Generating Human Motion in 3D Scenes from Text Descriptions

Zhi Cen[1] Huaijin Pi[1] Sida Peng[1†] Zehong Shen[1]
Minghui Yang[2] Shuai Zhu[2] Hujun Bao[1] Xiaowei Zhou[1]
[1]Zhejiang University [2]Ant Group

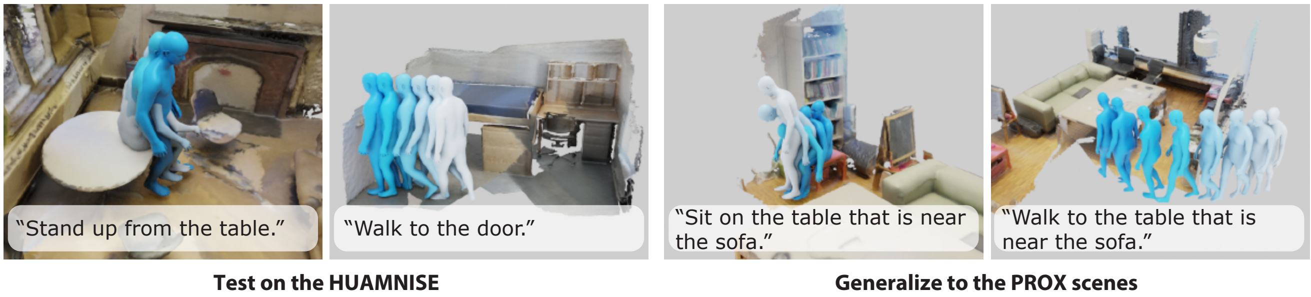

"Stand up from the table."
"Walk to the door."
"Sit on the table that is near the sofa."
"Walk to the table that is near the sofa."
Test on the HUAMNISE
Generalize to the PROX scenes

Figure 1. **Generating human motions in 3D scenes from text descriptions.** Our method can generate human motions containing accurate human-object interactions in 3D scenes based on textural descriptions. Although our method is trained and tested on the HUMANISE dataset, it can generalize to other scenes, e.g., the scenes in the PROX dataset. Left: test results on the HUMANISE dataset. Right: generalization results on the PROX scenes.

Abstract

Generating human motions from textual descriptions has gained growing research interest due to its wide range of applications. However, only a few works consider human-scene interactions together with text conditions, which is crucial for visual and physical realism. This paper focuses on the task of generating human motions in 3D indoor scenes given text descriptions of the human-scene interactions. This task presents challenges due to the multi-modality nature of text, scene, and motion, as well as the need for spatial reasoning. To address these challenges, we propose a new approach that decomposes the complex problem into two more manageable sub-problems: (1) language grounding of the target object and (2) object-centric motion generation. For language grounding of the target object, we leverage the power of large language models. For motion generation, we design an object-centric scene representation for the generative model to focus on the target object, thereby reducing the scene complexity and facilitating the modeling of the relationship between human motions and the object. Experiments demonstrate the better motion quality of our approach compared to baselines and validate our design choices. Code will be available at link.

1. Introduction

Human motion generation has been a long-standing problem due to its broad range of applications such as game development, virtual reality, and movie production. Recently, this area witnessed a paradigm shift from avatar animation given rich user input [29] to learning-based motion generation from high-level language prompts, e.g. text descriptions about the desired motion [2, 3, 15, 18, 19, 39, 56, 74, 75]. However, most prior works on text-driven motion synthesis do not consider human-scene interactions [39, 56, 74, 75] while the scene context and physical constraints of the environment largely define the fidelity of the generated human motions.

In this paper, we focus on generating motions from text descriptions in 3D indoor scenes. Specifically, given a 3D scan of the target scene and a text description of a human action that interacts with the scene, we aim to generate natural human motions that are consistent with the text description.

This problem presents several challenges, primarily due to the multi-modality nature of text, scene, and human motion. In contrast to previous methods [2, 3, 56, 74, 75] focusing solely on textual descriptions of how human moves, our task also includes texts that additionally describe the spatial details in the given scene (e.g., sit on the armchair near the desk). Therefore, this task requires spatial reason-

The authors from Zhejiang University are affiliated with the State Key Lab of CAD&CG. †Corresponding author.

ing skills [1], where the model should build a text-object mapping to locate a specific object in 3D scenes aligned with a natural language description. In addition, the generated motions should also be coherent with scene contexts.

As a pioneer work, HUMANISE [83] builds a conditional variational autoencoder (cVAE) [41, 69] with separate encoders of scenes and texts for multi-modality understanding. To enable spatial reasoning ability, they introduce auxiliary tasks of directly regressing object centers to learn 3D visual grounding in an implicit manner. But they do not explicitly utilize the predicted centers and thus the inductive bias of visual grounding cannot be fully incorporated. In addition, HUMANISE [83] encodes the entire scene with a single point transformer [99]. Directly generating motions with such a model is challenging, as 3D point clouds are inherently noisy and complex [48, 88], leading to the inability to locate the target object. As suggested by [51], not every point of the scene is relevant to the final human motions. Therefore, it is also necessary to develop a more targeted approach that focuses on the relevant parts of the scene to improve the quality of motion generation.

To tackle the problems mentioned above, we propose a novel approach that exploits the power of the large language models (LLMs) [6, 42, 52]. Our key idea is to address the challenging task of generating motion in a scene based on textual cues by breaking it down into two smaller problems: (1) language grounding of objects in 3D scenes and (2) generating motions with a focus on the target object. For language grounding, rather than directly learning text-object mapping, we propose to formulate it as question answering and utilize the large prior knowledge of LLMs. Specifically, we first construct scene graphs of 3D scenes and generate their textual descriptions. Then we employ ChatGPT [52] to analyze the relationship between scene descriptions and input instructions and respond with 3D visual grounding answers. Experiments prove the effectiveness of this strategy.

For motion generation, we design an object-centric representation to help the generative model focus on the target object. Specifically, we convert point clouds around the target object into volumetric sensors [70] to build object-centric representation. Then we employ diffusion models [57, 75] to synthesize human motions given object-centric representation and texts. Compared with original scene point clouds which might have various scales, object-centric representation is more compact and robust to scales as objects in the same category are of similar size. Therefore, this representation reduces scene complexity and facilitates the modeling of the relationship between motions and objects.

We conducted thorough comparative and ablation experiments on the HUMANISE dataset. The results demonstrate that our method outperforms the baseline, reflected in more accurate object grounding results and better motions that align with the textual descriptions and scenes. We further show that our approach can generalize to the PROX dataset [22] without any fine-tuning.

2. Related work

2.1. Motion synthesis

Deep learning for motion synthesis. Deep learning methods for motion synthesis have attracted increasing attention in recent years [14, 21, 25, 28, 29, 50, 71, 91]. Various techniques, including MLP [29], mixture of experts (MoE) [72, 91], and recurrent neural network (RNN) [21], are employed to tackle this task. To generate diverse motion results, previous works explore cVAE [46, 96], GANs [44, 61], normalizing flow [26], and diffusion models [75].

