mathcal{J}$ and optimized thorough diffusion latent optimization. The resulting keyjoint trajectory $\mathbf{C}$ is then imputed into the full-body motion completion model $\mathcal{D}_\mathbf{x}$ to synthesize the final full-body motion $\mathbf{X} \in R^{N\times D}$.

where $\circ$ denotes element-wise multiplication. Next, we concatenate the imputed sample $\mathbf{C}'_t$ with the indicator mask $\mathbf{m_C}$ to form the combined feature:

$$\tilde{\mathbf{C}}_t = \langle \mathbf{C}'_t, \mathbf{m_C} \rangle.$$

Finally, we train a keyjoint trajectory denoiser $\mathcal{D}_{\mathbf{c},\theta}$ by minimizing the following loss function:

$$\mathcal{L}_{\text{simple},\mathbf{c}} = \mathbb{E}_{\mathbf{C}_0 \sim p(\mathbf{C}_0|l), t\sim[1,T]} \left[\left\| \mathbf{C}_0 - \mathcal{D}_{\mathbf{c},\theta}(\tilde{\mathbf{C}}_t, t, l) \right\|_2^2 \right].$$

At inference, we synthesize the keyjoint trajectory $\mathbf{C}_0$ from the explicit control signal $\mathbf{s}$, along with the binary indicator $\mathbf{m_s} \in \{0,1\}^{N\times d}$. We start inference by imputing noisy samples with $\mathbf{s}$, similar to the training process

$$\mathbf{C}'_t = \mathbf{m_s} \circ \mathbf{s} + (1 - \mathbf{m_s}) \circ \mathbf{C}_t.$$

Then, we use the concatenated temporal features $\tilde{\mathbf{C}}_t = \langle \mathbf{C}'_t, \mathbf{m_s} \rangle$ and generate motion sequences using a conditional diffusion process with classifier-free guidance.

$$\hat{\mathbf{C}}_0 = \mathbf{w} \cdot \mathcal{D}_{\mathbf{c},\theta}(\tilde{\mathbf{C}}_t, t, l) + (1 - \mathbf{w}) \cdot \mathcal{D}_{\mathbf{c},\theta}(\tilde{\mathbf{C}}_t, t, \emptyset),$$

where $\mathbf{w}$ is the guidance weight at inference [24].

Goal-Driven Motion Synthesis One of highly sparse user input for explicit control is specifying the goal position of a selected joint, such as a hand. Goal-driven motion synthesis aims to complete the transition motion starting from a given pose and reaching a target goal position defined for a specific keyjoint. To tailor our approach for the task, we adapt the binary indicator for the control signal $\mathbf{m_C}$ during training to include only the start pose and the target control joint at the final frame. We train a unified network on three different task configurations. In addition to the action label for the goal task, the network is conditioned on the continuous body shape encoding to account for the spatial relationship between the body size and the specific goal locations. The detailed training procedures and implementation details are provided in the supplementary material.

3.1.2. Motion Generation with Implicit Keyjoint Control

In addition to explicit spatial control, we can synthesize motion that satisfies high-level constraints, such as avoiding obstacles. Specifically, our framework can leverage implicit control signals, which are defined as differentiable objective functions $\mathcal{F}(\cdot)$ defined over the keyjoint set $\mathcal{J}$. The functional objectives can impose flexible and implicit constraints. For example, we can generate motions that demonstrate 'Hand-to-Head,' by defining the objective function to be the distance between the hand and head, and minimizing it.

We maintain the same model trained in the previous stage (Sec. 3.1.1), with modifications applied only at inference time. Since the implicit control signal does not contain frame-wise index, we cannot employ the imputation as described in Sec. 3.1.1. Instead, we refine motion sequences by back-propagating the derivatives of the objective function while iteratively optimizing the diffusion noise in the latent space [31]. The motion sequence satisfies the desired functional criteria while maintaining natural movement.

Time-Agnostic Motion Control Our functional constraint can allow trajectory input without exact per-frame timestamps, namely *time-agnostic motion control*. One critical limitation of explicit spatial motion control is the requirement for the exact frame-wise timing. Such information is highly challenging for an animator to accurately annotate, limiting their practicality in real-world scenarios.

We can formulate a flexible input constraint based on arc-length reparametrization. We assume the input composed of only a start time t_0, an end time t_1, and the desired 3D target trajectory $\mathcal{T}$ of the specific keyjoint over time interval $[t_0, t_1]$. Given a target constraint $(\mathcal{T}, t_0, t_1)$, our method en-

forces geometric consistency against the generated keyjoint trajectories $\mathbf{C} = \{\mathbf{c}^n\}_{n=1}^N \in \mathbb{R}^{N\times 3}$ through time-agnostic trajectory alignment process. We first extract the segment $\mathbf{C}_{\text{seg}}$ of the specified keyjoint during the time window $[t_0, t_1]$ from the generated trajectory $\mathbf{C} = \{\mathbf{c}^n\}_{n=1}^N$:

$$\mathbf{C}_{\text{seg}} = \{\mathbf{c}^n\}_{n=t_0}^{t_1} \in \mathbb{R}^{L\times 3}, \quad L = t_1 - t_0 + 1.$$

