ROAM: Robust and Object-aware Motion Generation using Neural Pose Descriptors

Wanyue Zhang[1,2] Rishabh Dabral[1] Thomas Leimkühler[1]
Vladislav Golyanik[†,1] Marc Habermann[†,1] Christian Theobalt[1,2]
[1]Max Planck Institute for Informatics, SIC
[2]Saarbrücken Research Center for Visual Computing, Interaction and AI

Abstract

Existing automatic approaches for 3D virtual character motion synthesis supporting scene interactions do not generalise well to new objects outside training distributions, even when trained on extensive motion capture datasets with diverse objects and annotated interactions. This paper addresses this limitation and shows that robustness and generalisation to novel scene objects in 3D object-aware character synthesis can be achieved by training a motion model with as few as one reference object. We leverage an implicit feature representation trained on object-only datasets, which encodes an $\mathsf{SE}(3)$*-equivariant descriptor field around the object. Given an unseen object and a reference pose-object pair, we optimise for the object-aware pose that is closest in the feature space to the reference pose. Finally, we use l-NSM, i.e., our motion generation model that is trained to seamlessly transition from locomotion to object interaction with the proposed bidirectional pose blending scheme. Through comprehensive numerical comparisons to state-of-the-art methods and in a user study, we demonstrate substantial improvements in 3D virtual character motion and interaction quality and robustness to scenarios with unseen objects. Our project page is available at* `https://vcai.mpi-inf.mpg.de/projects/ROAM/`.

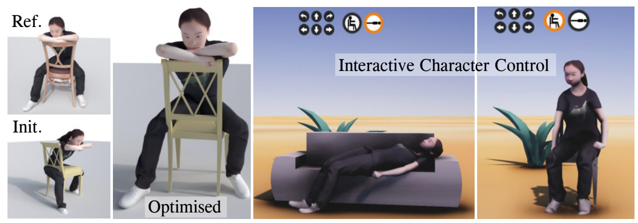

Figure 1. ROAM generates sitting and lying poses on a large variety of chairs and couches. It is $\mathsf{SE}(3)$-equivariant and robust to out-of-distribution objects of the same category. ROAM also allows interactive character control without requiring the model to be trained on mocap data with subjects interacting with diverse chairs and couches.

1. Introduction

Data-driven virtual 3D character animation has recently witnessed remarkable progress, and several neural methods modelling human motion have been proposed [19, 34, 36]. The realism of virtual characters is a core contributing factor to the quality of computer animations and user experience in immersive applications like games, movies, and VR/AR. In such settings, a virtual character often interacts with different assets in the environment (*e.g.,* pieces of furniture) and elaborate interactions usually require manual work of professional users. However, automatically synthesising natural animations with scene interaction constraints (as shown in Fig. 1) remains unsolved and challenging, especially when the virtual assets in the scene significantly differ from those captured during the data acquisition phase.

Approaches available in the literature enable scene-agnostic animation [2, 7, 9, 13, 18, 25, 27, 35, 40, 43, 44], fine-grained grasping synthesis [12, 28, 37–39, 48] and human motion and performance capture [3, 5, 8, 14, 22, 24, 32, 42]. In our context, several existing works on scene-aware character animation focus on specific interaction types such as *sitting* on a chair [16, 34, 36, 45]. Despite producing fairly natural animation for *known* objects, such methods struggle to generalise to unseen objects of the same category. Creating a generalisable motion model, which is capable of synthesising *object-aware* and *realistic* motions remains very challenging and unsolved. This is a major shortcoming due to the exponential number of possible asset-motion combinations. Naïvely training a generalisable neural network to synthesise sitting and lying motion would require collecting and annotating a large dataset of object-interaction motions with a large variety of chairs and couches. This is expensive and laborious. We believe all these challenges necessitate a fundamentally new approach to character-environment interaction modelling.

Hence, we propose ROAM, *i.e.,* an alternative solution that works around the requirement of large and diverse datasets while also providing a complete framework for vir-

[†]Equal advising/contribution and corresponding authors

tual character animation with object interaction; see Fig. 2 for an overview. ROAM is a robust and object-aware virtual character animation framework to synthesise human-object interactions; it generalises to unseen objects of the same category *without* relying on a large dataset of human-object animations. To this end, we follow a *divide-and-conquer* strategy to dataset collection. This affords us generalisation across instances of an object category while requiring character animation data with as few as *one* exemplar for an object category. We divide the problem into two subtasks: *goal pose estimation* and *goal-driven motion synthesis*.

Goal pose estimation is framed as a descriptor matching problem, which only requires a library of object shapes (such as ShapeNet [4] or ObjaVerse [10]) and a desired reference pose. By designing $\mathrm{SE}(3)$-invariant neural descriptors [33], we encode the geometric relations for human-object interaction. This allows us to optimise the reference pose for an unseen object by minimising the descriptor distance between the reference human-object pair and the pose for the target object.

Once the goal pose for a previously unseen target object is optimised, our proposed *l*-NSM auto-regressively animates the character's motion from a standing pose to the sitting and lying goal pose. Remarkably, *l*-NSM is trained using motion capture data with only a single exemplar object per category. This separation of tasks makes ROAM robust to variations in unseen object styles and scales while also avoiding expensive data capture overheads.

In summary, our technical contributions are as follows:

- ROAM, a new approach for synthesising virtual character-object animations in 3D, which generalises to unseen objects of the same type while avoiding the capture of large motion-capture datasets;
- A divide-and-conquer strategy simplifying the learning problem while also increasing the robustness of the method with respect to unseen object instances;
- A novel, $\mathrm{SE}(3)$-equivariant neural descriptor objective to generate the goal pose and its integration into *l*-NSM, producing realistic motions towards the goal pose (Sec. 3.1).