Text-driven motion generation. Recently, there has been a growing interest in text-driven motion generation [2, 56, 74]. This task takes natural language as input and synthesizes human motions that align with instructions. KIT-ML [58] is the first benchmark of this task. Some works further annotate the AMASS dataset [49] with action labels [59] and text [18]. To tackle this task, [2, 15, 74] propose to learn a shared latent space for text and motion. [56] employs transformer VAE [55] to generate diverse results. [19, 93] achieve better performance by discrete representation with VQ-VAE [65, 78]. [12, 39, 75, 94] successfully apply diffusion models in this direction. [5, 43] further explore latent diffusion models [67]. Based on [75], [38, 85] introduce sparse spatial control. [3, 60, 68] also investigate long sequence motion generation. Other existing works [4, 37, 82, 98] leverage Large Language Models (LLM) [6, 42, 53, 77] in the motion domain. [4] construct compositional motions by combining different body parts with LLM. [36, 98] regard motions as a kind of language and finetune LLMs [63] with motion tokens [19, 93].

Scene-aware motion generation This direction is to generate human motions in a 3D scene [70, 80]. To synthesize walking and sitting motions, [70] construct volumetric sensors to encode object information and the surroundings of the character. [23] extends it to synthesize motions with diverse sitting and lying styles. Furthermore, [95] proposes to control sitting styles by hand contacts. To improve the performance, [57] employs a hierarchical framework that generates goal poses, milestones, and motion sequentially. [73, 84] synthesize whole-body (body and hand fingers) grasping motions with small objects. [16, 45, 71, 86] explore interactions with dynamic objects including manipulation [16, 45] and carrying [86]. [8, 24] explore human-object interactions with physically simulated characters. These works mainly focus on one or two objects while others [80, 81] consider a more complex scene input (e.g., point clouds of the scene including walls and floors.). [80, 81] adopt hierarchical frameworks and generate trajectory and

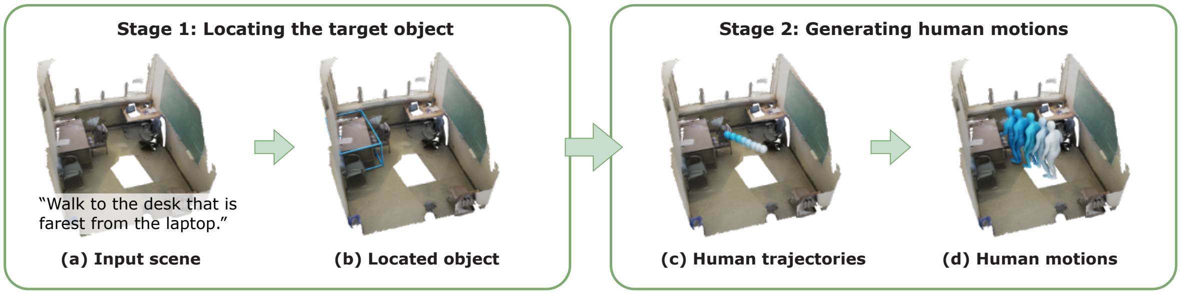

Stage 1: Locating the target object
"Walk to the desk that is farest from the laptop."
(a) Input scene
(b) Located object
Stage 2: Generating human motions
(c) Human trajectories
(d) Human motions

Figure 2. **Overview of our two-stage pipeline.** In the first stage, given an input scene and a text description (a), we use ChatGPT to locate the target object (b). In the second stage, human motions are synthesized by first producing human trajectories (c) and then generating local poses (d).

motion separately. [103] introduces gaze to help generation. [33] further employs diffusion models and [51] synthesizes very long-term motions in scenes by dividing long sequences into several short sequences. [101] designs a reinforcement learning pipeline to enable navigation in a complex scene and interaction style control.

Text-driven scene-aware motion generation Only a few works consider text and scenes simultaneously [38, 83, 101]. Although [101] enables text control of sitting styles by [100], the text descriptions in our setting are used to select an object in a cluttered scene. [38] could avoid obstacles during walking while our task needs to handle various actions. The most relevant work to us is HUMANISE [83], which employs a transformer VAE architecture with a two-stream condition module for text and scenes. To accurately localize target objects, they design auxiliary tasks like directly regressing object centers. In contrast to HUMANISE, we propose a two-stage pipeline where we first localize the target object with the help of ChatGPT [52] and then generate human motion using the object-centric representation.

2.2. 3D visual grounding.

In recent years, visual grounding in 3D scenes has been explored [1, 11, 76] and also tackled in 3D question answering [13, 89]. Given the point cloud of 3D scenes, this task [1] requires models to locate the target object according to text instructions. Most works follow a two-stage scheme where multiple bounding boxes [9, 66, 102] or segmentation results [31, 90] are first predicted and then selecting the object according to language descriptions. [32] introduces multi-view inputs, and [88] employs 2D semantics. [34] combines bottom-up [47] and top-down [7] detection methods. [48] designs a single-stage pipeline by progressively selecting key points. [20] extends [32] with the help of GPT [6] to generate multi-view text inputs. More recently, [30] proposes a neuro-symbolic framework with large language-to-code models [10]. Most works only localize a single object and [97] could localize a flexible number of objects. Different from previous works which directly handle point clouds or multi-view images, we convert the scene into textual descriptions and leverage large language models to infer the target object. Like [30], in this work, we also leverage large language models for object localization.

3. Problem setup and preliminaries

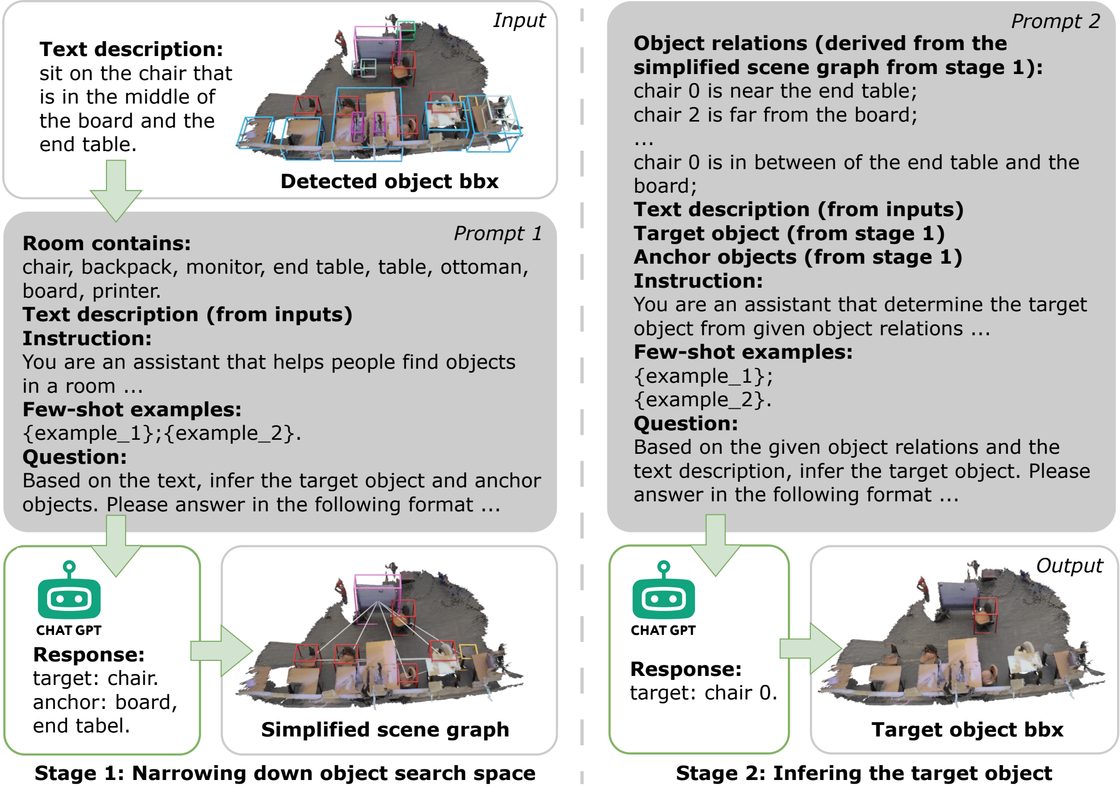
In this section, we discuss the definition of the task and preliminaries. We aim to generate human motion that is consistent with both the text description and the given scene.