Then, we calculate cumulative arc lengths $\{s_i\}_{i=1}^L$ of $\mathbf{C}_{\text{seg}}$:

$$s_i = \begin{cases} 0, & i = 1 \\ s_{i-1} + \|\mathbf{c}^{t_0+i-1} - \mathbf{c}^{t_0+i-2}\| & i = 2, \dots, L \end{cases}$$

where s_L is the total arc length. After that, we obtain a cubic spline parametrized by the temporal intervals $\mathcal{S}(s) : [0, s_L] \mapsto \mathbb{R}^3$, which approximates $\mathbf{C}_{\text{seg}}$, with a differentiable process [33]. To obtain a temporally invariant representation, we uniformly sample L points on the spline and construct a version with arc-length parameterization $\mathbf{C}_{\text{seg}}^{\text{res}} = \{\mathcal{S}(\tilde{s}_k)\}_{k=1}^L \in \mathbb{R}^{L\times 3}$:

$$\tilde{s}_k = \frac{k-1}{L-1} \cdot s_L, \quad k \in \{1, \dots, L\}.$$

Similarly, we uniformly sample the target trajectory $\mathcal{T}$ to obtain $\mathcal{T}^{\text{res}} \in \mathbb{R}^{L\times 3}$. The *time-agnostic motion alignment loss* combines geometric and scale consistency:

$$\mathcal{L}_{\text{align}} = \underbrace{\left\|\mathbf{C}_{\text{seg}}^{\text{res}} - \mathcal{T}^{\text{res}}\right\|}_{\text{Traj Align.}} + \lambda_l \cdot \underbrace{|s_L - L(\mathcal{T})|}_{\text{Length Const.}}, \tag{1}$$

where $L(\mathcal{T})$ is the total length of $\mathcal{T}$. The first term ensures geometric consistency by aligning the uniform samples of the generated trajectory with the target trajectory, independent of temporal information, while the second ensures consistent motion scale. By optimizing the diffusion latent noise [31] with the proposed objective in Eq. (1), our method enables control over the keyjoint trajectory without requiring exact per-frame time stamps.

3.2. Full-Body Motion Completion Model

The second stage generates a natural full-body motion sequence $\mathbf{X} = \{\mathbf{x}^n\} \in \mathbb{R}^{N\times D}$ that follows the dense keyjoint trajectory $\mathbf{C}$ from the first stage. We train the conditional diffusion model from the joint data distribution $p(\mathbf{C}_0, \mathbf{X}_0, l)$.

Training During training, we add noise corresponding to diffusion timestep t and obtain noisy full-body motion $\mathbf{X}_t = \sqrt{\bar{\alpha}_t}\mathbf{X}_0 + \sqrt{1-\bar{\alpha}_t}\boldsymbol{\epsilon}$. Similar to the keyjoint model training, we impute the clean keyjoint information $\mathbf{C}_0$ into $\mathbf{X}_t$. Specifically, given a noisy full-body motion sample $\mathbf{X}_t$ at diffusion timestep t, we replace its corresponding keyjoint component with the ground truth from $\mathbf{C}_0$:

$$\mathbf{X}_t^{\text{keyjoints}} = \mathbf{C}_0.$$

This imputation preserves the provided keyjoint movement throughout the diffusion process, effectively guiding the full-body motion synthesis.

The full-body denoiser $\mathcal{D}_{\mathbf{x},\theta}$ is then trained to reconstruct the original full-body motion $\mathbf{X}_0$ from the noisy input $\mathbf{X}_t$, conditioned on l. The training objective is given by:

$$\mathcal{L}_{\text{simple},\mathbf{x}} = \mathbb{E}_{\mathbf{X}_0\sim p(\mathbf{X}_0|\mathbf{C}_0,l),t\sim[1,T]}\left[\|\mathbf{X}_0 - \mathcal{D}_{\mathbf{x},\theta}(\mathbf{X}_t, t, l)\|_2^2\right].$$

Inference At inference, we synthesize the full-body motion $\mathbf{X}_0$ given estimated keyjoint trajectories $\hat{\mathbf{C}}_0$. We start inference by imputing noisy samples with $\hat{\mathbf{C}}_0$, similar to the training process

$$\mathbf{X}_t^{\text{keyjoints}} = \hat{\mathbf{C}}_0$$

and sample via a conditional diffusion process:

$$\hat{\mathbf{X}}_0 = \mathbf{w} \cdot \mathcal{D}_{\mathbf{x},\theta}(\mathbf{X}_t, t, l) + (1-\mathbf{w}) \cdot \mathcal{D}_{\mathbf{x},\theta}(\mathbf{X}_t, t, \emptyset),$$

where $\mathbf{w}$ is the guidance weight.

4. Experiments

We first evaluate our controllable motion synthesis framework in explicit control signal where exact time and locations for keyjoints are given (Sec. 4.1). Additionally, we present results from implicit motion control via differentiable objectives defined with keyjoints (Sec. 4.2). After analyzing full-body motion completion model (Sec. 4.3), we further investigate diverse aspects of our keyjoint control pipeline (Sec. 4.4). We provide detailed descriptions on the datasets and baselines in the supplementary material.

Implementation Details Our pipeline is implemented in PyTorch [48] and trained on a single NVIDIA RTX 2080 GPU. We adopt the U-Net architecture from [32]. Further details on model architecture and diffusion hyperparameters are provided in the supplementary material.

4.1. Explicit Keyjoint Control

We begin our evaluation using the HumanML3D dataset, which contains text annotations paired with motion data captured at 20 FPS. We compare our method with state-of-the-art approaches for joint-level motion synthesis, selecting those with publicly available code.

For our experiments, we evaluate our framework under a highly sparse control setting. We set the control signal interval to $r = 30$ frames (i.e., one frame every 1.5 seconds) and further randomly select 50% out of six keyjoints at each frame. This corresponds to only 0.454% of the total joint signals contributing to the full-body motion, calculated as $(1/30) \times (6/22) \times (1/2) = 0.00454$, where $1/30$ accounts for the frame sampling rate, $6/22$ represents the ratio of key

joints to total joints, and $1/2$ reflects the random retention rate of control signals.

