Autonomous Character-Scene Interaction Synthesis from Text Instruction

NAN JIANG[*], Institute for AI, Peking University, China and National Key Lab of General AI, BIGAI, China
ZIMO HE[*], Institute for AI, Peking University, China
ZI WANG, Beijing University of Posts and Telecommunications, China and National Key Lab of General AI, BIGAI, China
HONGJIE LI, Institute for AI, Peking University, China
YIXIN CHEN, National Key Lab of General AI, BIGAI, China
SIYUAN HUANG[†], National Key Lab of General AI, BIGAI, China
YIXIN ZHU[†], Institute for AI, Peking University, China

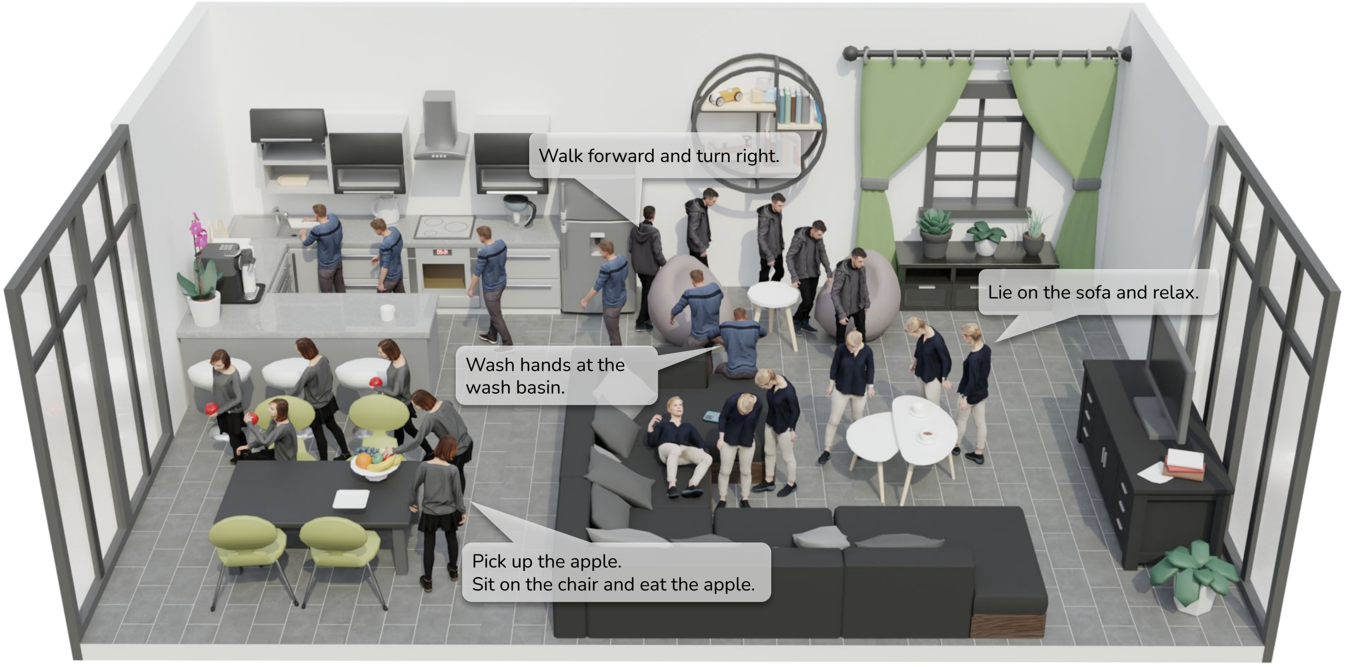

Walk forward and turn right.
Lie on the sofa and relax.
Wash hands at the wash basin.
Pick up the apple.
Sit on the chair and eat the apple.

Fig. 1. **Autonomous HSI synthesis**. Our proposed method generates realistic character motion in 3D scenes based on a single textual instruction and goal location, incorporating seamless transitions between locomotion and HOI autonomously.

Synthesizing human motions in 3D environments, particularly those with complex activities such as locomotion, hand-reaching, and Human-Object Interaction (HOI), presents substantial demands for user-defined waypoints and stage transitions. These requirements pose challenges for current models, leading to a notable gap in automating the animation of characters from simple human inputs. This paper addresses this challenge by introducing a comprehensive framework for synthesizing multi-stage scene-aware interaction motions directly from a single text instruction and goal location. Our approach employs an auto-regressive diffusion model to synthesize the next motion segment, along with an autonomous scheduler predicting the transition for each action stage. To ensure that the synthesized motions are seamlessly integrated within the environment, we propose a scene representation that considers the local perception both at the start and the goal location. We further enhance the coherence of the generated motion by integrating frame embeddings with language input. Additionally, to support model training, we present a comprehensive motion-captured (MoCap)

[*] Both authors contributed equally to this research.
[†] Corresponding authors

Authors' addresses: Nan Jiang, Institute for AI, Peking University, China and National Key Lab of General AI, BIGAI, China, nan.jiang@stu.pku.edu.cn; Zimo He, Institute for AI, Peking University, China, milleret@stu.pku.edu.cn; Zi Wang, Beijing University of Posts and Telecommunications, China and National Key Lab of General AI, BIGAI, China, wangzi1@bupt.edu.cn; Hongjie Li, Institute for AI, Peking University, China, lihongjie@stu.pku.edu.cn; Yixin Chen, National Key Lab of General AI, BIGAI, China, ethanchen@g.ucla.edu; Siyuan Huang, National Key Lab of General AI, BIGAI, China, huangsiyuan@ucla.edu; Yixin Zhu, Institute for AI, Peking University, China, yixin.zhu@pku.edu.cn.

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. Copyrights for components of this work owned by others than the author(s) must be honored. Abstracting with credit is permitted. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee. Request permissions from permissions@acm.org.

SA Conference Papers '24, December 3–6, 2024, Tokyo, Japan

© 2024 Copyright held by the owner/author(s). Publication rights licensed to ACM.

ACM ISBN 979-8-4007-1131-2/24/12
https://doi.org/10.1145/3680528.3687595

dataset comprising 16 hours of motion sequences in 120 indoor scenes covering 40 types of motions, each annotated with precise language descriptions. Experimental results demonstrate the efficacy of our method in generating high-quality, multi-stage motions closely aligned with environmental and textual conditions. Project page: https://lingomotions.com

Additional Key Words and Phrases: Character-scene interaction, motion synthesis, scene representation

ACM Reference Format:
Nan Jiang, Zimo He, Zi Wang, Hongjie Li, Yixin Chen, Siyuan Huang, and Yixin Zhu. 2024. Autonomous Character-Scene Interaction Synthesis from Text Instruction. In *SIGGRAPH Asia 2024 Conference Papers (SA Conference Papers '24), December 3–6, 2024, Tokyo, Japan.* ACM, New York, NY, USA, 16 pages. https://doi.org/10.1145/3680528.3687595

1 INTRODUCTION

Language-guided character motion synthesis within dynamic 3D environments presents a profound challenge in addressing the complexity of multi-stage interactions such as locomotion, hand reaching, and HOI. Unlike humans, who effortlessly interpret and respond to verbal instructions in varied contexts, current motion synthesis methods often fall short of replicating this intuitive capability. Despite recent advances in motion synthesis methods, several challenges remain unaddressed.

Firstly, a primary obstacle in this field is the absence of a unified framework that integrates the various stages of Human-Scene Interactions (HSIs) into a single pipeline. Instead, different stages of motion synthesis, such as walking, reaching, or interacting with objects, are often modeled by separate specialized systems [Hassan et al. 2023; Liu and Hodgins 2017, 2018; Merel et al. 2020]. This fragmented approach results in a lack of coherence in the synthesized motions, making it difficult to sustain long-term interaction sequences that are contextually aligned to the input. Secondly, another significant gap lies in the complexity of the input data required. Some recent works rely on additional inputs that specify the trajectories of objects [Li et al. 2023a], keypoints of humans [Jiang et al. 2024], or motion phase labelling [Starke et al. 2019]. These dependencies constraint the flexibility of the methods and limit their practical deployment. Finally, the absence of scene-level, well-annotated datasets with text labels poses another critical challenge. Although several datasets [Bhatnagar et al. 2022; Hassan et al. 2019; Jiang et al. 2023; Taheri et al. 2020] have recently been proposed, they have failed to cover all spectrums of character-scene interaction, such as grasping, context-rich HOI, and complex scene constraints. Such datasets are essential for developing models that accurately interpret and execute comprehensive motion instructions.

In this paper, we tackle the intricate task of language-guided synthesis of multistage HSIs, directly confronting the challenges outlined earlier. We aim to reduce the dependency on extra, user-provided input data, aiming to autonomously synthesize motion from a single text instruction and goal location as control signals. Furthermore, we integrate the disjointed processes of locomotion, hand reaching, and HOI into a seamless pipeline. This unified approach ensures not only the realism of the motion synthesis but also its functional coherence.

To address the challenge of synthesizing multi-stage HSIs, our method employs an auto-regressive diffusion model designed to generate subsequent motion segments and an autonomous scheduler predicting the transition timing between stages. Recognizing the need for minimal user inputs, which consist only of a text description and a goal location, we leverage the stage-specific goal encoder to produce appropriate condition terms relative to the goal of the current segment. More specifically, we introduce a dual voxel scene encoder that captures detailed context from both the starting and target positions of the motion segment. For precise and time-specific semantic guidance, we propose integrating the time frame embedding with language embeddings to process textual inputs. This dual representation strategy ensures that the synthesized motions are contextually accurate and highly integrated with the surrounding environment, enhancing realism and applicability.

To bridge the gap between scene, motion, and language, we present LINGO (Language-annotated INteraction, Grasping, and lOcomotion), a comprehensive MoCap dataset that unifies HSI activities. LINGO exceeds prior efforts by alleviating the significant labor in real-world motion capturing through a VR-assisted setup, where the vision of synthetic scenes is projected into a VR headset worn by the motion actor. LINGO provides fully text-annotated, long-term human motions and dynamic object interactions of 16 hours within 120 diverse scenes spanning 40 types of diverse motions.