Text descriptions. The text description follows the template in Sr3D [1] (e.g., "<sit on> <the chair> [<in the center of> <desk and bookshelf>]"). There are four actions (walk, sit, stand up, and lie). The $target$ represents the object that the agents need to interact with, and the $anchor$ objects help to determine the $target$. A certain type of target furniture category usually has many instances in one scene, while the anchor furniture should be unique. To specify the exact target object, there are five types of spatial relations [1] between the $target$ and the $anchor$: horizontal proximity, vertical proximity, between, allocentric, and support.

Scene representations. The scene is denoted as a point cloud of N points: $S \in \mathbf{R}^{N\times 6}$, containing the position and normal direction information of each point.

Motion representations. The output motion sequence is represented as a sequence of SMPL-X [54] body meshes M. SMPL-X is a parametric human body model. In this work, body parameters include body shape parameters $\beta \in \mathbf{R}^{10}$, global translation $r \in \mathbf{R}^3$, 6D global orientation $\gamma \in \mathbf{R}^6$, and 6D pose parameters of J joints $\boldsymbol{\theta} \in \mathbf{R}^{J\times 6}$ [104]. Following [83], β is treated as a condition to model the effect of body shape, and we omit it for ease of notation. Note that start position and pose are also generated by the model.

Diffusion models. The diffusion [27] is defined as a Markov noising process $\{\mathbf{x}_t\}_{t=0}^{T}$ that follows $q(\mathbf{x}_t \mid \mathbf{x}_0)$, where $\mathbf{x}_0 \sim q(\mathbf{x}_0)$ is the data and $\mathbf{x}_t$ is the noised data at



Text description: sit on the chair that is in the middle of the board and the end table.

Room contains: chair, backpack, monitor, end table, table, ottoman, board, printer.
Text description (from inputs)
Instruction: You are an assistant that helps people find objects in a room ...
Few-shot examples: {example_1};{example_2}.
Question: Based on the text, infer the target object and anchor objects. Please answer in the following format ...

Response: target: chair. anchor: board, end tabel.

Object relations (derived from the simplified scene graph from stage 1):
chair 0 is near the end table;
chair 2 is far from the board;
...
chair 0 is in between of the end table and the board;
Text description (from inputs)
Target object (from stage 1)
Anchor objects (from stage 1)
Instruction: You are an assistant that determine the target object from given object relations ...
Few-shot examples:
{example_1};
{example_2}.
Question: Based on the given object relations and the text description, infer the target object. Please answer in the following format ...

Response: target: chair 0.

Figure 3. **Pipeline of localizing the target object.** In stage 1, given the input text description and detected object bounding boxes (bbx), we construct the first prompt asking ChatGPT the categories of target objects and anchor objects. Based on the response, the scene graph can be simplified. In stage 2, we construct the second prompt with inputs and results from stage 1, including object relations derived from the simplified scene graph. The second prompt is designed for asking ChatGPT to infer the target object. Finally, we can get the target object bounding box from the response of ChatGPT.

noising step t. The formal definition is:

$$q\left(\mathbf{x}_t \mid \mathbf{x}_0\right)=\mathcal{N}\left(\mathbf{x}_t ; \sqrt{\bar{\alpha}_t} \mathbf{x}_0,\left(1-\bar{\alpha}_t\right) \mathbf{I}\right), \tag{1}$$

where $\bar{\alpha}_t$ are constants with monotonically decreasing schedule. When $\bar{\alpha}_t$ is small enough, we can approximate $\mathbf{x}_T \sim \mathcal{N}(\mathbf{0}, \mathbf{I})$. In our context, we use conditional diffusion models like [64, 75]. The training loss is defined as:

$$\mathcal{L}=\mathrm{E}_{t \in[1, T], \mathbf{x}_0 \sim q\left(\mathbf{x}_0\right)}\left[\left\|\mathbf{x}_0-G\left(\mathbf{x}_t, t, \mathbf{C}\right)\right\|\right], \tag{2}$$

where G is the generative model, and $\mathbf{C}$ is the condition.

4. Method

The overview of our method is shown in Fig. 2. In Sec. 4.1, we leverage ChatGPT to localize the target object given a 3D scene and a textual description. Based on the accurate localization, we can focus on the target object and employ an object-centric generation pipeline to separately synthesize trajectories (Sec. 4.2) and motions (Sec. 4.3). Implementation details are discussed in Sec. 4.4.

4.1. Language grounding of objects in 3D scenes

Scene-aware motion generation from textual descriptions requires the scene-understanding ability and build the relationship between scenes and texts. Since the instructions describe how the character moves and interacts with a single target object, the majority of the scene might bear negligible relevance to the final motions [51]. Motivated by this, we propose to locate the target object to identify the most pertinent information. Inspired by the recent progress of LLM [42, 52, 53], ChatGPT [52] is utilized to find the specific objects in the given text. We first obtain textual descriptions of scenes by building scene graphs. Then we feed them with text instructions to ChatGPT with specially designed prompts and parse the response to get target objects.

Spatial scene graph extraction. To utilize LLM, the initial step involves converting a 3D scene into text. This is achieved by building a *spatial scene graph*. We utilize a pre-trained 3D object detection model in [47] to provide 3D box proposals. Subsequently, we follow the approach of [1] to obtain the relationships between objects. Specifically, for

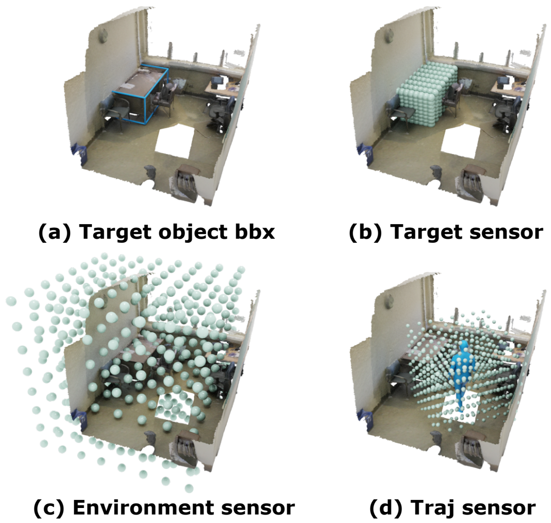


Figure 4. **The visualization of the environment sensor, target sensor, and trajectory sensor.** The target sensor (b) gives detailed geometry of the target object. The environment sensor (c) gives coarse spatial information around the target object. The trajectory sensor (d) is located around the human.

every set of two objects, we can infer their relationships in three types from their bounding boxes: *horizontal proximity*, *vertical proximity*, and *support* as mentioned in Sec. 3; for every set of three objects, we can infer if one of them is in *between* of the other two objects. As detection results do not contain object poses, we do not construct *allocentric* relations (e.g., "the shelf that is behind the sofa"). Then, we build a scene graph, where each object is assigned as a node in the graph, and edges between nodes represent the relationships between objects. By converting 3D scenes into text, we can apply ChatGPT to extract meaningful insights from the data.