We evaluate our framework across spatial control accuracy, motion realism, text fidelity, and diversity. For spatial control accuracy, we assess motion controllability via *Control Error* by computing the Euclidian distance between generated motions and the provided spatial targets. Motion quality is evaluated through two metrics: *Frechet Inception Distance (FID)* which measures distribution alignment between ground truth and generated, and the *Foot Skating Ratio* qunatifies the artifact of foot's movement. *R-Precision* measures text-motion alignment and *Diversity* measures the variability within the generated motion.

Table 1 presents the quantitative results. Compared to other baselines, our model achieves lower *FID* scores, lower *control error*, while maintaining *diversity* close to the ground truth, demonstrating superior motion quality and precision under sparse control conditions. These results highlight the effectiveness of our approach in generating high-quality, diverse, and precise motions from sparse control input. The two stage diffusion framework is not real-time as it involves multiple sampling steps (DDPM). We also test a variation of our pipeline using the DDIM [58] sampling strategy, which reduces sampling time, with sampling steps of five (DDIM5) and ten (DDIM10). Notably, our method consistently outperforms baselines even when employing the faster DDIM sampling strategy. We provide more analysis of runtime performance in Sec. 4.4.

Method	FID ↓	Control Err. (m) ↓	R-precision (Top-3) ↑	Div. →	Foot Skating ↓
Real	0.002	0.000	0.797	9.503	0.000
CondMDI [10]	0.498	0.507	0.631	9.013	0.099
OmniControl [68]	0.689	0.111	**0.689**	<u>9.381</u>	0.091
TLControl [64]	4.637	0.128	0.473	8.078	**0.058**
MotionLCM [11]	1.013	0.418	<u>0.676</u>	8.722	0.141
Ours (DDIM5)	0.338	0.037	0.655	9.356	0.065
Ours (DDIM10)	<u>0.254</u>	<u>0.037</u>	0.673	**9.621**	0.063
Ours	**0.224**	**0.036**	0.673	9.674	<u>0.061</u>

Table 1. Quantitative evaluation of the sparse joint control. We validate our framework under a highly sparse control setting, which uses only 0.454% of total joint signal as control input. **Bold** represents the best value, and <u>underlined</u> represents the second-best.

4.1.1. Goal-Driven Motion Synthesis

As a practical extension of sparse explicit control, we adapt the framework for goal-driven motion synthesis by training a unified network that integrates multiple motion scenarios. In each scenario, control signals are defined by an initial pose and target control joints at the final frame. Specifically, *reaching target hand positions* [2] focuses on controlling the right-hand position, *climbing with rock constraints* [71] requires coordination of both hands and feet, and *sitting with hand control* [78] necessitates precise control of both hands at the final frame, as illustrated in Figure 3. Our unified network is designed to handle multiple motion dynamics and control settings within a single framework.

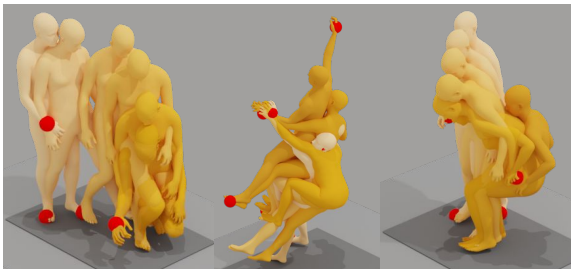


Figure 3. Qualitiative results of goal-driven motion scenarios, demonstrating *reaching target hand positions*, *climbing with rock constraints*, and *sitting with hand control*, respectively.

We benchmark our approach against a single-stage version that does not detach keyjoint movements during training. To assess goal-reaching accuracy, we measure the average Euclidean distance between the final generated poses and the target keyjoint positions (*Distance to Goal*). And also measure the deviation from the input starting full-body pose (*Distance to Start*), and *Foot Skating*. As shown in Tab. 2, our decomposed pipeline demonstrates substantial advantages by prioritizing on low-dimensional key joint movements in the first stage. Ours can satisfy the start and the goal positions more precisely with minimal foot skating artifacts. Furthermore, our model maintains reasonable performance even with limited task-specific data: For the *sitting with hand control* and *climbing with rock constraints* scenarios, we train on only 160 and 156 motion segments, respectively, yet achieve competitive results. We further analyze generalization performance to different control inputs in Sec. 4.4.

	Scenario	Dist. to Goal (m) ↓	Dist. to Start (m) ↓	Foot Skating ↓
Ours w/o decomp.	All	0.206	0.116	0.057
	Reaching	0.141	0.061	0.051
	Climbing	0.529	0.409	0.079
	Sitting	0.327	0.126	0.092
Ours	All	**0.093**	**0.065**	**0.047**
	Reaching	**0.054**	**0.043**	**0.048**
	Climbing	**0.288**	**0.185**	**0.046**
	Sitting	**0.156**	**0.064**	**0.050**

Table 2. Quantitative evaluation on the *goal-driven* scenarios. We train a unified network across three different tasks and evaluate it separately for each task as well as collectively.

4.2. Implicit Keyjoint Control

Additionally, we evaluate our approach under implicit control settings, where constraints are defined through differentiable functions, including a time-agnostic control scenario.

Task	Method	Constraint Error ↓	Critic Score ↑	Foot Skating
Walking Narrow	DNO [31]	0.0101	-2.024	0.142
	Ours w/o decomp.	0.0093	-2.192	0.090
	Ours	**0.0084**	**0.253**	**0.066**
Hand-to -Head	DNO [31]	0.0084	-6.303	0.064
	Ours w/o decomp.	0.0093	-5.842	0.059
	Ours	**0.0036**	**-2.505**	**0.045**
Time-Agnostic	DNO [31]	0.0412	0.246	0.061
	Ours w/o decomp.	0.0581	0.234	0.058
	Ours w/o time-agnostic	0.4827	0.213	0.097
	Ours	**0.0256**	**0.398**	**0.048**

Table 3. Quantitative evaluation on different objective defined task scenarios.