The primary contributions of this work are threefold. First, we propose a framework synthesizing multi-stage scene-aware human motions autonomously and directly from the text instruction and goal location. Second, our model combines the auto-regressive diffusion model with a novel 3D scene representation and a joint time frame and language embedding, achieving seamless integration between human motions and 3D physical environments. Third, we introduce a comprehensive language-annotated MoCap dataset featuring HSI motions.

2 RELATED WORK

The field of HSI synthesis has seen significant progress in recent years, with two main research directions: interacting with static scenes and dynamic objects.

2.1 Interaction Synthesis in Static Scenes

Regarding the static scenes, early work in this area primarily focused on synthesizing single-frame human poses given a 3D scene configuration [Li et al. 2019; Zhang et al. 2020b,a]. Zhao et al. [2022] leveraged contact priors to generate human poses following text conditions. For motion synthesis with static objects, most works have focused on producing motions of locomotion, sitting, and lying with large furniture from pre-defined milestones [Jiang et al. 2024; Wang et al. 2021a,b; Zhao et al. 2023], or tackling short-term motion generation [Huang et al. 2023; Wang et al. 2022]. Reinforcement learning-based methods have also been developed. Zhang and Tang [2022] employs generative motion primitives and a policy network that enables goal-oriented locomotion. Zhao et al. [2023] aims to allow digital humans to perform interactive actions such as sitting on a chair or lying on a sofa. UniHSI [Xiao et al. 2024] introduces a unified framework for multiple types of HSI. However, the framework's reliance on a contact-based driver guided by a large language model limits its applicability. HUMANISE [Wang et al. 2022] was the first to explore text-conditioned scene interactions but relied entirely on short synthetic sequences for training, which was expanded by Yi et al. [2024]. GOAL [Taheri et al. 2022a] and SAGA [Wu et al. 2022] generate full-body poses that aim to reach a specific object. However, these models often specialize in limited interaction types, failing to capture detailed semantics from the instruction and geometric constraints of 3D scenes. This limits their applicability for animating realistic characters for desired interactions.

2.2 Interaction Synthesis for Dynamic Objects

Another line of work focuses on synthesizing human motion with dynamically involved scene objects. Reinforcement learning has been widely used to learn different skills. Early work simplified the object manipulation problem by explicitly attaching an object to the character's hands [Coros et al. 2010; Mordatch et al. 2012; Peng et al. 2019], avoiding the grasping process. Other methods have been proposed to train task-specific policies, such as basketball dribbling [Liu and Hodgins 2018], skateboarding [Liu and Hodgins 2017], and box manipulation [Hassan et al. 2023; Merel et al. 2020]. Lee and Joo [2023] built a unified framework to encompass a series of everyday motions, including locomotion, scene interaction, and object manipulation, similar to our goal. However, it is driven by designated interaction cues, limiting motion diversity and flexibility. Some work tackles interaction motion synthesis without physical simulators[Cui et al. 2024; Starke et al. 2019]. Starke et al. [2019] automates character movements and interactions with objects and uses neural networks to update the state of action dynamically. However, the method faces difficulty in generalizing to acyclic motions. IMoS [Ghosh et al. 2023] synthesizes human and object motions simultaneously after grasping an object, and Li et al. [2023b] proposed a framework that synthesizes human motion given object pose trajectories. The most similar works to ours are TRUMANS [Jiang et al. 2024], which is based on frame-wise action labels rather than text instructions, and Li et al. [2023a], which neglects fine-grained scene geometrical constraints. Furthermore, both require predefined waypoints for humans or objects, making them incapable of being applied to an autonomous digital character.

2.3 Character-scene Interaction Synthesis Datasets

HSI datasets can be broadly classified into two categories: those that focus on human interactions with objects [Bhatnagar et al. 2022; Jiang et al. 2023; Taheri et al. 2020] and those that capture human motion within a scene [Hassan et al. 2019; Jiang et al. 2024; Monszpart et al. 2019; Savva et al. 2016].

A small number of datasets have been developed to capture human manipulation of small objects [Fan et al. 2023; Taheri et al. 2020] or interactions with large objects [Bhatnagar et al. 2022; Jiang et al. 2023; Zhang et al. 2022]. The BEHAVE dataset [Bhatnagar et al. 2022], for instance, contains long sequences of humans interacting with 20 types of large objects. The GRAB dataset [Taheri et al. 2020], on the other hand, focuses on full-body motions of small object manipulation using hands. The CHAIRS [Jiang et al. 2023] and COUCH [Zhang et al. 2022] datasets specifically target human interactions with sittable objects. However, a common limitation of these datasets is that they only consider the object during the interaction, disregarding the surrounding environment.

Early attempts to record human activities in a scene rely on multi-view camera setups and reconstruct human motion through key-point detection [Hassan et al. 2019; Monszpart et al. 2019; Savva et al. 2016]. These datasets typically feature a limited set of actions, such as sitting, lying, and locomotion. Later, researchers begin to employ MoCap equipment, including IMU-based and optic-based systems like VICON. The SAMP dataset [Hassan et al. 2021], for example, covers various sitting, lying down, walking, and running styles in a scene. Guzov et al. [2023] tracks human movements in large indoor scenes, including interactions with articulated objects. Another notable dataset is CIRCLE [Araújo et al. 2023], which captures reaching motions to specific locations in cluttered scenes. In CIRCLES, the scene is virtually presented to the MoCap actor through a VR headset. TRUMANS [Jiang et al. 2024], a recent dataset, marks an advancement by capturing a diverse range of human activities in indoor scenes with dynamically involved objects. However, TRUMANS only provides frame-wise action labels without contextually separable clips, which restricts its applicability in training text-guided motion synthesis models. To address the shortcomings of existing datasets, we introduce a large-scale dataset by realizing real-world motion capture through a VR-assisted setup. The improved efficiency for deployment enhances the dataset diversity encompassing locomotion in various cluttered scenes, grasping, and HOI. Our dataset is uniquely annotated with detailed action descriptions in natural language.

3 METHOD

In this paper, we present a framework for HSI synthesis that seamlessly integrates locomotion, hand-reaching motion, and HOI into a unified model. Our approach enables users to control a virtual character and execute complex interactions using the goal location and simple text instructions as input, which provide concise commands of "go somewhere" and "do something." Our proposed method automatically generates smooth and realistic human motions to navigate the environment and engage in various actions, eliminating the need for manual animation or separate models for each type of action.

3.1 Data Representation

Formally, our objective is to synthesize a human motion sequence $\mathcal{M}$ of length L, given a 3D scene $\mathcal{S}$, a textual instruction $\mathcal{V}$, and a goal position $\mathcal{G}$. We also consider the dynamic object $\mathbf{O}$ if relevant. In this case, we produce the corresponding object pose sequence $\{\mathcal{P}_i\}_{i=1}^{L}$, which contains location and rotation. Due to the autoregressive generation strategy, all the generated human motion and object poses are based on the previous two frames of human motion $\mathcal{M}_{hist}$.

Human Motion. We represent human motion $\mathcal{M}$ using the parameterized human model SMPL-X [Pavlakos et al. 2019]. The motion is initially generated as body joints locations $\{\mathcal{X}_i\}_{i=1}^{L}$, where $\mathcal{X}_i \in \mathbb{R}^{J\times 3}$ represents the 3D positions of $J = 28$ selected joints.

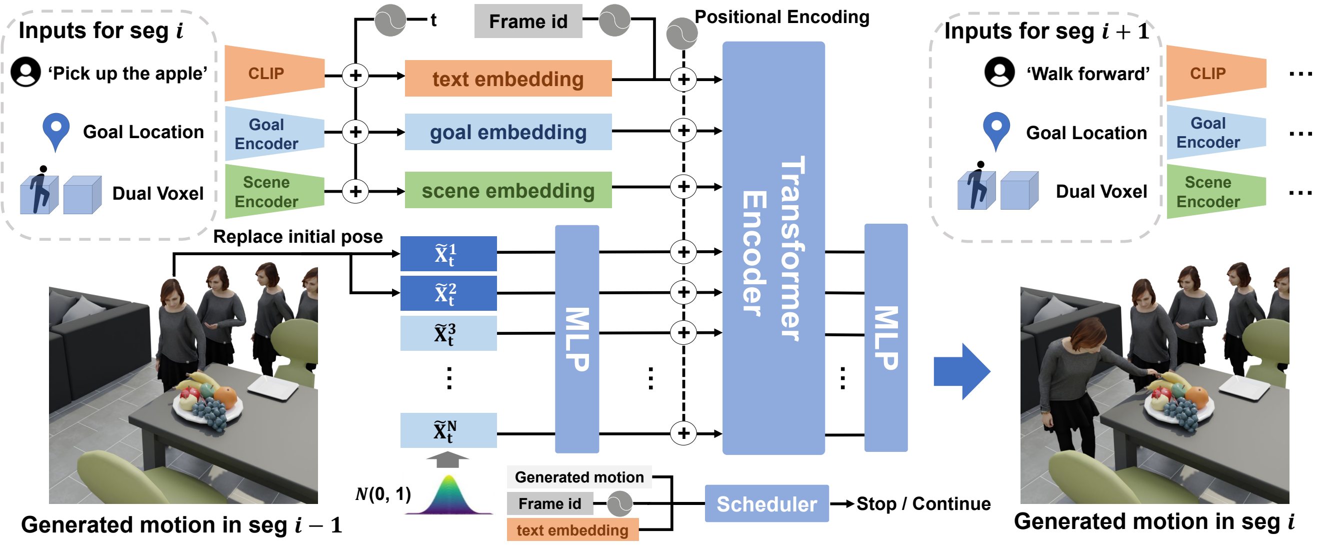


Fig. 2. **Overview of our method**. Our method uses an auto-regressive diffusion model that generates the next motion segment based on existing motions (Section 3.2). The 3D environment is captured through a dual voxel scene encoder (Section 3.3). The text instructions are encoded with the time frame to provide precise and time-specific semantic guidance (Section 3.4). The goal encoder (Section 3.5) embeds the sub-goal locations for different interaction stages, which are automatically determined by our autonomous scheduler (Section 3.6).