Leveraging ChatGPT to localize the target object. A simple approach is to directly input the entire scene graph into ChatGPT and ask it to select the target objects. However, scenes may contain many objects, resulting in extremely long textural descriptions. We observe that ChatGPT is often confused in such settings and fails to respond with the right answer. To narrow the search space, we first employ ChatGPT to recognize objects that correlate to the provided text. We then exclude the unrelated objects from the scene graph, focusing solely on those with pertinent information, enabling us to pinpoint the target object effectively. This approach has the advantage of reducing the number of objects that need to be considered, making it easier for ChatGPT to identify the target object.

As shown in Fig. 3, we construct two prompts in sequence. Take "sit on the chair that is in the middle of the board and the end table" as an example, we first need to narrow down the object category search space. In order to find which type of objects we care about, we construct the first prompt that asks ChatGPT to find out *target objects* and *anchor objects*. *Target object* is defined as the final object that we want the agent to interact with, which in this case, is the "table". *Anchor objects* are the objects that help with determining the *target object*, for there might be many chairs in one scene. Based on the target object "chair" and anchor objects "board" and "end table", we can filter out all the unrelated objects in the scene, only keeping chairs, the board, and the end table in our scene graph. Next, according to the simplified scene graph, we can describe object relations in text: every edge in the scene graph could be converted to an *edge sentence* like "chair 4 is far from the end table 0". Converting all edges to such edge sentences gives a full description of the current scene. Finally, we construct a second prompt by asking ChatGPT to infer the target object from the accumulated edge sentences.

4.2. Diffusion-based trajectory generation

Given the localized object from ChatGPT, we first generate the trajectory based on the instructions and then synthesize local human poses. Trajectory is defined as a sequence of characters' translations and orientations. As suggested by [51], not every point of the scene is relevant to the final human motions. Inspired by NSM [70] and ManipNet [92], we employ volumetric sensors (as shown in Fig. 4) around the target object to represent the scene.

Object-centric scene representation. Denote the target object center location as $c_o = (c_x, c_y, c_z)$. We transform the point cloud of the scene S to an object-centered coordinate axis centered at c_o. To recognize the surrounding geometry of the target object, we create volumetric sensors called Environment Sensor and Target Sensor.

Environment sensor. The environment sensor is centered at c_o in a cubic shape with a volume of $4 \times 4 \times 4\ m^3$, containing $8 \times 8 \times 8$ cubic voxels as shown in Fig. 4 (c). It is constructed by collecting all occupancies o_s, center positions c_v, and normal directions n_v of each voxel to form a feature vector E. Like [70], the scene occupancy $o_s \in R^1$ in each voxel is defined based on the given scene mesh:

$$o_s = \begin{cases} 1 & \text{if } d_s < 0, \\ 0 & \text{if } d_s > a_s, \\ 1 - \frac{d_s}{a_s} & \text{otherwise,} \end{cases} \tag{3}$$

where d_s is the signed distance between the scene mesh and the voxel center, and a_s is the voxel edge length. n_v is the normal direction of the closest scene point to the voxel center. The environment sensor provides coarse scene geometry around the target object.

Target sensor. To capture the detailed geometry of the target object, we further build a target sensor. As we already obtain the 3D bounding box of the target object in Sec. 4.1, we crop the point clouds according to the bounding box and construct an $8 \times 8 \times 8$ cubic volumes that cover the bounding box, as shown in Fig. 4 (b). Target sensor T is in the same form as the environment sensor E, except the target sensor has a different voxel size, as visualized in Fig. 4 (b) and (c).

Given the constructed object-centric representations, we follow [57] to employ a transformer decoder architecture [79], which enables arbitrary length motions. As for the text input, we use CLIP [62] text encoder to encode input text to the text feature L. The time-step t is injected into the decoder in sinusoidal position embeddings form [79]. In summary, the condition for this generation model is

$$\mathbf{C_t} = \{L, E, T\}, \tag{4}$$

where L is the text feature, E is the environment sensor, and T is the target sensor. All the conditions are projected to the same dimension $D = 512$ by MLPs and summed with positional embeddings to form tokens. We use the simple objective described in Eq. 2 to train the trajectory generation model G_r to generate the trajectory $\mathbf{r}_{1:N}$ with length N.

4.3. Diffusion-based motion completion

Given the trajectory from Sec. 4.2, the next step is to complete the whole motion. Based on the generated trajectories, we construct a Trajectory Sensor O to explicitly reason about the interaction between the character and scenes.

Trajectory sensor. The trajectory sensor is also a volumetric sensor which is similar to the form of the environment sensor but is designed for ego-centric perception [95] as shown in Fig. 4 (d). Specifically, trajectory sensors are positioned around characters in each frame. This sensor O_i is centered at the predicted root position of the i-th frame and faces to the i-th frame's predicted root orientation, containing $8 \times 8 \times 8$ cubic voxels that store scene occupancy. The occupancy calculation is the same as Eq. 3.

Another transformer-based conditional diffusion model is used to synthesize local poses along trajectories. The condition is defined as:

$$\mathbf{C_m} = \{L, E, T, O_1, ..., O_N\}, \tag{5}$$

where L, E, and T have the same meanings in Eq. 4. Simple objective described in Eq. 2 is used to train the motion generation model G_m to generate global orientation $\gamma_{1:N}$ and local poses $\boldsymbol{\theta}_{1:N}$ with length N.

4.4. Implementation details

Following [38, 57, 80, 81], we use separate diffusion models for the trajectory generation and the motion generation. Because the HUMANISE dataset [83] contains a relatively small number (51 minutes) of pure motion samples from the AMASS dataset (3772 minutes) [49], we first pre-train the models on the whole AMASS dataset for 200 epochs and then fine-tune on the HUMANISE dataset for 200 epochs to improve the motion quality. During pretraining, the text feature L is set to all zeros. Both models are trained with the AdamW optimizer [40], using a learning rate of 0.0001 on a single Nvidia RTX 3090 GPU. The batch size is set to 128. The version of ChatGPT is *gpt-3.5-turbo*. More details can be found in the supplementary material.

5. Experiments

5.1. Evaluation metrics

We evaluate the generated motions in three aspects: scene-conditional, action-conditional, and pure motion quality.

Scene-conditional motion quality. To evaluate how the generated motion is aligned with the scene, we calculate the body-to-goal distance (**goal dist.**) [83] to measure how accurately the character interacts with target objects. However, goal dist. does not consider the consistency of the entire motion and scene (e.g., sitting on the sofa with incorrect orientation). To compensate for goal dist., a human perceptual study (indicated by **scene score**) is performed by randomly sampling 20 scenarios for each model, where ten workers are required to score each sample.

Action-conditional motion quality. To measure how the generated motion is aligned with the text, we follow [17] to evaluate action recognition accuracy (**accuracy**), **diversity**, and **multimodality** of the results. Calculating these metrics relies on a pre-trained action recognition model [87] and we train the recognition model on the HUMANISE dataset.

Pure motion quality. We evaluate the realism of generated motions using the metric suggested by [17], namely Frechet Inception Distance (**FID**). A lower FID indicates that generated motions are closer to the groundtruth motions. We also perform a human perceptual study (indicated by **quality score**) to measure pure motion quality.