To assess its effectiveness, we adapt evaluation tasks from [45], such as generating hand-to-head contact movements and walking in narrow spaces. To quantify performance, we report *Constraint Error*, which measures the degree to which user-defined constraints are satisfied, and the *Critic Score* [65], which assesses motion naturalness and quality based on human perceptual judgments. For each scenario, we randomly sample 10 instances for evaluation.

We compare our approach against both the original implementation of [31], which directly generates full-body motion, and our single-stage variant (referred to as "Ours w/o decomposed"). As presented in Table 3, our decomposed pipeline consistently improves constraint error while enhancing motion quality. By applying objectives at the keyjoint trajectory level rather than at the full-body motion synthesis stage, our approach better satisfies objective constraints while preserving the realism of the full-body motion after completion.

To further validate the effectiveness of our *time-agnostic motion control* framework, we compare our proposed implicit control approach with an explicit approach as "Ours w/o time-agnostic". Specifically, for a trajectory $\mathcal{T}$ where only the start and end times, t_0 and t_1, are provided, we sample points along the trajectory to have the same length between them. Then we assign timestamps with a uniform time interval between t_0 and t_1 on the sampled points and provide the locations as explicit control signal. Table 3 shows that the explicit model without exact time information struggles to accurately follow the given trajectory. In contrast, our implicit formulation for time-agnostic control effectively overcome the limitation with our dedicated time-agnostic motion alignment objective. Figure 4 visualizes the input trajectory $\mathcal{T}$, which lacks timing annotations, alongside the corresponding trajectory generated using our time-agnostic control. Notably, our method produces target trajectory with non-uniform intervals, effectively capturing velocity variations. Overall, our decomposed pipeline offers flexibility under implicit control, including time-agnostic motion alignment mechanism.

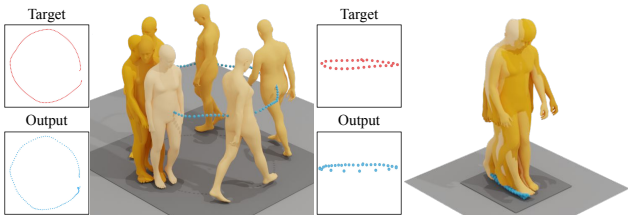


Figure 4. Example of time-agnostic trajectory target input and synthesized motion from time-agnostic control.

4.3. Full-Body Motion Completion Model

We also evaluate the performance of our full-body motion completion model to assess how accurately the synthesized full-body motion follows the key joint movements. In Tab. 4, we measure the *Mean Euclidean Distance Error* between input keyjoint trajectories and the final full-body motion derived from different keyjoint sources: ground-truth and synthesized keyjoints from keyjoint trajectory model. The results demonstrate that our full-body motion precisely adheres to the given key joint movements and that the synthesized keyjoint trajectories remain faithful to those of the ground truth.

KeyJoint Source	Ground Truth	Synthesized
Mean Distance Error (m)	0.0082	0.0115

Table 4. Quantitative evaluation of full-body motion completion model on different keyjoint source.

4.4. Additional Study

Runtime We analyze the runtime efficiency of our controllable motion synthesis approach. Each stage of our diffusion model is trained with a diffusion time step $T = 50$. Table 5 presents the sampling times of each baseline method for comparison.

While MotionLCM [11] demonstrates real-time runtime performance, it lacks controllability and produces relatively low-quality motion. Additionally, we validate that by leveraging the DDIM sampling strategy, our method not only reduces sampling time but also maintains high control precision and motion quality in Tab. 1 and Fig. 5. This allows us to outperform other optimization-based controllable motion synthesis works [64, 68] in terms of runtime efficiency, control accuracy, and overall motion quality. Note that our decomposed pipeline significantly reduces the runtime compared to other single-stage pipelines (CondMDI, Ours w/o decomp. OmniControl).

Method	MotionLCM [11]	Ours (DDIM 5)	Ours (DDIM 10)	TLControl [64]
Time (s)	0.034	0.7	1.4	3.5
Method	Ours (DDPM)	CondMDI [10]	Ours w/o decomp.	OmniControl [68]
Time (s)	7.1	39.5	41.2	152.6

Table 5. Time required for motion control.

Robustness to Varying
motion synthesis framew
mask, $\mathbf{m_C}$, within the ke
stable performance acros
trol signals. Table 6 pr
sets of control joints. Fo
we first sample frames
with a random probability
keyjoint control signals f
cess results in multiple jo
chosen frame. For the *pe*
schemes, frames are sim
of r or with probability p
signals are retained. The
sistently achieves strong
trol signal selection stra
robustness to different frame sampling and joint selection approaches, ranging from single to multi-joint control.

Joint Select	Frame Select	FID ↓	Control Err. (m) ↓	R-precision (Top-3) ↑	Div. →	Foot Skating ↓
-	-	0.002	0.000	0.797	9.503	0.000
Cross	$r = 1$	0.127	0.019	0.681	9.518	0.071
	$r = 2$	0.128	0.019	0.680	9.539	0.070
	$r = 5$	0.148	0.024	0.681	9.554	0.069
	$r = 10$	0.171	0.027	0.678	9.402	0.074
	$r = 20$	0.195	0.033	0.677	9.575	0.064
	$r = 30$	0.224	0.036	0.673	9.674	0.061
	$r = 60$	0.263	0.044	0.659	9.627	0.062
Cross	$p = 0.02$	0.261	0.037	0.658	9.621	0.061
	$p = 0.05$	0.210	0.030	0.673	9.589	0.063
	$p = 0.1$	0.179	0.027	0.677	9.489	0.065
	$p = 0.2$	0.160	0.023	0.680	9.566	0.067
	$p = 0.5$	0.135	0.020	0.681	9.535	0.070
	$p = 1.0$	0.127	0.019	0.681	9.520	0.072
Pelvis	$p = 0.02$	0.245	0.039	0.656	9.681	0.059
	$p = 0.5$	0.243	0.023	0.675	9.583	0.073
Right Wrist	$p = 0.02$	0.278	0.031	0.655	9.731	0.058
	$p = 0.5$	0.352	0.023	0.665	9.361	0.046

Table 6. Quantitative evaluation of diverse control signal selection strategies. Our method demonstrates robust performance, regardless of sparsity, the number or combinations of multiple joints.