These joints include 22 from the body, plus an additional two each for the left hand, right hand, and head, to better capture the rotations at the leaf nodes. The joints are fitted to the SMPL-X pose parameters θ, global orientation ϕ, relaxed hand poses h, and root translation r, resulting in the posed human mesh.

Conditions. The 3D scene is represented using a voxel grid $\mathcal{S} \in \{0, 1\}^{N_x \times N_y \times N_z}$, where 1 indicates that the position is occupied by scene objects or unreachable. When the scene includes dynamic objects, their representations are additionally captured by sampling 256 points from the object's surface along with their corresponding normal vectors facing outwards, denoted as $O \in \mathbb{R}^{256\times 6}$.

The goal location is represented as a 3D location $\mathcal{G} \in \mathbb{R}^3$ specified by the user, where the interpretation of $\mathcal{G}$ differs for locomotion tasks and HOI, discussed in Section 3.5.

3.2 Motion Diffusion Module

The motion diffusion module is responsible for synthesizing human motions based on given text instructions, scene information, and goals. This module leverages the power of diffusion models, specifically the Denoising Diffusion Probabilistic Models (DDPM) [Ho et al. 2020], to generate realistic and coherent human motion sequences. We employ an auto-regressive generation strategy, recently gaining popularity in motion synthesis [Chen et al. 2024]. This strategy allows us to generate motion sequences of arbitrary length by recursively generating motion segments $X = \{X_i\}_{i=1}^W$ of fixed length W.

In the forward diffusion process, random noise is gradually added to the canonicalized input motion data over a series of timesteps. Here, *canonicalized* means the joint locations of the current segment are represented in the coordinates of the character's pelvis bone in the first frame. The reverse diffusion process learns to denoise the corrupted motion data, starting from the pure noise denoted as $\tilde{X}_T$ and progressively removing the noise to recover the original motion.

Following Ho et al. [2020], we train a network ϵ_θ to predict the noise ϵ that was added to the original data at each timestep, based on the noisy motion data $\tilde{X}_t$, the corresponding timestep t, and the condition terms $C = \{\mathcal{S}_{emb}, \mathcal{V}_{emb}, \mathcal{G}_{emb}\}$ introduced in the three sections followed. The learning objective thus follows a simple loss:

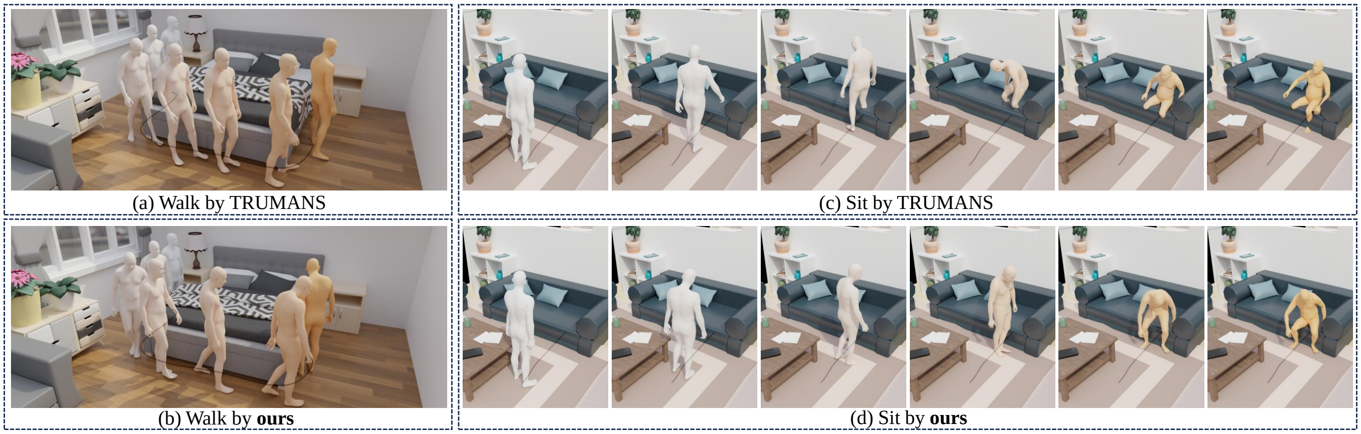
$$\mathcal{L} = \mathbb{E}_{\tilde{X}_t \sim q(\tilde{X}_t|C), t\sim U(1,T)} ||\epsilon - \epsilon_\theta(\tilde{X}_t, t, C)||_2^2. \tag{1}$$

3.3 Dual Voxel Scene Encoder

To achieve responsive motions within the 3D scene, including collision avoidance and natural interactions (*e.g.*, sitting and lying), the character must be aware of not only the immediate local scene but also the forthcoming environment with which it will engage in the near future. To address this challenge, we introduce a dual voxel scene representation.

Our approach builds on the work presented in TRUMANS [Jiang et al. 2024], utilizing a 3D occupancy voxel grid to encapsulate local scene information. The primary mechanism involves constructing a $32 \times 32 \times 32$ grid centered around the location of interest, covering an area where each side spans 1.2 meters. Each cell within this grid is then queried in the scene mesh for occupancy, resulting in a binary 3D array where 1 indicates an occupied cell and 0 indicates an unoccupied one.

Given the dynamic nature of human motion and interaction, our method does not rely on a static scene representation. Instead, it employs a dual voxel system designed to capture both the current and imminent scene contexts that influence the character's movement:



(a) Walk by TRUMANS

(c) Sit by TRUMANS

(b) Walk by **ours**

(d) Sit by **ours**

Fig. 3. **Comparison results.** We qualitatively compare our method with TRUMANS [Jiang et al. 2024]. The left side shows the locomotion along a trajectory, and the right side shows the interaction of sitting on the sofa. Our method generates characters that actively avoid penetrating the scene and exhibit natural cues of scene awareness. For more qualitative results, we refer readers to the supplementary video.

- *Current Scene Voxel*: This voxel grid is centered on the pelvis location of the character at the first frame of the current motion segment, aligning with the character's orientation at that frame.
- *Predictive Scene Voxel*: The position of this voxel depends on the stage of the motion segment. For locomotion, the Predictive Scene Voxel is placed 0.8 meters away in the direction of the goal condition. This voxel is directly aligned with the object's location for interactions with specific scene objects, ensuring that the character's movements are both anticipatory and contextually appropriate. Predictive Scene Voxel does not apply to interactions with hand-held small objects and is copied from Current Scene Voxel. The orientation of the voxel is aligned in a way akin to the Current Scene Voxel.

To extract meaningful features $\mathcal{S}_{emb}$ from the 3D voxel data, we employ a Vision Transformer (ViT) architecture [Dosovitskiy et al. 2021]. The vertical dimension of the voxel grid (height) is considered the channel dimension (akin to color channels in images), and the remaining two dimensions (width and depth) serve as the spatial dimensions (similar to width and height in images).

3.4 Frame-embedded Text Encoder

Unlike traditional text-to-motion methods that directly generate an entire sequence of motion from a sentence, our approach autoregressively generates motion for the next short period without pre-defined timing. This challenges seamlessly linking multiple segments into a full and semantically correct motion based on the text instruction. For example, a target motion clip labeled "read a book" includes the full motion sequence of first turning the page and then reading. To address this, we propose embedding the frame number of action execution into the text during the training and sampling phases.

More specifically, the frame number to be embedded is the number of the first frame in the currently generated motion segment, referenced from the beginning of the original semantically meaningful motion clip. We utilize a sinusoidal positional encoder to convert the frame number from an integer to a 512-dimensional vector, forming the frame embedding. The frame embedding is then added to the embedding of the input textual guidance $\mathcal{V}_{emb} \in \mathbb{R}^{512}$ from the CLIP encoder [Radford et al. 2021]. The summation of these two vectors provides the text conditioning token for the motion synthesis model.

By integrating the frame embedding into the text encoding process, our method learns the temporal patterns of semantic motion, ensuring the generation of coherent and contextually accurate motion segments and effectively linking them into a seamless sequence.

3.5 Stage-specific Goal

The stage-specific goal represents one location $\mathcal{G} \in \mathbb{R}^3$ in the scene as the condition for the current segment. To fit into the autoregressive generation pipeline, $\mathcal{G}$ is represented in the character's pelvis coordinate of the first frame. $\mathcal{G}$ goes through MLPs to obtain the goal embedding $\mathcal{G}_{emb}$ as a condition for the motion diffusion module. The goal here is not forced to reach, unlike TRUMANS, which controls character movement by fixing pelvis or hand joints at specific frames. Our approach leaves more space for the generative model to decide how to reach the goal autonomously.

For locomotion, the goal term is a two-dimensional vector representing the walking direction of the current segment multiplied by a user-specified speed. The direction is derived by sampling a walkable sub-goal near the segment's starting point, which is close to the direction towards the intended goal location. Walkability is determined by checking if the line segment to the sub-goal intersects any unwalkable areas. We neglect the vertical component of $\mathcal{G}$, focusing on horizontal movement. The method can also accommodate predefined trajectories, where the trajectory ahead determines the direction.

In hand grasping interactions, like "pick up the apple," $\mathcal{G}$ corresponds to the position of the index finger. This goal term is similarly applied to put-down actions, setting the goal to the finger's release position. In both settings, the goal term guides the hand towards the target position, and the object is attached to the hand/released at the frame in which the index finger is closest to the target. We apply guidance to the last ten diffusion steps to avoid hand-object penetration, inspired by SceneDiffuser [Huang et al. 2023]. When a penetration occurs, we first identify the closest surface points for each penetrated hand joint. Next, we calculate the average direction of the normal of these surface points as the out-moving direction. To resolve the penetration, we shift the predicted denoised hand joints along this out-moving direction.

For scene-level interactions with specific 3D objects, such as "sit on the sofa," the character receives $\mathcal{G}$ as a precise location indicating the pelvis position for the interaction, like the place to sit. The goal embedding is set to zero for interactions involving small objects, as the body location is not specified during the interaction.