5.2. Generating human motion from text and scene

Our experiments are conducted on the HUMANISE dataset [83]. Following the previous setting [83], there are $16.5k$ motion sequences in 543 scenes for training and $3.1k$ motion sequences in 100 scenes for testing. We compare our method with four baselines: (1) **MDM*** [75]: a diffusion-based motion generator. (2) **GMD*** [38]: GMD proposes various techniques for enhancing the control and quality of MDM. We employ the single-stage setting of GMD*. Similar to HUMANISE, we use a point-transformer to provide scene features for MDM*and GMD*. (3) **GMD$^{\mathbf{HC}}$**: we provide object centers predicted by HUMANISE (denoted by HC) to guide the motion generation process in GMD by adding a proximity loss to encourage the motion to be close to the predicted object center. The proximity loss is defined

Methods	Scene-conditional		Action-conditional			Pure motion quality	
	goal dist.↓	scene score↑	accuracy↑	diversity→	multimodality→	FID↓	quality score↑
Real	0.014	-	99.1%	4.82	2.28	0.00	-
MDM*	2.048	1.99	96.0%	**4.83**	2.57	0.16	2.45
GMD*	1.229	2.50	96.6%	4.91	**2.28**	0.24	2.64
GMD^{HC}	1.130	2.49	96.4%	4.93	2.43	0.23	2.76
HUMANISE	0.995	2.33	89.7%	4.25	2.66	1.11	2.81
Ours	**0.384**	**3.54**	**97.1%**	4.78	2.31	**0.12**	**3.50**

Table 1. **Quantitative results on the HUMANISE dataset.** We compare our method with four baselines (please refer to Sec. 5.2) and the real data. GMD^{HC} means using HUMANISE's predicted center to guide motion generation in GMD*. Among the metrics, scene score and quality score are perceptual studies. ↑ means higher is better and ↓ means lower is better. → means closer to the real data is better. **Bold** indicates the best results. Underline indicates the second best.

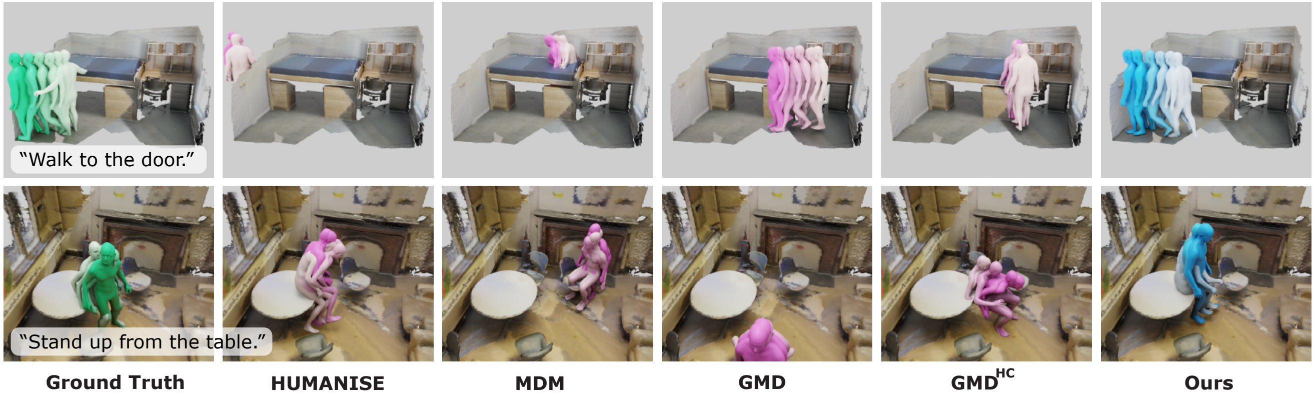


Figure 5. **Qualitative results.** We compare our method with groundtruth and four baselines (please refer to Sec. 5.2) given the same text descriptions. Our method synthesizes motions that interact with the object precisely as the groundtruth data while the baselines fail.

as the distance from HC to the predicted human pelvis at the interacting frame. (4) **HUMANISE**: we directly use their released models.

The quantitative results are shown in Tab. 1. Our method outperforms the baseline in terms of goal dist., scene score, accuracy, FID, and quality score and achieves competitive results in diversity and multimodality. The quantitative results show that our approach can generate better-quality motions and are more consistent with all the conditions. The qualitative results are demonstrated in Fig. 5. Please refer to the supplementary material for more visualizations.

5.3. Ablation study

Ablation of main components. Four variants are constructed to explore the effect of our design choice. (1) **"w/o localization"**: the localization module is removed and trajectories and motions are generated directly using scene point clouds. Scene features are given as the same form of MDM*and GMD*. (2) **"w/o object-centric"**: we remove our object-centric representation and predict motions in the scene coordinate. The scene features of our sensors are still employed. (3) **"w/o two-stage"**: trajectories and motions are generated together. (4) **"w/o diffusion"**: we employ cVAE instead of the diffusion model as the architecture. (5) **"w/o pretrain"**: the models are not pre-trained on the AMASS dataset. As shown in Tab. 2, the localization enhances the precise interaction ability and the object-centric representation improves motion quality. Although the ("w/o two-stage") is more efficient and has competitive performance metrics comparing to our method, it is prone to collapse directly into the target position, lacking gradual transition and realism.

The design choice of object localization. For object localization, we compare our method with two baselines and five variants. (1) **HUMANISE**: predicts the object center in the auxiliary task. (2) **BUTD-DETR** [34]: is a 3D visual grounding method. (3) **"ours w/o two-stage"**: we employ a one-stage question-answering process. (4) **"ours w/o few-shot"**: the few-shot examples are not provided. (5) **"ours using text matching"**: we compute CLIP similarities instead of using ChatGPT. (6) **"ours using LLaMA"**: ChatGPT is replaced by LLaMA 2 7B model. (7) **"ours using Mistral"**: ChatGPT is replaced by Mistral 7B [35], an open-source LLM. The prompts for LLaMA 2and Mistral

Variants	Scene	Action			Pure
	goal dist.↓	accuracy↑	diversity→	mm→	FID↓
Real	0.014	99.1%	4.82	2.28	0.00
w/o localization	0.592	74.5%	3.71	2.75	3.14
w/o object-centric	0.413	90.6%	4.13	2.63	1.01
w/o two-stage	0.385	**97.2%**	4.93	2.33	0.14
w/o diffusion	0.390	40.0%	3.69	3.86	3.61
w/o pretrain	0.392	82.6%	3.88	2.37	2.50
Ours	**0.384**	97.1%	**4.78**	**2.31**	**0.12**

Table 2. **Ablation of main components.** We compare our method with five variants (please refer to Sec. 5.3). Among them, *mm* indicates multimodality. **Bold** indicates the best results. Underline indicates the second best.

7B are slightly adjusted. We evaluate these variants under two scenarios: predicted detection [47] and groundtruth detection. The metrics include the accuracy (**acc.**) and the distance from predicted centers to the object center (**center dist.**). "acc." is the percentage of times with an IoU (Intersection over Union) higher than the threshold (0.25) following [34]. The results are shown in Tab. 3. Since the results of Mistral are only slightly behind of ChatGPT, ChatGPT can be replaced by Mistral for better reproducibility, but cannot be replaced by the text matching method. Moreover, if groundtruth detection is provided, our method can achieve even better results.

The design choice of sensor density. Please refer to the supplementary material.

5.4. Generalization

To validate the generalization ability of our method, we run our method directly on the unseen PROX dataset [22] without fine-tuning. We provide illustrations in Fig. 6. With our localization method based on ChatGPT and motion generation method based on volumetric sensors, our pipeline can easily generalize to other datasets. Please refer to the supplementary material for more results.