Figure 5 presents a further analysis across varying levels of input sparsity. Our method maintains consistent performance even under highly sparse control signals while synthesizing natural motion with high precision. Furthermore, even when employing DDIM sampling, our method outperforms all baselines while benefiting from faster sampling speeds compared to our original sampling strategy with full diffusion steps. In contrast, the performance of baseline methods degrades as control signals become sparser. By focusing on low-dimensional keyjoint movements in a separate stage, our model achieves superior controllability and motion quality, regardless of control sparsity.

Experiments on Out-of-Distribution Signal We evaluate the robustness of our method to out-of-distribution (OOD) control signals by introducing perturbations into sparse control signals. Specifically, we select root positions with frame intervals of 20 or 30 as the sparse control, and inject random noise of up to 5 cm to create the OOD samples. As shown in Table 7, our method maintains robust performance and preserves controllability, demonstrating resilience to both in-distribution and OOD data. Since *FID* and *Diversity* metrics are designed to assess closeness to ground truth data, they are not reported in this setting.

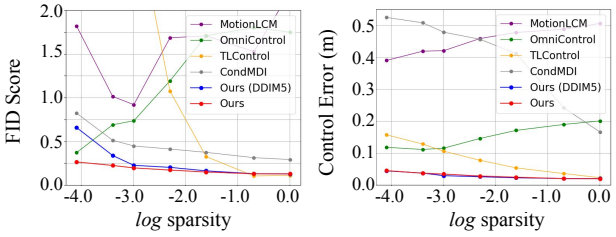


Figure 5. We plot the performance of log control sparsity (x-axis) against the *FID* score and *Control error* (y-axis), which assess motion quality and precision, respectively. Our framework maintains consistent performance across varying control input sparsity, outperforming all baselines.

Control Signal	Control Err. (m) ↓	R-precision (Top-3) ↑	Foot Skating ↓
$r = 20$ (In-data)	0.036	0.669	0.058
$r = 20$ (OOD)	0.039	0.676	0.061
$r = 30$ (In-data)	0.038	0.676	0.061
$r = 30$ (OOD)	0.043	0.661	0.060

Table 7. Experiments on Out-of-Distribution control signals. By comparing with in-distribution results (denoted as "In-data"), our method demonstrates robust performance with out-of-distribution signals.

5. Conclusion

We propose a controllable motion synthesis framework that utilizes sparse and flexible keyjoint inputs. Our decomposed, diffusion-based motion synthesis approach effectively handles highly sparse signals while preserving motion quality. Notably, with only 0.0453% of the input signals, our model achieves a control error of $0.036m$ and an *FID* score of 0.224, demonstrating strong adherence to provided constraints while preserving natural movement. We further demonstrate the effectiveness and versatility of our framework in goal-driven motion synthesis across a range of scenarios. Additionally, we formulate keyjoint constraints as various functions—including a time-agnostic control approach that eliminates the need for exact timestamps, allowing for temporally flexible synthesis. We believe that our framework holds significant potential for practical applications in controllable motion synthesis.

Our method is not yet real-time and incorporating the autoregressive sampling while executing motion technique described in [30] would require developing a real-time controller. We leave this extension for future work.

Acknowledgements This work was supported by Creative-Pioneering Researchers Program through Seoul National University and BK21 FOUR program of the Education and Research Program for Future ICT Pioneers, Seoul National University in 2025. Inwoo Hwang is supported by Hyundai Motor Chung Mong-Koo Foundation.

References

[1] Simon Alexanderson, Rajmund Nagy, Jonas Beskow, and Gustav Eje Henter. Listen, denoise, action! audio-driven motion synthesis with diffusion models. *ACM Transactions on Graphics (TOG)*, 42(4):1–20, 2023. 2

[2] Joao Pedro Araujo, Jiaman Li, Karthik Vetrivel, Rishi Agarwal, Deepak Gopinath, Jiajun Wu, Alexander Clegg, and C. Karen Liu. Circle: Capture in rich contextual environments, 2023. 2, 6

[3] Okan Arikan and David A Forsyth. Interactive motion generation from examples. *ACM Transactions on Graphics (TOG)*, 21(3):483–490, 2002. 2

[4] Jinseok Bae, Jungdam Won, Donggeun Lim, Cheol-Hui Min, and Young Min Kim. Pmp: Learning to physically interact with environments using part-wise motion priors, 2023. 2

[5] Jinseok Bae, Inwoo Hwang, Young Yoon Lee, Ziyu Guo, Joseph Liu, Yizhak Ben-Shabat, Young Min Kim, and Mubbasir Kapadia. Less is more: Improving motion diffusion models with sparse keyframes, 2025. 2

[6] Philippe Beaudoin, Stelian Coros, Michiel Van de Panne, and Pierre Poulin. Motion-motif graphs. In *Proceedings of the 2008 ACM SIGGRAPH/Eurographics symposium on computer animation*, pages 117–126, 2008. 2

[7] Bharat Lal Bhatnagar, Xianghui Xie, Ilya Petrov, Cristian Sminchisescu, Christian Theobalt, and Gerard Pons-Moll. Behave: Dataset and method for tracking human object interactions. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. IEEE, 2022. 2