3.6 Autonomous Scheduler

An applicable solution to complex instruction is to leverage foundation models to decompose the instruction into manageable stages, including locomotion and interaction. The autonomous scheduler is trained to determine the optimal stopping point for each stage of the interaction process. The autonomous scheduler decides whether to conclude the current stage and move on to the next based on the latest motion segment, the current frame number, and the given language instruction. Its architecture closely resembles that of the motion diffusion module, leveraging a Transformer Encoder. The frame number and language instruction are embedded in the same way as the Frame-embedded Text Encoder (Section 3.4), which occupies one input token in the Transformer Encoder. Each frame of the encoded motion is represented by a single token that occupies the rest of the tokens. The autonomous scheduler predicts a value between 0 and 1, indicating the likelihood of the current stage concluding at the present motion segment.

4 LINGO DATASET

We propose LINGO, a large-scale MoCap dataset featuring HSIs, including locomotion, grasping, and HOI. The dataset includes various indoor environments, such as bedrooms, dining rooms, offices, and shops, with 3D models sourced from artists on online marketplaces. Skilled actors with motion capture experience performed a variety of interactions, following detailed instructions for common indoor activities like drinking water, playing instruments, using electronic devices, and engaging in physical activities. The LINGO dataset and the code will be made publicly available for research purposes.

4.1 VR-assisted MoCap

Constructing human-scene interaction datasets in the physical world presents two significant challenges: the resource-intensive process of capturing a wide variety of scenes and the difficulty in obtaining accurate pose data to effectively incorporate dynamic objects. To address the challenge, LINGO leverages a "synthetic vision" projected on a VR headset worn by the motion actor, as Figure 4 shows. Building on the concept introduced by the CIRCLE dataset [Araújo et al. 2023], LINGO goes a step further by incorporating interactive objects, both static and dynamic, into captured scenes.

For static objects like chairs and beds, a sittable object with the same height as the virtual object is placed in the physical space, matching the scene in the VR environment. This enables the actor to physically interact with the object while maintaining the illusion of the virtual environment. Capturing dynamic object interactions poses an additional challenge, as the objects must move with the actor's hand during grasping and manipulation. LINGO incorporates a human assistant during the interaction data recording process to overcome this. The assistant precisely marks the frame when the actor grasps the object, at which point the object becomes bound to the actor's hand and follows their movements seamlessly. When the actor releases the object, the assistant marks the moment, unbinding the object from the actor's hand.

We utilize the advanced VICON MoCap system, which results in higher motion data quality than multi-view cameras [Hassan et al. 2019] or IMU-based sensors [von Marcard et al. 2018] commonly used in traditional methods. VR also helps minimize occlusions by avoiding large furniture within the VICON space.

4.2 Statistics and Details

LINGO features 16 hours of motion sequences captured in 120 unique indoor scenes. The dataset covers a wide spectrum of 40 motion types, each accompanied by precise language descriptions for clarity and usability, summarized in Figures 6 and 7. The language descriptions are further augmented for generalizability. The motion sequences are organized into 4 to 6 long-term segments per scene, each spanning 1 to 3 minutes. The human motion is captured at 30 FPS, leveraging a 53-point FrontWaist MoCap suit. The LINGO-train and LINGO-eval sets are split by the scenes, ensuring a 4:1 proportion for each room type without overlap. Accompanying the interactions, the LINGO dataset incorporates 20 types of objects commonly used in everyday life.

To represent human subjects, LINGO employs the SMPL-X model [Pavlakos et al. 2019]. The dataset utilizes the MANO hand model with 12 PCA components for efficient hand representation. Facial expressions are not captured in this dataset. For more detailed information regarding the dataset, such as the specific interaction types, object categories, and scene descriptions, the readers are encouraged to refer to the supplementary material.

5 EXPERIMENTS

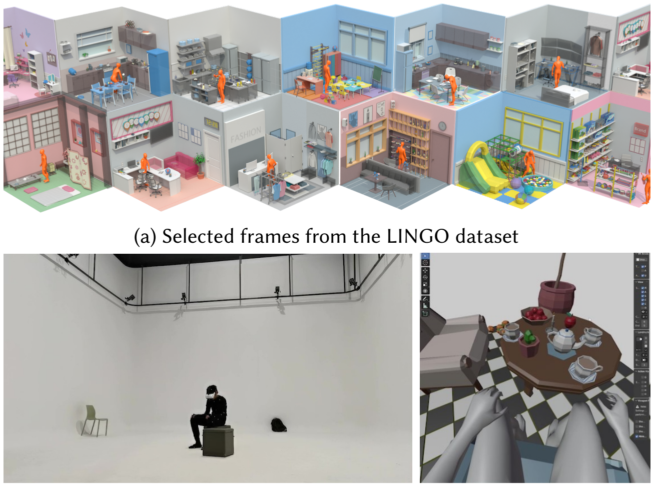
Our experimental evaluation is divided into three primary settings associated with three stages of motion: locomotion, object reaching, and interactive motion. We compare our method with state-of-the-art methods leveraging generative models. All comparison methods are trained in LINGO-train and evaluated in LINGO-eval.

5.1 Evaluation Settings

Locomotion. In the locomotion experiment, we investigate the character's ability to navigate a scene while avoiding collisions and maintaining natural movement. We randomly generate 100 pairs of start and end locations, ensuring that these direct paths intersect with objects in the scene, thereby testing the character's collision avoidance skills in cluttered environments. We compare our method with TRUMANS [Jiang et al. 2024] and include an ablation study to evaluate the significance of dual voxel scene representation. In this ablation case, we flatten the 3D scene voxel along the vertical axis, forming a walkable map where 0 indicates a walkable area

Table 1. **Quantitative results of interactive motion synthesis.** The instructions involve performing interaction with an object in the scene.

Interactive motion	FID↓	Diversity→	Multi-modality→	Precision↑	Recall↑	F1 score↑
Real motions	-	6.379	1.119	0.888	0.907	0.907
TRUMANS	2.438±.041	6.182±.078	3.353±.077	0.628±.004	0.557±.004	0.552±.004
Ours	**2.048**±.058	6.220±.076	**2.919**±.081	**0.695**±.004	**0.629**±.004	**0.622**±.004
w/o frame embedding	2.368±.070	**6.300**±.062	3.600±.072	0.615±.005	0.553±.005	0.540±.006



(a) Selected frames from the LINGO dataset

(b) MoCap setup

(c) View in VR

Fig. 4. **LINGO dataset.** We show some selected frames and the setup of the VR-assisted MoCap.

Table 2. **Quantitative results of locomotion**, where the character walks from one place to another in cluttered scenes.

Locomotion	Pene$_{\%scene}$↓	Pene$_{mean}$↓	Pene$_{max}$↓	FS↓
TRUMANS	0.048±.006	1.011±.012	7.441±.383	0.472±.013
Ours	**0.038**±.001	**0.402**±.004	**0.948**±.065	**0.432**±.004
flattened voxel	0.045±.002	0.587±.005	3.373±.117	0.470±.030

Table 3. **Quantitative results of object reaching**, where the character is instructed to walk toward and reach for an object.

Object reaching	Error dist.↓	Pene$_{\%scene}$↓	Time used↓
GOAL	0.156±.028	0.057±.004	4.880±.356
Ours	**0.061**±.004	**0.045**±.003	**3.073**±.351
w/o dual voxel	0.111±.013	0.057±.008	3.147±.360
w/o frame embedding	0.135±.026	0.046±.003	3.980±.242

and 1 indicates an occupied location. For evaluation, we measure scene penetration using metrics adapted from DIMOS [Zhao et al. 2023], which include the ratio of body vertices that penetrate scene objects (Pene$_{\%scene}$), the average penetration distance throughout the synthesized sequence (Pene$_{mean}$), and the maximum penetration distance in any frame during the sequence (Pene$_{max}$). Furthermore, we assess foot sliding using the metric from [He et al. 2022].

Object Reaching. We evaluate the character's ability to reach for an object while being aware of the surrounding environment. We randomly select 100 pairs of starting positions for the character and goal positions for the object. The evaluation focuses on both the quality of the reaching motion and collision with scenes. We compare the results with GOAL [Taheri et al. 2022b] as a baseline. We include two ablation studies where the frame embedding or dual voxel is removed. Metrics used include the error distance between the intended goal position and the actual hand position after reaching the object, the same scene penetration metrics as in the locomotion setting (Pene$_{\%scene}$), and the time to complete the reaching task (no more than 20 cm between the hand and the object) measured in seconds.

Interactive Motion. In this setting, we generate motions that involve interaction with objects from the LINGO dataset. For small objects like bottles and gamepads, we assume that the objects are already grasped and focus on the semantic quality of the motion. For larger objects such as sofas and beds, the character first walks towards the object and then interacts by sitting or lying down. We compare our method with TRUMANS, using several metrics following [Tevet et al. 2022], including Fréchet Inception Distance (FID) to measure motion realism, Diversity and Multi-modality to evaluate the range of different motions produced, Precision to measure how closely the generated motions match the reference, Recall to determine the coverage of the reference motions by the generated ones, and the F1 score to provide a balanced measure. We modify the action encoder of TRUMANS to use the same language encoder as our method.

5.2 Results

Table 1 presents the quantitative results of semantic motion synthesis, demonstrating that our method achieves the highest scores on all metrics except Diversity. The ablation study, which removes the frame embedding, results in higher Diversity because the synthesized motions become disordered, often repeating the same actions without maintaining a coherent flow. It lacks awareness of the overall motion sequence, which can be observed in Figures 5c and 5d. All other metrics consistently show that the motion synthesized by our method is of high quality and maintains coherence with the given semantics. This indicates that our approach effectively balances realism and semantic alignment, producing natural and contextually correct motions.

Table 2 shows the quantitative results for the locomotion task. Our method outperforms TRUMANS in all evaluated metrics. The scene penetration from our method is significantly lower, highlighting its superior capability to autonomously plan and adjust trajectories to avoid collisions rather than following a predefined path. This autonomous decision-making is crucial for navigating cluttered scenes, where subtle body adjustments are necessary to avoid obstacles, demonstrated in Figures 3, 5a and 5b. Additionally, the foot sliding metric for our method is also low, indicating high-quality motion with stable and realistic movement. From the ablations where the dual voxel representation is replaced with a flattened walkable map, the higher penetration and foot sliding underscores the effectiveness of our dual voxel scene representation in providing comprehensive 3D information about the surroundings, enabling more precise and realistic motion adjustments.