6. Discussion

We have demonstrated that our approach could synthesize motions with better quality and more precise interactions. However, restricted by the dataset, the duration of motions is relatively short (60% are around 1-3s), and most texts follow the template [1] without detailed descriptions. We only tackle static scenes and extending our method to interact with moving objects is a future direction. We acknowledge that leveraging LLMs has several problems. ChatGPT may fail when the detector misses the objects and the behavior of ChatGPT is regulated by prompts [37]. Despite we have shown in Tab. 3 that ChatGPT can be replaced by an open source LLM Mistral 7B, using LLMs instead of classical parsing approaches is less efficient in the inference stage.

Variants	Predicted detection		GT detection	
	acc. ↑	center dist.↓	acc. ↑	center dist.↓
HUMANISE	-	1.48	-	-
BUTD-DETR	62.5	1.33	63.1	1.23
Ours w/o two-stage	49.0	1.34	58.5	1.10
Ours w/o few-shot	69.3	0.80	86.3	0.45
Ours using text matching	67.0	0.89	79.7	0.52
Ours using LLaMA 2	40.3	1.55	42.5	1.52
Ours using Mistral 7B	72.4	0.74	87.2	**0.33**
Ours using ChatGPT-3.5	**75.6**	**0.61**	**90.5**	0.37

Table 3. **Design choices of object localization.** We compare our method with two baselines and five variants (please refer to Sec. 5.3). Since HUMANISE directly regresses the coordinate of centers without utilizing groundtruth detection, we do not calculate acc. and only include their results under predicted detection. **Bold** indicates the best results.

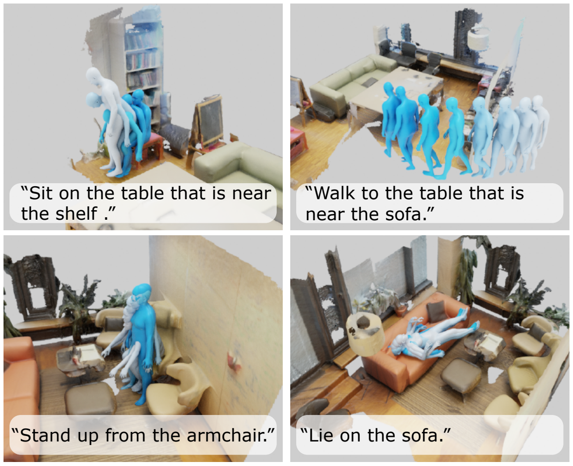

"Sit on the table that is near the shelf ."
"Walk to the table that is near the sofa."
"Stand up from the armchair."
"Lie on the sofa."

Figure 6. **Qualitative results of our method on the PROX dataset.** We run our method on the scenes from the PROX dataset without fine-tuning. Results show that our method is capable to generalize to unseen scenes and objects.

7. Conclusion

In this work, we introduce a novel method for generating motion from text in a scene. To tackle this problem, we propose a two-step approach. The first step involves 3D visual grounding, where we identify the target object. In the second step, concentrating on the target object, we build a diffusion-based motion generation method. Our approach offers several advantages, including improved 3D visual grounding accuracy and motion quality.

Acknowledgement

This work was partially supported by the NSFC (No. 62172364), Ant Group and Information Technology Center and State Key Lab of CAD&CG, Zhejiang University.

References

[1] Panos Achlioptas, Ahmed Abdelreheem, Fei Xia, Mohamed Elhoseiny, and Leonidas Guibas. Referit3d: Neural listeners for fine-grained 3d object identification in real-world scenes. In *ECCV*, 2020. 2, 3, 4, 8

[2] Chaitanya Ahuja and Louis-Philippe Morency. Language2pose: Natural language grounded pose forecasting. In *3DV*, 2019. 1, 2

[3] Nikos Athanasiou, Mathis Petrovich, Michael J. Black, and Gül Varol. TEACH: Temporal Action Compositions for 3D Humans. In *3DV*, 2022. 1, 2

[4] Nikos Athanasiou, Mathis Petrovich, Michael J. Black, and Gül Varol. SINC: Spatial composition of 3D human motions for simultaneous action generation. *ICCV*, 2023. 2

[5] German Barquero, Sergio Escalera, and Cristina Palmero. Belfusion: Latent diffusion for behavior-driven human motion prediction. In *ICCV*, 2023. 2

[6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. Language models are few-shot learners. In *NeurIPS*, 2020. 2, 3

[7] Nicolas Carion, Francisco Massa, Gabriel Synnaeve, Nicolas Usunier, Alexander Kirillov, and Sergey Zagoruyko. End-to-end object detection with transformers. In *ECCV*, 2020. 3

[8] Yu-Wei Chao, Jimei Yang, Weifeng Chen, and Jia Deng. Learning to sit: Synthesizing human-chair interactions via hierarchical control. In *AAAI*, 2021. 2

[9] Dave Zhenyu Chen, Angel X Chang, and Matthias Nießner. Scanrefer: 3d object localization in rgb-d scans using natural language. In *ECCV*, 2020. 3

[10] Mark Chen, Jerry Tworek, Heewoo Jun, Qiming Yuan, Henrique Ponde de Oliveira Pinto, Jared Kaplan, Harri Edwards, Yuri Burda, Nicholas Joseph, Greg Brockman, et al. Evaluating large language models trained on code. *arXiv preprint*, 2021. 3

[11] Yixin Chen, Qing Li, Deqian Kong, Yik Lun Kei, Song-Chun Zhu, Tao Gao, Yixin Zhu, and Siyuan Huang. Yourefit: Embodied reference understanding with language and gesture. In *ICCV*, 2021. 3

[12] Rishabh Dabral, Muhammad Hamza Mughal, Vladislav Golyanik, and Christian Theobalt. Mofusion: A framework for denoising-diffusion-based motion synthesis. In *CVPR*, 2023. 2

[13] Abhishek Das, Georgia Gkioxari, Stefan Lee, Devi Parikh, and Dhruv Batra. Neural Modular Control for Embodied Question Answering. In *CoRL*, 2018. 3

[14] Katerina Fragkiadaki, Sergey Levine, Panna Felsen, and Jitendra Malik. Recurrent network models for human dynamics. In *ICCV*, 2015. 2

[15] Anindita Ghosh, Noshaba Cheema, Cennet Oguz, Christian Theobalt, and Philipp Slusallek. Synthesis of compositional animations from textual descriptions. In *ICCV*, 2021. 1, 2

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

[17] 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 *ACM MM*, 2020. 6

[18] 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 *CVPR*, 2022. 1, 2

[19] 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. 1, 2

[20] Zoey Guo, Yiwen Tang, Ray Zhang, Dong Wang, Zhigang Wang, Bin Zhao, and Xuelong Li. Viewrefer: Grasp the multi-view knowledge for 3d visual grounding. In *ICCV*, 2023. 3

[21] Félix G Harvey, Mike Yurick, Derek Nowrouzezahrai, and Christopher Pal. Robust motion in-betweening. *ACM Trans. Graph.*, 2020. 2

[22] Mohamed Hassan, Vasileios Choutas, Dimitrios Tzionas, and Michael J. Black. Resolving 3d human pose ambiguities with 3d scene constraints. In *ICCV*, 2019. 2, 8

[23] Mohamed Hassan, Duygu Ceylan, Ruben Villegas, Jun Saito, Jimei Yang, Yi Zhou, and Michael J. Black. Stochastic scene-aware motion prediction. In *ICCV*, 2021. 2