[8] Zhi Cen, Huaijin Pi, Sida Peng, Zehong Shen, Minghui Yang, Zhu Shuai, Hujun Bao, and Xiaowei Zhou. Generating human motion in 3d scenes from text descriptions. In *CVPR*, 2024. 2

[9] Xin Chen, Biao Jiang, Wen Liu, Zilong Huang, Bin Fu, Tao Chen, and Gang Yu. Executing your commands via motion diffusion in latent space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 18000–18010, 2023. 2

[10] Setareh Cohan, Guy Tevet, Daniele Reda, Xue Bin Peng, and Michiel van de Panne. Flexible motion in-betweening with diffusion models. In *ACM SIGGRAPH 2024 Conference Papers*, pages 1–9, 2024. 2, 6, 7

[11] Wenxun Dai, Ling-Hao Chen, Jingbo Wang, Jinpeng Liu, Bo Dai, and Yansong Tang. Motionlcm: Real-time controllable motion generation via latent consistency model. In *ECCV*, pages 390–408, 2025. 3, 6, 7

[12] Giannis Daras, Kulin Shah, Yuval Dagan, Aravind Gollakota, Alexandros G. Dimakis, and Adam Klivans. Ambient diffusion: Learning clean distributions from corrupted data, 2023. 2

[13] Yinglin Duan, Tianyang Shi, Zhengxia Zou, Yenan Lin, Zhehui Qian, Bohan Zhang, and Yi Yuan. Single-shot motion completion with transformer. *arXiv preprint arXiv:2103.00776*, 2021. 2

[14] Anindita Ghosh, Rishabh Dabral, Vladislav Golyanik, Christian Theobalt, and Philipp Slusallek. Imos: Intent-driven full-body motion synthesis for human-object interactions. In *Eurographics*, 2023. 2

[15] Kehong Gong, Dongze Lian, Heng Chang, Chuan Guo, Zihang Jiang, Xinxin Zuo, Michael Bi Mi, and Xinchao Wang. Tm2d: Bimodality driven 3d dance generation via music-text integration. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 9942–9952, 2023. 2

[16] Chuan Guo, Xinxin Zuo, Sen Wang, Shihao Zou, Qingyao Sun, Annan Deng, Minglun Gong, and Li Cheng. Action2motion: Conditioned generation of 3d human motions. In *Proceedings of the 28th ACM International Conference on Multimedia*, pages 2021–2029, 2020. 2

[17] Chuan Guo, Shihao Zou, Xinxin Zuo, Sen Wang, Wei Ji, Xingyu Li, and Li Cheng. Generating diverse and natural 3d human motions from text. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 5152–5161, 2022. 2, 3

[18] Chuan Guo, Xinxin Zuo, Sen Wang, and Li Cheng. Tm2t: Stochastic and tokenized modeling for the reciprocal generation of 3d human motions and texts. In *ECCV*, 2022.

[19] Chuan Guo, Yuxuan Mu, Muhammad Gohar Javed, Sen Wang, and Li Cheng. Momask: Generative masked modeling of 3d human motions. 2023. 2

[20] Chuan Guo, Inwoo Hwang, Jian Wang, and Bing Zhou. Snapmogen: Human motion generation from expressive texts, 2025. 2

[21] Félix G Harvey and Christopher Pal. Recurrent transition networks for character locomotion. In *SIGGRAPH Asia 2018 Technical Briefs*, pages 1–4. 2018. 2

[22] Félix G Harvey, Mike Yurick, Derek Nowrouzezahrai, and Christopher Pal. Robust motion in-betweening. *ACM Transactions on Graphics (TOG)*, 39(4):60–1, 2020. 2

[23] William Harvey, Saeid Naderiparizi, Vaden Masrani, Christian Weilbach, and Frank Wood. Flexible diffusion modeling of long videos, 2022. 2

[24] Jonathan Ho and Tim Salimans. Classifier-free diffusion guidance, 2022. 4

[25] Nhat M. Hoang, Kehong Gong, Chuan Guo, and Michael Bi Mi. Motionmix: Weakly-supervised diffusion for controllable motion generation, 2024. 2

[26] Siyuan Huang, Zan Wang, Puhao Li, Baoxiong Jia, Tengyu Liu, Yixin Zhu, Wei Liang, and Song-Chun Zhu. Diffusion-based generation, optimization, and planning in 3d scenes. *arXiv preprint arXiv:2301.06015*, 2023. 2

[27] Inwoo Hwang, Jinseok Bae, Donggeun Lim, and Young Min Kim. Goal-driven human motion synthesis in diverse task. In *Proceedings of the Computer Vision and Pattern Recognition Conference (CVPR) Workshops*, pages 2920–2930, 2025. 2

[28] Inwoo Hwang, Bing Zhou, Young Min Kim, Jian Wang, and Chuan Guo. Scenemi: Motion in-betweening for modeling human-scene interactions, 2025. 2

[29] Nan Jiang, Zimo He, Zi Wang, Hongjie Li, Yixin Chen, Siyuan Huang, and Yixin Zhu. Autonomous character-scene interaction synthesis from text instruction, 2024. 2

[30] Nan Jiang, Zhiyuan Zhang, Hongjie Li, Xiaoxuan Ma, Zan Wang, Yixin Chen, Tengyu Liu, Yixin Zhu, and Siyuan Huang. Scaling up dynamic human-scene interaction modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1737–1747, 2024. 2, 8

[31] Korrawe Karunratanakul, Konpat Preechakul, Emre Aksan, Thabo Beeler, Supasorn Suwajanakorn, and Siyu Tang. Optimizing diffusion noise can serve as universal motion priors. In *arxiv:2312.11994*, 2023. 3, 4, 5, 7