Table 3 confirms the efficiency and robustness of our method in synthesizing realistic hand-oriented motions. Specifically, our method achieves the best scores in terms of low reaching error, low penetration with the scene, and short time used for reaching the goal. The ablations further validate the efficacy of our dual voxel scene encoder and the frame embedding. Tables 2 and 3 jointly demonstrate that our method can generalize to unseen environments. Figures 5c and 5d shows that the frame embedding helps to generate semantically coherent motions.

6 CONCLUSION

Limitations. (i) Our approach concentrates on body-level motions, ignoring the intricate details of hand-level manipulation and facial expressions. (ii) Although quantitative results indicate that our method achieves superior scene awareness compared to existing techniques, it does not guarantee a perfect physical plausibility of the generated motions. (iii) We have not researched the generalizability of unseen interaction types in this work.

In summary, this work introduces a comprehensive generative framework that autonomously synthesizes multi-stage, scene-aware human motions directly from text instructions and goal locations. We present an approach that seamlessly integrates human motions with 3D scenes with a novel 3D scene representation and a joint time frame and language embedding. We also contribute a detailed, language-annotated MoCap dataset, providing a valuable resource for future research in human motion synthesis.

ACKNOWLEDGMENTS

The authors would like to thank NVIDIA for generously providing the necessary GPUs and hardware support. This work is supported in part by the National Science and Technology Major Project (2022ZD0114900), an NSFC fund (62376009), and the Beijing Nova Program.

REFERENCES

Joao Pedro Araújo, Jiaman Li, Karthik Vetrivel, Rishi Agarwal, Jiajun Wu, Deepak Gopinath, Alexander William Clegg, and Karen Liu. 2023. CIRCLE: Capture In Rich Contextual Environments. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Bharat Lal Bhatnagar, Xianghui Xie, Ilya A. Petrov, Cristian Sminchisescu, Christian Theobalt, and Gerard Pons-Moll. 2022. BEHAVE: Dataset and Method for Tracking Human Object Interactions. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Rui Chen, Mingyi Shi, Shaoli Huang, Ping Tan, Taku Komura, and Xuelin Chen. 2024. Taming Diffusion Probabilistic Models for Character Control. In *SIGGRAPH Conference Papers*.

Stelian Coros, Philippe Beaudoin, and Michiel Van de Panne. 2010. Generalized biped walking control. *ACM Transactions on Graphics (TOG)* 29, 4 (2010), 1–9.

Jieming Cui, Tengyu Liu, Nian Liu, Yaodong Yang, Yixin Zhu, and Siyuan Huang. 2024. Anyskill: Learning Open-Vocabulary Physical Skill for Interactive Agents. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, Jakob Uszkoreit, and Neil Houlsby. 2021. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. In *International Conference on Learning Representations (ICLR)*.

Zicong Fan, Omid Taheri, Dimitrios Tzionas, Muhammed Kocabas, Manuel Kaufmann, Michael J. Black, and Otmar Hilliges. 2023. ARCTIC: A Dataset for Dexterous Bimanual Hand-Object Manipulation. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Anindita Ghosh, Rishabh Dabral, Vladislav Golyanik, Christian Theobalt, and Philipp Slusallek. 2023. IMoS: Intent-Driven Full-Body Motion Synthesis for Human-Object Interactions. In *Eurographics*.

Vladimir Guzov, Julian Chibane, Riccardo Marin, Yannan He, Yunus Saracoglu, Torsten Sattler, and Gerard Pons-Moll. 2023. Interaction Replica: Tracking human–object interaction and scene changes from human motion. In *International Conference on 3D Vision (3DV)*.

Mohamed Hassan, Duygu Ceylan, Ruben Villegas, Jun Saito, Jimei Yang, Yi Zhou, and Michael Black. 2021. Stochastic Scene-Aware Motion Prediction. In *International Conference on Computer Vision (ICCV)*.

Mohamed Hassan, Vasileios Choutas, Dimitrios Tzionas, and Michael J Black. 2019. Resolving 3D human pose ambiguities with 3D scene constraints. In *International Conference on Computer Vision (ICCV)*.

Mohamed Hassan, Yunrong Guo, Tingwu Wang, Michael Black, Sanja Fidler, and Xue Bin Peng. 2023. Synthesizing Physical Character-Scene Interactions. In *SIGGRAPH Conference Papers*.

Chengan He, Jun Saito, James Zachary, Holly Rushmeier, and Yi Zhou. 2022. Nemf: Neural motion fields for kinematic animation. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Jonathan Ho, Ajay Jain, and Pieter Abbeel. 2020. Denoising diffusion probabilistic models. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Siyuan Huang, Zan Wang, Puhao Li, Baoxiong Jia, Tengyu Liu, Yixin Zhu, Wei Liang, and Song-Chun Zhu. 2023. Diffusion-based Generation, Optimization, and Planning in 3D Scenes. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Nan Jiang, Tengyu Liu, Zhexuan Cao, Jieming Cui, Zhiyuan Zhang, Yixin Chen, He Wang, Yixin Zhu, and Siyuan Huang. 2023. Full-Body Articulated Human-Object Interaction. In *International Conference on Computer Vision (ICCV)*.

Nan Jiang, Zhiyuan Zhang, Hongjie Li, Xiaoxuan Ma, Zan Wang, Yixin Chen, Tengyu Liu, Yixin Zhu, and Siyuan Huang. 2024. Scaling up dynamic human-scene interaction modeling. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Jiye Lee and Hanbyul Joo. 2023. Locomotion-Action-Manipulation: Synthesizing Human-Scene Interactions in Complex 3D Environments. In *International Conference on Computer Vision (ICCV)*.

Jiaman Li, Alexander Clegg, Roozbeh Mottaghi, Jiajun Wu, Xavier Puig, and C Karen Liu. 2023a. Controllable human-object interaction synthesis. In *European Conference on Computer Vision (ECCV)*.

Jiaman Li, Jiajun Wu, and C Karen Liu. 2023b. Object motion guided human motion synthesis. *ACM Transactions on Graphics (TOG)* 42, 6 (2023), 1–11.

Xueting Li, Sifei Liu, Kihwan Kim, Xiaolong Wang, Ming-Hsuan Yang, and Jan Kautz. 2019. Putting humans in a scene: Learning affordance in 3d indoor environments. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Libin Liu and Jessica Hodgins. 2017. Learning to schedule control fragments for physics-based characters using deep q-learning. *ACM Transactions on Graphics (TOG)* 36, 3 (2017), 1–14.

Libin Liu and Jessica Hodgins. 2018. Learning basketball dribbling skills using trajectory optimization and deep reinforcement learning. *ACM Transactions on Graphics (TOG)* 37, 4 (2018), 1–14.

Josh Merel, Saran Tunyasuvunakool, Arun Ahuja, Yuval Tassa, Leonard Hasenclever, Vu Pham, Tom Erez, Greg Wayne, and Nicolas Heess. 2020. Catch & carry: reusable neural controllers for vision-guided whole-body tasks. *ACM Transactions on Graphics (TOG)* 39, 4 (2020), 39–1.

Aron Monszpart, Paul Guerrero, Duygu Ceylan, Ersin Yumer, and Niloy J Mitra. 2019. iMapper: interaction-guided scene mapping from monocular videos. *ACM Transactions on Graphics (TOG)* 38, 4 (2019), 1–15.

Igor Mordatch, Emanuel Todorov, and Zoran Popović. 2012. Discovery of complex behaviors through contact-invariant optimization. *ACM Transactions on Graphics (TOG)* 31, 4 (2012), 1–8.

Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed AA Osman, Dimitrios Tzionas, and Michael J Black. 2019. Expressive body capture: 3d hands, face, and body from a single image. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Xue Bin Peng, Michael Chang, Grace Zhang, Pieter Abbeel, and Sergey Levine. 2019. Mcp: Learning composable hierarchical control with multiplicative compositional policies. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. 2021. Learning transferable visual models from natural language supervision. In *International Conference on Machine Learning (ICML)*.

Manolis Savva, Angel X Chang, Pat Hanrahan, Matthew Fisher, and Matthias Nießner. 2016. Pigraphs: learning interaction snapshots from observations. *ACM Transactions* *on Graphics (TOG)* 35, 4 (2016), 1–12.

Sebastian Starke, He Zhang, Taku Komura, and Jun Saito. 2019. Neural state machine for character-scene interactions. *ACM Transactions on Graphics (TOG)* 38, 6 (2019), 178.

Omid Taheri, Vasileios Choutas, Michael J Black, and Dimitrios Tzionas. 2022a. GOAL: Generating 4D Whole-Body Motion for Hand-Object Grasping. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Omid Taheri, Vasileios Choutas, Michael J. Black, and Dimitrios Tzionas. 2022b. GOAL: Generating 4D Whole-Body Motion for Hand-Object Grasping. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Omid Taheri, Nima Ghorbani, Michael J Black, and Dimitrios Tzionas. 2020. GRAB: A dataset of whole-body human grasping of objects. In *European Conference on Computer Vision (ECCV)*.

Guy Tevet, Sigal Raab, Brian Gordon, Yoni Shafir, Daniel Cohen-or, and Amit Haim Bermano. 2022. Human Motion Diffusion Model. In *International Conference on Learning Representations (ICLR)*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems (NeurIPS)*.

Timo von Marcard, Roberto Henschel, Michael Black, Bodo Rosenhahn, and Gerard Pons-Moll. 2018. Recovering Accurate 3D Human Pose in The Wild Using IMUs and a Moving Camera. In *European Conference on Computer Vision (ECCV)*.

Jiashun Wang, Huazhe Xu, Jingwei Xu, Sifei Liu, and Xiaolong Wang. 2021a. Synthesizing Long-Term 3D Human Motion and Interaction in 3D Scenes. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Jingbo Wang, Sijie Yan, Bo Dai, and Dahua Lin. 2021b. Scene-aware generative network for human motion synthesis. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

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

Yan Wu, Jiahao Wang, Yan Zhang, Siwei Zhang, Otmar Hilliges, Fisher Yu, and Siyu Tang. 2022. SAGA: Stochastic Whole-Body Grasping with Contact. In *European Conference on Computer Vision (ECCV)*.