[24] Mohamed Hassan, Yunrong Guo, Tingwu Wang, Michael Black, Sanja Fidler, and Xue Bin Peng. Synthesizing physical character-scene interactions. In *SIGGRAPH*, 2023. 2

[25] Chengan He, Jun Saito, James Zachary, Holly Rushmeier, and Yi Zhou. Nemf: Neural motion fields for kinematic animation. In *NeurIPS*, 2022. 2

[26] Gustav Eje Henter, Simon Alexanderson, and Jonas Beskow. Moglow: Probabilistic and controllable motion synthesis using normalising flows. *ACM Trans. Graph.*, 2020. 2

[27] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models. In *NeurIPS*, 2020. 3

[28] Daniel Holden, Jun Saito, and Taku Komura. A deep learning framework for character motion synthesis and editing. *ACM Trans. Graph.*, 2016. 2

[29] Daniel Holden, Taku Komura, and Jun Saito. Phase-functioned neural networks for character control. *ACM Trans. Graph.*, 2017. 1, 2

[30] Joy Hsu, Jiayuan Mao, and Jiajun Wu. Ns3d: Neuro-symbolic grounding of 3d objects and relations. In *CVPR*, 2023. 3

[31] Pin-Hao Huang, Han-Hung Lee, Hwann-Tzong Chen, and Tyng-Luh Liu. Text-guided graph neural networks for referring 3d instance segmentation. *AAAI*, 2021. 3

[32] Shijia Huang, Yilun Chen, Jiaya Jia, and Liwei Wang. Multi-view transformer for 3d visual grounding. In *CVPR*, 2022. 3

[33] 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. In *CVPR*, 2023. 3

[34] Ayush Jain, Nikolaos Gkanatsios, Ishita Mediratta, and Katerina Fragkiadaki. Bottom up top down detection transformers for language grounding in images and point clouds. In *ECCV*, 2022. 3, 7, 8

[35] Albert Q Jiang, Alexandre Sablayrolles, Arthur Mensch, Chris Bamford, Devendra Singh Chaplot, Diego de las Casas, Florian Bressand, Gianna Lengyel, Guillaume Lample, Lucile Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023. 7

[36] Biao Jiang, Xin Chen, Wen Liu, Jingyi Yu, Gang Yu, and Tao Chen. Motiongpt: Human motion as a foreign language. *arXiv preprint arXiv:2306.14795*, 2023. 2

[37] Sai Shashank Kalakonda, Shubh Maheshwari, and Ravi Kiran Sarvadevabhatla. Action-gpt: Leveraging large-scale language models for improved and generalized action generation. In *ICME*, 2023. 2, 8

[38] Korrawe Karunratanakul, Konpat Preechakul, Supasorn Suwajanakorn, and Siyu Tang. Guided motion diffusion for controllable human motion synthesis. In *ICCV*, 2023. 2, 3, 6

[39] Jihoon Kim, Jiseob Kim, and Sungjoon Choi. Flame: Free-form language-based motion synthesis & editing. In *AAAI*, 2023. 1, 2

[40] Diederik P. Kingma and Jimmy Ba. Adam: A Method for Stochastic Optimization. *arXiv e-prints*, 2014. 6

[41] Diederik P Kingma and Max Welling. Auto-Encoding Variational Bayes. *arXiv e-prints*, 2013. 2

[42] Takeshi Kojima, Shixiang (Shane) Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. Large language models are zero-shot reasoners. In *NeurIPS*, 2022. 2, 4

[43] Hanyang Kong, Kehong Gong, Dongze Lian, Michael Bi Mi, and Xinchao Wang. Priority-centric human motion generation in discrete latent space. In *ICCV*, 2023. 2

[44] Peizhuo Li, Kfir Aberman, Zihan Zhang, Rana Hanocka, and Olga Sorkine-Hornung. Ganimator: Neural motion synthesis from a single sequence. *ACM Trans. Graph.*, 2022. 2

[45] Quanzhou Li, Jingbo Wang, Chen Change Loy, and Bo Dai. Task-oriented human-object interactions generation with implicit neural representations. *arXiv preprint*, 2023. 2

[46] Hung Yu Ling, Fabio Zinno, George Cheng, and Michiel Van De Panne. Character controllers using motion vaes. *ACM Trans. Graph.*, 2020. 2

[47] Ze Liu, Zheng Zhang, Yue Cao, Han Hu, and Xin Tong. Group-free 3d object detection via transformers. In *ICCV*, 2021. 3, 4, 8

[48] Junyu Luo, Jiahui Fu, Xianghao Kong, Chen Gao, Haibing Ren, Hao Shen, Huaxia Xia, and Si Liu. 3d-sps: Single-stage 3d visual grounding via referred point progressive selection. In *CVPR*, 2022. 2, 3

[49] Naureen Mahmood, Nima Ghorbani, Nikolaus F Troje, Gerard Pons-Moll, and Michael J Black. Amass: Archive of motion capture as surface shapes. In *ICCV*, 2019. 2, 6

[50] Julieta Martinez, Michael J. Black, and Javier Romero. On human motion prediction using recurrent neural networks. In *CVPR*, 2017. 2

[51] Aymen Mir, Xavier Puig, Angjoo Kanazawa, and Gerard Pons-Moll. Generating continual human motion in diverse 3d scenes. *arXiv preprint*, 2023. 2, 3, 4, 5

[52] OpenAI. Openai: Introducing chatgpt. `https://openai.com/blog/chatgpt`, 2022. 2, 3, 4

[53] OpenAI. Gpt-4 technical report, 2023. 2, 4

[54] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive body capture: 3d hands, face, and body from a single image. In *CVPR*, 2019. 3

[55] Mathis Petrovich, Michael J. Black, and Gül Varol. Action-conditioned 3D human motion synthesis with transformer VAE. In *ICCV*, 2021. 2

[56] Mathis Petrovich, Michael J. Black, and Gül Varol. TEMOS: Generating diverse human motions from textual descriptions. In *ECCV*, 2022. 1, 2

[57] Huaijin Pi, Sida Peng, Minghui Yang, Xiaowei Zhou, and Hujun Bao. Hierarchical generation of human-object interactions with diffusion probabilistic models. In *ICCV*, 2023. 2, 6

[58] Matthias Plappert, Christian Mandery, and Tamim Asfour. The kit motion-language dataset. *Big data*, 2016. 2

[59] Abhinanda R. Punnakkal, Arjun Chandrasekaran, Nikos Athanasiou, Alejandra Quiros-Ramirez, and Michael J. Black. Babel: Bodies, action and behavior with english labels. In *CVPR*, 2021. 2

[60] Yijun Qian, Jack Urbanek, Alexander G Hauptmann, and Jungdam Won. Breaking the limits of text-conditioned 3d motion synthesis with elaborative descriptions. In *ICCV*, 2023. 2

[61] Sigal Raab, Inbal Leibovitch, Peizhuo Li, Kfir Aberman, Olga Sorkine-Hornung, and Daniel Cohen-Or. Modi: Unconditional motion synthesis from diverse data. In *CVPR*, 2023. 2

[62] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, 2021. 6

[63] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *The Journal of Machine Learning Research*, 21(1):5485–5551, 2020. 2

[64] Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical Text-Conditional Image Generation with CLIP Latents. *arXiv e-prints*, 2022. 4