[32] Korrawe Karunratanakul, Konpat Preechakul, Supasorn Suwajanakorn, and Siyu Tang. Guided motion diffusion for controllable human motion synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2151–2162, 2023. 3, 5

[33] Patrick Kidger. Torchcubicspline: a pytorch implementation of cubic splines. https://github.com/patrick-kidger/torchcubicspline, 2022. 5

[34] Jihoon Kim, Taehyun Byun, Seungyoun Shin, Jungdam Won, and Sungjoon Choi. Conditional motion in-betweening. *Pattern Recognition*, 132:108894, 2022. 2

[35] Lucas Kovar, Michael Gleicher, and Frédéric Pighin. Motion graphs. In *Seminal Graphics Papers: Pushing the Boundaries, Volume 2*, pages 723–732. 2023. 2

[36] Nilesh Kulkarni, Davis Rempe, Kyle Genova, Abhijit Kundu, Justin Johnson, David Fouhey, and Leonidas Guibas. Nifty: Neural object interaction fields for guided human motion synthesis, 2023. 2

[37] Hsin-Ying Lee, Xiaodong Yang, Ming-Yu Liu, Ting-Chun Wang, Yu-Ding Lu, Ming-Hsuan Yang, and Jan Kautz. Dancing to music. *Advances in neural information processing systems*, 32, 2019. 2

[38] Jiye Lee and Hanbyul Joo. Locomotion-action-manipulation: Synthesizing human-scene interactions in complex 3d environments. *arXiv preprint arXiv:2301.02667*, 2023. 2

[39] Andreas M. Lehrmann, Peter V. Gehler, and Sebastian Nowozin. Efficient nonlinear markov models for human motion. In *2014 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1314–1321, 2014. 2

[40] Jiaman Li, Alexander Clegg, Roozbeh Mottaghi, Jiajun Wu, Xavier Puig, and C. Karen Liu. Controllable human-object interaction synthesis, 2023. 2

[41] Jiaman Li, Jiajun Wu, and C Karen Liu. Object motion guided human motion synthesis. *ACM Trans. Graph.*, 42(6), 2023. 2

[42] Ruilong Li, Shan Yang, David A. Ross, and Angjoo Kanazawa. Ai choreographer: Music conditioned 3d dance generation with aist++, 2021. 2

[43] Donggeun Lim, Cheongi Jeong, and Young Min Kim. Mammos: Mapping multiple human motion with scene understanding and natural interactions. In *2023 IEEE/CVF International Conference on Computer Vision Workshops (ICCVW)*, pages 4280–4289, 2023. 2

[44] J. Lin, J. Chang, L. Liu, G. Li, L. Lin, Q. Tian, and C. Chen. Being comes from not-being: Open-vocabulary text-to-motion generation with wordless training. In *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[45] Hanchao Liu, Xiaohang Zhan, Shaoli Huang, Tai-Jiang Mu, and Ying Shan. Programmable motion generation for open-set motion control tasks. In *CVPR*, 2024. 7

[46] Tomohiko Mukai and Shigeru Kuriyama. Geostatistical motion interpolation. *ACM Trans. Graph.*, 24(3):1062–1070, 2005. 2

[47] Boris N Oreshkin, Antonios Valkanas, Félix G Harvey, Louis-Simon Ménard, Florent Bocquelet, and Mark J Coates. Motion in-betweening via deep δ-interpolator. *IEEE Transactions on Visualization and Computer Graphics*, 2023. 2

[48] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019. 5

[49] Mathis Petrovich, Michael J Black, and Gül Varol. Action-conditioned 3d human motion synthesis with transformer vae. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 10985–10995, 2021. 2

[50] Mathis Petrovich, Michael J. Black, and Gül Varol. TEMOS: Generating diverse human motions from textual descriptions. In *European Conference on Computer Vision (ECCV)*, 2022. 2

[51] Mathis Petrovich, Or Litany, Umar Iqbal, Michael J. Black, Gül Varol, Xue Bin Peng, and Davis Rempe. STMC: Multi-track timeline control for text-driven 3d human motion generation. *arXiv:2401.08559*, 2024. 2

[52] Ekkasit Pinyoanuntapong, Muhammad Usama Saleem, Korrawe Karunratanakul, Pu Wang, Hongfei Xue, Chen Chen, Chuan Guo, Junli Cao, Jian Ren, and Sergey Tulyakov. Controlmm: Controllable masked motion generation. *arXiv preprint arXiv:2410.10780*, 2024. 3

[53] Qiaosong Qi, Le Zhuo, Aixi Zhang, Yue Liao, Fei Fang, Si Liu, and Shuicheng Yan. Diffdance: Cascaded human motion diffusion model for dance generation. In *Proceedings of the 31st ACM International Conference on Multimedia*, page 1374–1382, New York, NY, USA, 2023. Association for Computing Machinery. 2

[54] Jia Qin, Youyi Zheng, and Kun Zhou. Motion in-betweening via two-stage transformers. *ACM Trans. Graph.*, 41(6):184–1, 2022. 2

[55] C. Rose, M.F. Cohen, and B. Bodenheimer. Verbs and adverbs: multidimensional motion interpolation. *IEEE Computer Graphics and Applications*, 18(5):32–40, 1998. 2

[56] Yonatan Shafir, Guy Tevet, Roy Kapon, and Amit H Bermano. Human motion diffusion as a generative prior. *arXiv preprint arXiv:2303.01418*, 2023. 3

[57] Li Siyao, Weijiang Yu, Tianpei Gu, Chunze Lin, Quan Wang, Chen Qian, Chen Change Loy, and Ziwei Liu. Bailando: 3d dance generation by actor-critic gpt with choreographic memory. In *Proceedings of the IEEE/CVF Conference on Com-*

puter Vision and Pattern Recognition, pages 11050–11059, 2022. 2

[58] Jiaming Song, Chenlin Meng, and Stefano Ermon. Denoising diffusion implicit models. *arXiv:2010.02502*, 2020. 6