Zeqi Xiao, Tai Wang, Jingbo Wang, Jinkun Cao, Wenwei Zhang, Bo Dai, Dahua Lin, and Jiangmiao Pang. 2024. Unified Human-Scene Interaction via Prompted Chain-of-Contacts. In *International Conference on Learning Representations (ICLR)*.

Hongwei Yi, Justus Thies, Michael J Black, Xue Bin Peng, and Davis Rempe. 2024. Generating Human Interaction Motions in Scenes with Text Control. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Siwei Zhang, Yan Zhang, Qianli Ma, Michael J Black, and Siyu Tang. 2020b. Generating person-scene interactions in 3d scenes. In *International Conference on 3D Vision (3DV)*.

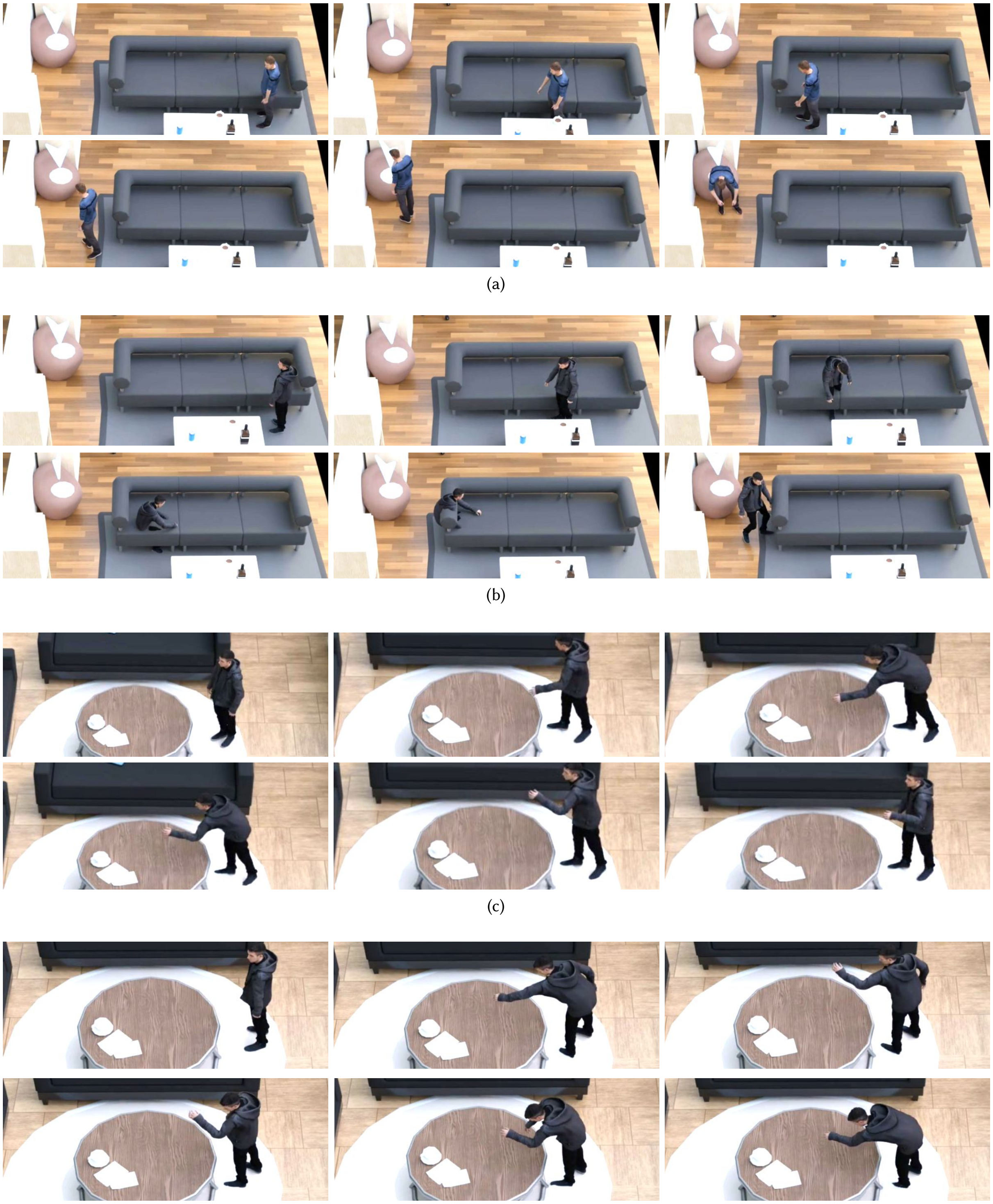
Xiaohan Zhang, Bharat Lal Bhatnagar, Sebastian Starke, Vladimir Guzov, and Gerard Pons-Moll. 2022. Couch: Towards controllable human-chair interactions. In *European Conference on Computer Vision (ECCV)*.

Yan Zhang, Mohamed Hassan, Heiko Neumann, Michael J Black, and Siyu Tang. 2020a. Generating 3d people in scenes without people. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Yan Zhang and Siyu Tang. 2022. The Wanderings of Odysseus in 3D Scenes. In *Conference on Computer Vision and Pattern Recognition (CVPR)*.

Kaifeng Zhao, Shaofei Wang, Yan Zhang, Thabo Beeler, and Siyu Tang. 2022. Compositional human-scene interaction synthesis with semantic control. In *European Conference on Computer Vision (ECCV)*.

Kaifeng Zhao, Yan Zhang, Shaofei Wang, Thabo Beeler, and Siyu Tang. 2023. Synthesizing Diverse Human Motions in 3D Indoor Scenes. In *International Conference on Computer Vision (ICCV)*.


(a)

(b)

(c)

(d)

Fig. 5. **Qualitative comparison.** We compare (a) our method with (b) TRUMANS [Jiang et al. 2024] on the task of walking to the goal location. It is shown that our method is aware of the surroundings for collision avoidance, while TRUMANS depends on a pre-defined trajectory. We show (c) our method and (d) w/o frame embedder given “grasp an object” instruction. The synthesized motion without a frame embedder is disordered and tends to repeat.

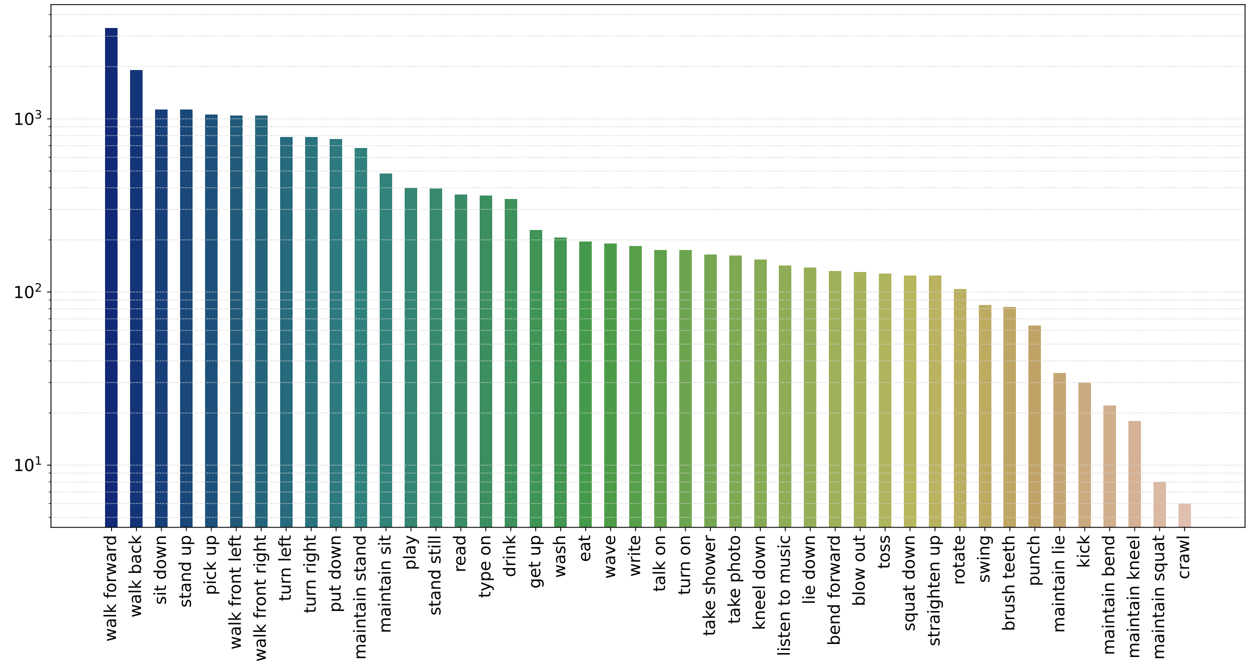


Fig. 6. **Number of occurrences of each motion type in LINGO dataset.**

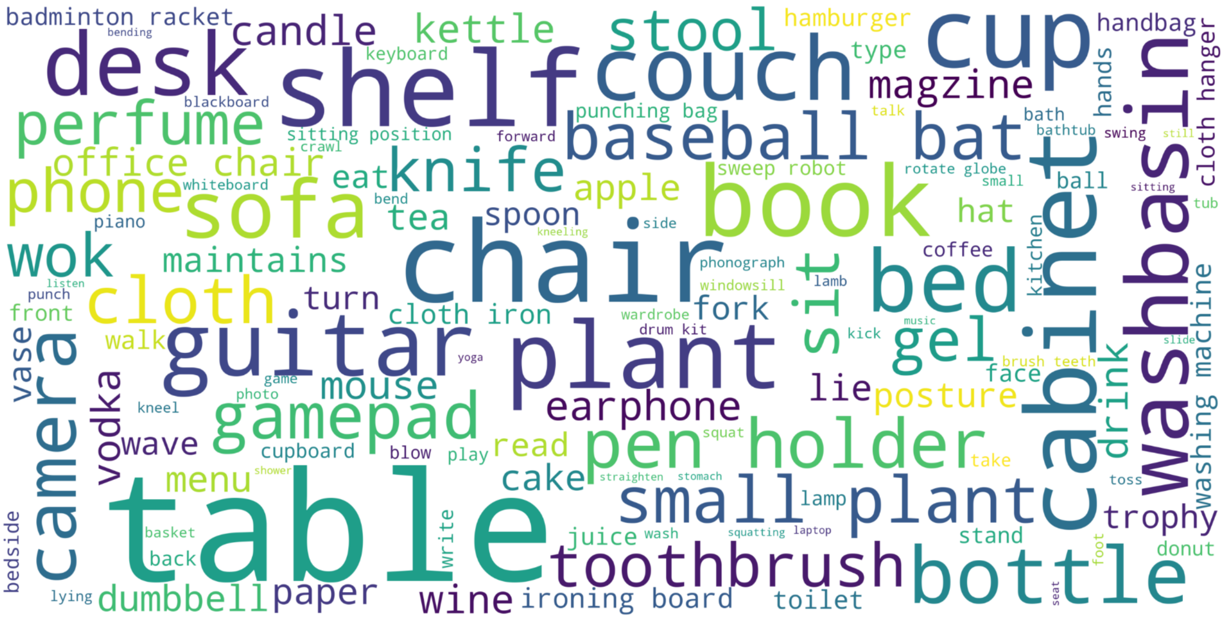


Fig. 7. **Word cloud built from the language annotations in the LINGO dataset.**

A IMPLEMENTATION DETAILS

In this section, we describe the detailed architecture of each module in our framework, along with the training configurations.