[65] Ali Razavi, Aaron van den Oord, and Oriol Vinyals. Generating diverse high-fidelity images with vq-vae-2. In *NeurIPS*, 2019. 2

[66] Junha Roh, Karthik Desingh, Ali Farhadi, and Dieter Fox. Languagerefer: Spatial-language model for 3d visual grounding. In *CoRL*, 2022. 3

[67] Robin Rombach, Andreas Blattmann, Dominik Lorenz, Patrick Esser, and Björn Ommer. High-resolution image synthesis with latent diffusion models. In *CVPR*, 2022. 2

[68] Yonatan Shafir, Guy Tevet, Roy Kapon, and Amit H Bermano. Human motion diffusion as a generative prior. *arXiv preprint*, 2023. 2

[69] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning structured output representation using deep conditional generative models. In *NeurIPS*, 2015. 2

[70] Sebastian Starke, He Zhang, Taku Komura, and Jun Saito. Neural state machine for character-scene interactions. *ACM Trans. Graph.*, 2019. 2, 5

[71] Sebastian Starke, Yiwei Zhao, Taku Komura, and Kazi Zaman. Local motion phases for learning multi-contact character movements. *ACM Trans. Graph.*, 2020. 2

[72] Sebastian Starke, Ian Mason, and Taku Komura. Deepphase: Periodic autoencoders for learning motion phase manifolds. *ACM Trans. Graph.*, 2022. 2

[73] Omid Taheri, Vasileios Choutas, Michael J. Black, and Dimitrios Tzionas. Goal: Generating 4d whole-body motion for hand-object grasping. In *CVPR*, 2022. 2

[74] Guy Tevet, Brian Gordon, Amir Hertz, Amit H Bermano, and Daniel Cohen-Or. Motionclip: Exposing human motion generation to clip space. In *ECCV*, 2022. 1, 2

[75] Guy Tevet, Sigal Raab, Brian Gordon, Yoni Shafir, Daniel Cohen-or, and Amit Haim Bermano. Human motion diffusion model. In *ICLR*, 2023. 1, 2, 4, 6

[76] Jesse Thomason, Mohit Shridhar, Yonatan Bisk, Chris Paxton, and Luke Zettlemoyer. Language grounding with 3d objects. In *CoRL*, 2022. 3

[77] Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. Llama: Open and efficient foundation language models. *arXiv preprint*, 2023. 2

[78] Aaron van den Oord, Oriol Vinyals, and koray kavukcuoglu. Neural discrete representation learning. In *NeurIPS*, 2017. 2

[79] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. Attention is all you need. In *NeurIPS*, 2017. 6

[80] Jiashun Wang, Huazhe Xu, Jingwei Xu, Sifei Liu, and Xiaolong Wang. Synthesizing long-term 3d human motion and interaction in 3d scenes. In *CVPR*, 2021. 2, 6

[81] Jingbo Wang, Yu Rong, Jingyuan Liu, Sijie Yan, Dahua Lin, and Bo Dai. Towards diverse and natural scene-aware 3d human motion synthesis. In *CVPR*, 2022. 2, 6

[82] Xi Wang, Gen Li, Yen-Ling Kuo, Muhammed Kocabas, Emre Aksan, and Otmar Hilliges. Reconstructing action-conditioned human-object interactions using commonsense knowledge priors. In *3DV*, 2022. 2

[83] Zan Wang, Yixin Chen, Tengyu Liu, Yixin Zhu, Wei Liang, and Siyuan Huang. Humanise: Language-conditioned human motion generation in 3d scenes. In *NeurIPS*, 2022. 2, 3, 6

[84] Yan Wu, Jiahao Wang, Yan Zhang, Siwei Zhang, Otmar Hilliges, Fisher Yu, and Siyu Tang. Saga: Stochastic whole-body grasping with contact. In *ECCV*, 2022. 2

[85] Yiming Xie, Varun Jampani, Lei Zhong, Deqing Sun, and Huaizu Jiang. Omnicontrol: Control any joint at any time for human motion generation. *arXiv preprint*, 2023. 2

[86] 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

[87] Sijie Yan, Yuanjun Xiong, and Dahua Lin. Spatial temporal graph convolutional networks for skeleton-based action recognition. In *AAAI*, 2018. 6

[88] Zhengyuan Yang, Songyang Zhang, Liwei Wang, and Jiebo Luo. Sat: 2d semantics assisted training for 3d visual grounding. In *ICCV*, pages 1856–1866, 2021. 2, 3

[89] Shuquan Ye, Dongdong Chen, Songfang Han, and Jing Liao. 3d question answering. *IEEE Transactions on Visualization and Computer Graphics*, 2022. 3

[90] Zhihao Yuan, Xu Yan, Yinghong Liao, Ruimao Zhang, Sheng Wang, Zhen Li, and Shuguang Cui. Instancerefer: Cooperative holistic understanding for visual grounding on point clouds through instance multi-level contextual referring. In *ICCV*, 2021. 3

[91] He Zhang, Sebastian Starke, Taku Komura, and Jun Saito. Mode-adaptive neural networks for quadruped motion control. *ACM Trans. Graph.*, 2018. 2

[92] He Zhang, Yuting Ye, Takaaki Shiratori, and Taku Komura. Manipnet: Neural manipulation synthesis with a hand-object spatial representation. *ACM Trans. Graph.*, 2021. 5

[93] Jianrong Zhang, Yangsong Zhang, Xiaodong Cun, Yong Zhang, Hongwei Zhao, Hongtao Lu, Xi Shen, and Ying Shan. Generating human motion from textual descriptions with discrete representations. In *CVPR*, 2023. 2

[94] 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*, 2022. 2

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

[96] Yan Zhang and Siyu Tang. The wanderings of odysseus in 3d scenes. In *CVPR*, 2022. 2

[97] Yiming Zhang, ZeMing Gong, and Angel X Chang. Multi3drefer: Grounding text description to multiple 3d objects. In *ICCV*, 2023. 3

[98] Yaqi Zhang, Di Huang, Bin Liu, Shixiang Tang, Yan Lu, Lu Chen, Lei Bai, Qi Chu, Nenghai Yu, and Wanli Ouyang. Motiongpt: Finetuned llms are general-purpose motion generators. *arXiv preprint*, 2023. 2

[99] Hengshuang Zhao, Li Jiang, Jiaya Jia, Philip HS Torr, and Vladlen Koltun. Point transformer. In *ICCV*, 2021. 2

[100] Kaifeng Zhao, Shaofei Wang, Yan Zhang, Thabo Beeler, and Siyu Tang. Compositional human-scene interaction synthesis with semantic control. In *ECCV*, 2022. 3

[101] Kaifeng Zhao, Yan Zhang, Shaofei Wang, Thabo Beeler, and Siyu Tang. Synthesizing diverse human motions in 3d indoor scenes. In *ICCV*, 2023. 3

[102] Lichen Zhao, Daigang Cai, Lu Sheng, and Dong Xu. 3dvg-transformer: Relation modeling for visual grounding on point clouds. In *ICCV*, 2021. 3

[103] Yang Zheng, Yanchao Yang, Kaichun Mo, Jiaman Li, Tao Yu, Yebin Liu, C Karen Liu, and Leonidas J Guibas. Gimo: Gaze-informed human motion prediction in context. In *ECCV*, 2022. 3

[104] Yi Zhou, Connelly Barnes, Lu Jingwan, Yang Jimei, and Li Hao. On the continuity of rotation representations in neural networks. In *CVPR*, 2019. 3