[59] Justin Studer, Dhruv Agrawal, Dominik Borer, Seyedmorteza Sadat, Robert W. Sumner, Martin Guay, and Jakob Buhmann. Factorized motion diffusion for precise and character-agnostic motion inbetweening. In *Proceedings of the 17th ACM SIGGRAPH Conference on Motion, Interaction, and Games*, New York, NY, USA, 2024. Association for Computing Machinery. 3

[60] Kewei Sui, Anindita Ghosh, Inwoo Hwang, Jian Wang, and Chuan Guo. A survey on human interaction motion generation, 2025. 2

[61] Omid Taheri, Nima Ghorbani, Michael J Black, and Dimitrios Tzionas. Grab: A dataset of whole-body human grasping of objects. In *Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part IV 16*, pages 581–600. Springer, 2020. 2

[62] Guy Tevet, Sigal Raab, Brian Gordon, Yoni Shafir, Daniel Cohen-or, and Amit Haim Bermano. Human motion diffusion model. In *The Eleventh International Conference on Learning Representations*, 2023. 2, 3

[63] Jonathan Tseng, Rodrigo Castellon, and C Karen Liu. Edge: Editable dance generation from music. *arXiv preprint arXiv:2211.10658*, 2022. 2

[64] Weilin Wan, Zhiyang Dou, Taku Komura, Wenping Wang, Dinesh Jayaraman, and Lingjie Liu. Tlcontrol: Trajectory and language control for human motion synthesis. *arXiv preprint arXiv:2311.17135*, 2023. 3, 6, 7

[65] Haoru Wang, Wentao Zhu, Luyi Miao, Yishu Xu, Feng Gao, Qi Tian, and Yizhou Wang. Aligning motion generation with human perceptions. In *International Conference on Learning Representations (ICLR)*, 2025. 7

[66] Zan Wang, Yixin Chen, Tengyu Liu, Yixin Zhu, Wei Liang, and Siyuan Huang. Humanise: Language-conditioned human motion generation in 3d scenes. In *Advances in Neural Information Processing Systems (NeurIPS)*, 2022. 2

[67] Zan Wang, Yixin Chen, Baoxiong Jia, Puhao Li, Jinlu Zhang, Jingze Zhang, Tengyu Liu, Yixin Zhu, Wei Liang, and Siyuan Huang. 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. 2

[68] Yiming Xie, Varun Jampani, Lei Zhong, Deqing Sun, and Huaizu Jiang. Omnicontrol: Control any joint at any time for human motion generation. *arXiv preprint arXiv:2310.08580*, 2023. 3, 6, 7

[69] Liang Xu, Ziyang Song, Dongliang Wang, Jing Su, Zhicheng Fang, Chenjing Ding, Weihao Gan, Yichao Yan, Xin Jin, Xiaokang Yang, Wenjun Zeng, and Wei Wu. Actformer: A gan-based transformer towards general action-conditioned 3d human motion generation, 2022. 2

[70] Sirui Xu, Zhengyuan Li, Yu-Xiong Wang, and Liang-Yan Gui. Interdiff: Generating 3d human-object interactions with physics-informed diffusion. In *ICCV*, 2023. 2

[71] Ming Yan, Xin Wang, Yudi Dai, Siqi Shen, Chenglu Wen, Lan Xu, Yuexin Ma, and Cheng Wang. Cimi4d: A large multimodal climbing motion dataset under human-scene interactions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12977–12988, 2023. 2, 6

[72] Hongwei Yi, Justus Thies, Michael J. Black, Xue Bin Peng, and Davis Rempe. Generating human interaction motions in scenes with text control. *arXiv:2404.10685*, 2024. 2

[73] Ye Yuan, Jiaming Song, Umar Iqbal, Arash Vahdat, and Jan Kautz. PhysDiff: Physics-guided human motion diffusion model. In *IEEE International Conference on Computer Vision (ICCV)*, 2023. 2

[74] Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Shaoli Huang, Yong Zhang, Hongwei Zhao, Hongtao Lu, and Xi Shen. T2m-gpt: Generating human motion from textual descriptions with discrete representations. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[75] Lvmin Zhang, Anyi Rao, and Maneesh Agrawala. Adding conditional control to text-to-image diffusion models. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 3836–3847, 2023. 3

[76] Mingyuan Zhang, Zhongang Cai, Liang Pan, Fangzhou Hong, Xinying Guo, Lei Yang, and Ziwei Liu. Motiondiffuse: Text-driven human motion generation with diffusion model. *arXiv preprint arXiv:2208.15001*, 2022. 2

[77] Xinyi Zhang and Michiel Van De Panne. Data-driven autocompletion for keyframe animation. In *Proceedings of the 11th ACM SIGGRAPH Conference on Motion, Interaction and Games*, pages 1–11, 2018. 2

[78] Xiaohan Zhang, Bharat Lal Bhatnagar, Sebastian Starke, Vladimir Guzov, and Gerard Pons-Moll. Couch: Towards controllable human-chair interactions. 2022. 2, 6

[79] Kaifeng Zhao, Shaofei Wang, Yan Zhang, Thabo Beeler, , and Siyu Tang. Compositional human-scene interaction synthesis with semantic control. In *European conference on computer vision (ECCV)*, 2022. 2

[80] Kaifeng Zhao, Yan Zhang, Shaofei Wang, Thabo Beeler, , and Siyu Tang. Synthesizing diverse human motions in 3d indoor scenes. In *International conference on computer vision (ICCV)*, 2023. 2