A.1 Motion Diffusion Module

The motion diffusion module employs a Transformer encoder architecture [Vaswani et al. 2017] with 8 layers and 16 attention heads, which has proven highly effective in modeling sequential data. The input to the model consists of tokens representing the noised body joints and additional tokens for the condition information. To achieve auto-regressive generation, we fix the first two tokens of the current segment, copy the value of the last two frames from the last segment, and zero out the noise applied to them, during both training and sampling. In addition to the body joint tokens, four other tokens are introduced to incorporate the conditions, representing the scene, text, pelvis, and hand goal location, which provide crucial context for generating coherent and relevant motions. The details of these conditioning tokens will be discussed in the following three subsections. An embedding of diffusion timestep is added to the four conditioning tokens to incorporate temporal information into the model. All tokens undergo a positional encoding before being fed into the Transformer model.

A.2 Scene Encoder

The current scene voxel and its predictive counterpart are concatenated along the channel dimension, creating a unified 64-channel, 32x32 image. This image is segmented into 8x8 patches, which serve as input for a ViT [Dosovitskiy et al. 2021] consisting of 6 layers and 16 attention heads. The ViT processes these patches and produces a 512-dimensional feature vector. This vector is then used as the scene conditioning token in the motion diffusion module, ensuring context-aware motion synthesis.

When the target action involves interactions unrelated to the scene, such as drinking water from a bottle or talking on the phone, we mask the scene conditional token as all zeros. This approach is intended to prevent interference from scene information during the generation process of scene-independent interactions.

A.3 Frame-embedded Text Encoder

We employ the CLIP encoder [Radford et al. 2021] to convert raw text descriptions into 768-dimensional latent vectors. These vectors are then transformed into 512-dimensional vectors using an MLP model. Simultaneously, a sinusoidal positional encoder converts the frame number from an integer to a 512-dimensional vector, forming the frame embedding. We then add the text embedding to the frame embedding and pass the result through another MLP layer to obtain the final text conditional token, the input for our motion diffusion module.

During the sampling phase, at each step of autoregressive generation, the frame number input to the model increases as the number of generated frames increases. This aligns with the increased rate during training. In particular, by controlling the rate at which the frame number increases, we can adjust the total duration of the generated action. Due to the periodic nature of locomotion, such as walking, we set the frame number to zero during both the training and sampling processes for locomotion.

A.4 Goal Encoder

We train separate MLPs to embed locomotion and hand goals, resulting in embeddings for pelvic and hand goals, respectively. For locomotion tasks, we remove the vertical component of the goal location and retain only the two-dimensional horizontal coordinates as input for the model. In object-reaching tasks, the target coordinates of the hand serve as input for the model. We mask the pelvis or hand goal tokens as all zeros for actions that do not involve locomotion or hand reaching.

A.5 Autonomous Scheduler

Our scheduler model generates a value ranging from 0 to 1, which indicates whether the previously generated motion clip has completed its entire semantic motion. This value subsequently determines whether the current motion clip should maintain the existing semantic or initiate the first motion clip of a new semantic. Our Scheduler model utilizes a Transformer encoder with 3 layers, 8 attention heads, and a hidden dimension of 512. We leverage a model identical to the structure described in Section 3.4 to embed the current frame number and the given language instruction as a text conditioning token. This token, along with other motion frames, serves as the input to the Transformer encoder.

Due to the simplicity of this task, we train the scheduler model on the entire LINGO dataset for only 5 epochs. We use a batch size of 1024 and a learning rate of 0.0001, employing the Adam optimizer with its default parameter settings. The model converges effectively and demonstrates strong performance.

A.6 Training Configuration

The training of our motion diffusion module is conducted with a carefully selected set of hyperparameters to ensure optimal performance. We utilize a learning rate of 0.0001. The number of diffusion timesteps is fixed at 100, balancing computational feasibility and the quality of generated samples. In addition, we adopt a linear noise schedule, which gradually increases noise levels throughout the diffusion process. We use 4 NVIDIA A100 GPUs to train over 500 epochs with a batch size of 1024, ensuring sufficient exposure to the training data for convergence. These hyperparameter choices are informed by prior literature and empirical experimentation.

B LINGO DATASET

In this section, we elaborate on the recording process of the LINGO dataset and statistics of the LINGO dataset in detail.

B.1 How LINGO Dataset Is Produced

Producing a VR-assisted motion-captured dataset is a complex process that involves multiple people, specialized equipment, and custom software. In this section, we provide an overview of the key components and steps involved in this process.

B.1.1 People Participants. The MoCap process involves three main roles.

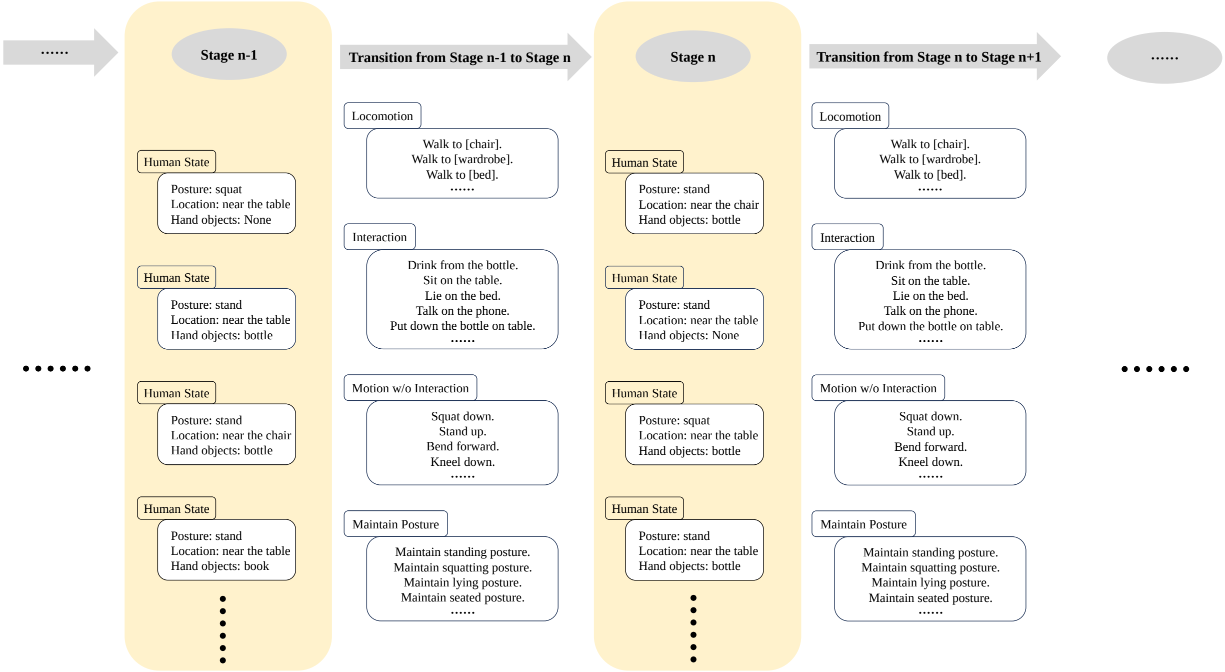


Fig. A1. **Motion Planner.** A Markov Chain generates the next instruction to guarantee plausible interaction and maintain a balanced distribution of motion types. The Motion Planner provides language instructions to the Actor.

Actor. The Actor performs the motions while wearing a MoCap suit and a VR headset.

Controller. The Controller provides the Actor with language descriptions of the motions to perform.

Assistant. The Assistant marks the frames when the Actor picks up or puts down hand-held objects, ensuring accurate synchronization between the motion data and the object interactions.

B.1.2 MoCap Add-on.

We designed a custom Blender add-on to facilitate the MoCap process. This addon has three main functions. First, it displays the live motion of the Actor in the physical environment, superimposed on the virtual scene. This allows the production team to identify and correct errors such as penetration between the Actor's virtual representation and the scene objects or erroneous motion capture data. Second, the addon shows the Actor's VR headset view, which helps the team adjust the motion capture setup to suit the current motion type and Actor best. Third, the add-on provides a third-person view that follows the Actor's movement, similar to a third-person video game camera. This helps the Controller give orientation-related locomotion instructions, such as "walk to the right." The addon also links the VR and VICON systems, projecting rendered views and language instructions to the Actor and repeatedly aligning the real-world and virtual-world coordinates to maintain synchronization.

B.1.3 Motion Planner Add-on.

Another custom addon, the Motion Planner (Figure A1), generates a sequence of instructions for the Actor to perform in the current scene as a Markov Chain. The input to the Motion Planner is a list of candidate interactions, their properties, and constraints. For example, some interactions may require the use of one or both hands or may have specific starting or ending positions. The Motion Planner considers these constraints and outputs a sequence of motion instructions that satisfy them. The Motion Planner also helps maintain a balanced distribution in the dataset. The Controller advances the instructions displayed in the Actor's VR headset by pressing a "go to next instruction" button.

B.1.4 Preparation.

Before starting the MoCap process, the scene files are prepared in Blender. This involves selecting scene objects for the Actor to interact with and adding small interactable objects as needed. For sittable objects such as sofas and chairs, placeholders are placed in the physical environment to support the Actor during the capture session. The interaction types for each scene are specified and input to the Motion Planner. The VICON system is warmed up and calibrated to ensure accurate tracking. The VR headset is initialized by aligning the virtual world with the real world using a calibration procedure.

B.1.5 Motion Capturing.

During the MoCap session, the Actor stands in a plausible location within the scene. The Controller checks that everything is ready and displays the first motion instruction in the Actor's headset. For interactive motions, such as picking up or manipulating objects, the Controller determines when the interaction is finished and advances to the next instruction. For grasping motions, the Assistant marks the time frames when the Actor grasps or releases the object, ensuring that these critical events are accurately recorded in the dataset and the object is attached to hands correctly. For locomotion, the Controller guides the Actor to the goal location using arrow keys, with specific direction-related information projected in the VR headset and recorded as annotations for the LINGO dataset.

During the MoCap process, the Actor performs the actions based on the language instructions projected on their VR headset. The VICON system tracks and records their body movements in real-time as the Actor moves and interacts within the physical space. Simultaneously, the captured motion data is instantly transmitted to the virtual scene, where the Actor's virtual body is rendered in real-time. This real-time reconstruction allows the Actor to see their virtual representation within the VR headset, creating a highly immersive and interactive experience.

B.1.6 Data Post-processing. After the MoCap session, the raw motion capture data is split into segments according to motion types. The Motion Planner Add-on accompanies each segment with a raw text annotation describing the performed motion. To enhance the richness and variety of annotations, GPT-4 is used to augment the raw text into multiple versions, providing alternative descriptions and additional context. We also double the data size by mirroring both the motions and the annotations.

B.2 Dataset Statistics

B.2.1 Interaction Types. The LINGO dataset covers 40 types of motion listed in Table A1, including non-interactive motions such as locomotion and maintaining posture. For interactive motions, the LINGO dataset contains interaction with static scene objects (*e.g.*, sit down, lie down) and small hand-held objects (*e.g.*, cellphone, gamepad). Figure 6 counts the number of occurrences of each motion type, and Figure A2 shows the distribution of the motion length. The detailed categorization is listed in Table A1.

B.2.2 Locomotion Goal Distribution. In the analysis of locomotion clips, we visualize the distribution of goal locations in Figure A3 represented in the canonical coordinate system of the first frame. This coordinate system is defined to align the character's initial orientation with the positive yaw direction. Using this consistent frame of reference, we can compare and study the relative positions of the goal locations across different clips. The plot represents the spatial distribution of the goal locations, with the origin (0, 0) corresponding to the character's starting position in the first frame. The plot's x-axis and y-axis represent the lateral and forward/backward directions, respectively, relative to the character's initial orientation.

B.2.3 Motion Length Distribution. Figure A2 presents the motion length distribution for various motion types in the LINGO dataset. Each violin represents a motion type, with the width of the violin indicating the density of data points at different motion lengths. The vertical axis measures the motion length, while the horizontal axis lists the motion types.

Table A1. **Motion types of LINGO.**

Motion Description	Motion Name
Move from one place to another by taking steps.	walk forward walk back walk front left walk front right
Change the orientation of the body.	turn left turn right
Change to a standing position.	stand up get up
Interact with hand-held objects.	pick up put down take photo turn on write type on read play[1] drink eat talk on listen to music brush teeth toss swing wave
Pick up and put down hand-held objects.	pick up put down
Stationary motions.	stand still maintain lie maintain sit maintain bend maintain kneel maintain squat
In-place motions.	bend forward straighten up kneel down squat down crawl
Interact with static scene objects.	sit down lie down punch kick wash take shower rotate play[2] type

[1] Play game and guitar. [2] Play drums and piano.

The motion lengths span from 1 to 12 seconds across all motion types. However, most of the data points lie between 2 and 6 seconds. The median motion length for most motion types is around 4-5 seconds. Some motions, like "walk forward," are close to a normal

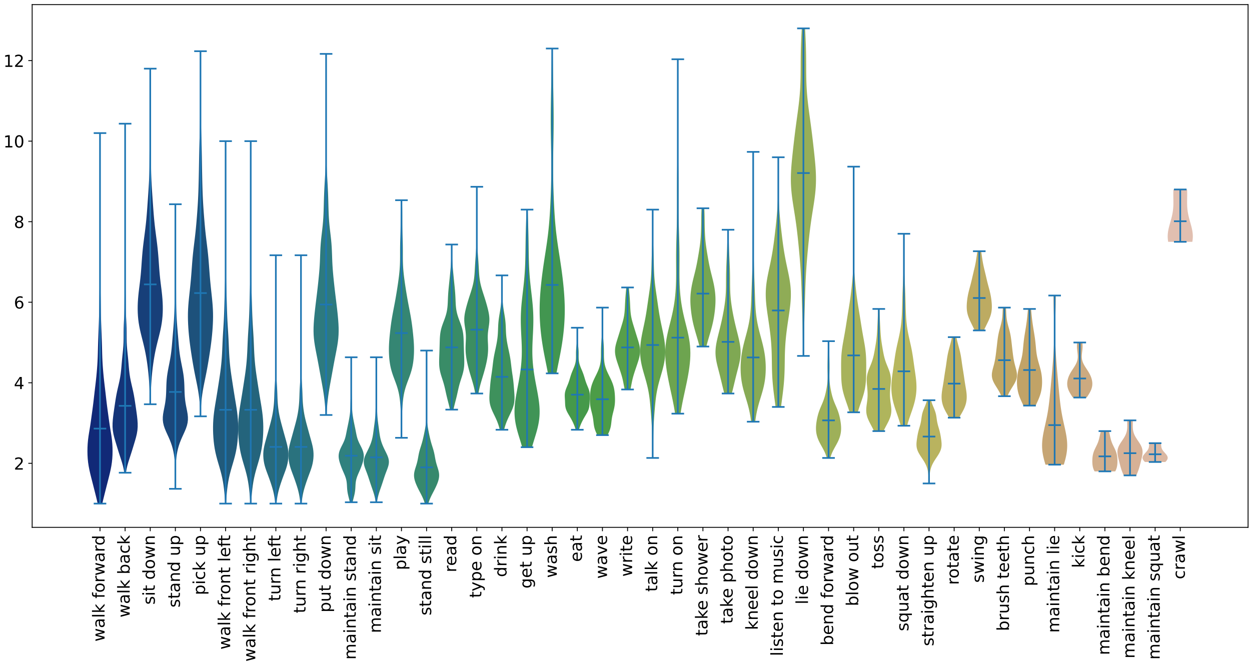


Fig. A2. **Motion length distribution of each motion type in LINGO dataset.**

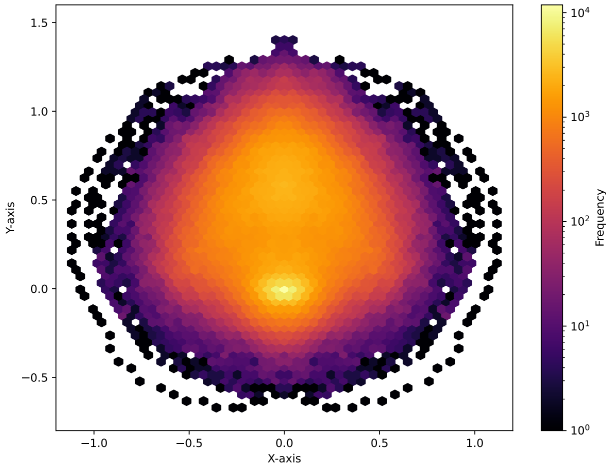


Fig. A3. **Distribution of goal locations for all locomotion clips in the local coordinate system of the first frame.** The character is aligned to initially face the y-axis direction. Unit: meter.

distribution, while others, such as “sit down” or “stand up,” have a longer tail towards higher motion lengths due to the varied Actor preferences. Some motion types, such as “lie,” have significantly longer motions. Motions with respect to locomotion, such as “walk forward” and “walk front left” have similar distributions, while interactive motions have distinct distributions compared to the rest.

B.2.4 Motion Occurance Count. Figure 6 displays the number of occurrences for each motion type in the LINGO dataset. The vertical axis represents the count on a logarithmic scale, while the horizontal axis lists the various motion types.

The number of occurrences varies across motion types. Locomotion-related motions have the highest number of occurrences, exceeding 1000 instances in the dataset. This is because locomotion occurs between two interaction motions. Most interaction motion types, such as “drink,” “eat,” and “wash,” have around 200 occurrences or segments in the LINGO dataset.

C TASK PLANNER

We show a workable pipeline that leverages GPT4-o to break down complex instruction into sub-tasks.

- prompt = “I need you to help me complete a task now. I will give you a target action. The target action is:“+action+.” You need to give in English a number of steps that I need to complete the target action. The steps should be as concise as possible without the need for irrelevant attributives. The details of the interaction action are not required. For example, there is no need to open the game console. The steps are divided into three categories: locomotion, grasp, and interaction, where locomotion only includes the movement of the person’s position, Grasp only consists of the grabbing and touching of objects, and interaction includes people’s operations on the appliance (such as listening to music with headphones, turning the door handle to open the door (excluding grabbing the door handle))), indicate the hand when it comes to hand movements. Please complete this task according to some step examples I gave you. Example: “+str(text_list)+,” output format: [{"step" :,"step_id":1,"category":},...], only output the final format, no other nonsense”

- output = [{"step": "walk to the sofa", "step_id": 1, "category": "locomotion"}, {"step": "sit down on the sofa", "step_id": 2, "category": "locomotion"}, {"step": "pick up remote with left hand", "step_id": 3, "category": "grasp"}, {"step": "turn on TV with left hand", "step_id": 4, "category": "interaction"}, {"step": "watch TV", "step_id": 5, "category": "interaction"}]