To evaluate our design choices, we record a dataset of sitting motion on a reference chair and a reference sofa as well as a lying down motion on a reference sofa, which we will make publicly available for future research. Through a comprehensive set of experiments and perceptual study, we show the effectiveness of our method in terms of motion plausibility and robustness to unseen chair and sofa types.

2. Related Work

We now discuss two closely-related method classes, *i.e.,* methods that generate *static* poses conditioned on the scene (Sec. 2.1) and methods supporting *dynamic* and scene-aware animations (Sec. 2.2). Tab. 1 provides a conceptual comparison of our method to previous works.

	w/o Man. Label	Robust	Single Object
NSM [34]	✗	✗	✗
SAMP [16]	✓	✗	✗
COUCH [45]	✗	✗	✗
Ours	✓	✓	✓

Table 1. Closely related works in scene-aware 3D human pose and motion synthesis. *Manual labelling* means that objects have to be manually annotated (*e.g.,* contact points) at inference time. *Robust* approaches can account for various unseen instances of the same object category. Finally, *single object* refers to the requirement of motion capture data with *one* object instance, *i.e.,* a single chair.

2.1. Scene-aware Human Pose Generation

This setting requires generating a plausible 3D pose of a character, given the scene geometry. In this regard, Zhang et al. [46] use a conditional Variational Auto-Encoder (cVAE) to estimate a pose conditioned on the latent space of the scene and further optimise it to achieve physical plausibility. Li et al. [21] build a large-scale dataset with scene affordance and use it to train a 3D generative model, which plausibly places humans in a scene. POSA [17] encodes contact probabilities between body mesh vertices and scene semantics. COIN [47] synthesises compositional human-object interaction given object semantics and intended action while requiring extensive labelling of objects and interactions on the PROX dataset [15]. In contrast to existing methods, we introduce a different paradigm for object-aware human pose generation. We also differ in that we are interested in synthesising dynamic motion sequences which can be controlled by a user.

2.2. Scene-aware Human Motion Generation

Dynamic, scene-aware motion generation requires a motion synthesis pipeline such that the characters perform a desired interaction with the scene. Towards this goal, Neural State Machine (NSM) [34] synthesises periodic and non-periodic human motion with scene interactions. It encodes the environment and the objects using a coarse volumetric representation which can often lack robustness for unseen objects. However, it is not $\mathrm{SE}(3)$-equivariant and therefore requires the user to *manually label* the goal position and orientation for the character to interact with the object. SAMP [16] improves upon NSM by predicting the goal location. However, it also uses the voxel-based scene representation, which limits the method's robustness and generalisability to novel instances of the same object category. COUCH [45] studies human–chair interaction by first sampling hand contact points on the chair using a VAE and then synthesising motion for the hands to reach the regressed target locations. This approach only allows partial variability and controllability as it provides no constraints to joints that are not in contact with the object. Moreover, it relies on a large dataset

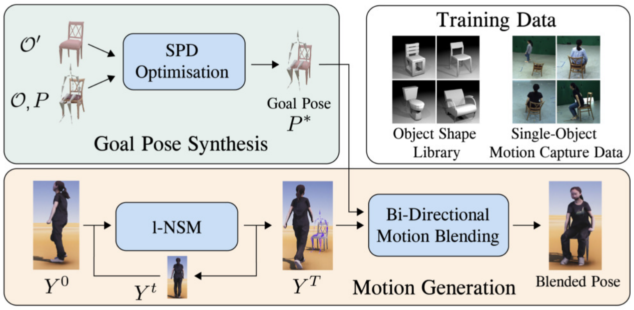

Figure 2. **Method overview.** Given a starting standing pose, a sitting pose on a reference chair as well as a novel object (not seen during training), our goal is to synthesise a character animation, which approaches the novel object (either chair or sofa) and then sits or lies on it while the motion should be aware of the shape of the novel object. We achieve this by our proposed three-stage procedure. First, we propose a skeletal pose field, which adapts the sitting pose on the reference chair to the novel chair in a shape-aware manner while not requiring any manual labelling. We call this pose the goal sitting pose. Second, *l*-NSM generates an approaching and interaction motion towards the novel object.

with labelled contact points and lacks generalisability to unseen objects. Wang et al. [41] tackles long-duration motion synthesis by first generating intermediate static poses and then synthesising short in-between motions. Scene constraints such as collision avoidance and affordance are incorporated during post-processing, prohibiting real-time performance.

We argue that previous methods on 3D human pose and motion generation either tend to be bound to object instances close to the ones seen during training [34] or rely on large and expensive data collections [16, 45]. In contrast, our method does not rely on extensive motion capture sequences with diverse objects while still achieving robust motion synthesis.

3. Method

Our goal is to synthesise 3D motion sequences of a character approaching and interacting with chairs and couches of various geometries, scales, and orientations. To learn such motions, we restrict ourselves to using motion-capture data with only one reference exemplar object per category; chairs and couches are considered separate categories. The overall schema of our method is shown in Fig. 2. We propose a *divide-and-conquer* strategy consisting of two main stages: First, our Goal Pose Synthesis (GPS) module takes a reference pose and an unseen target object and then generates an adapted goal pose, which fits the shape of the target object. We do so using our novel Skeletal Descriptor Fields (Sec. 3.1) which help the optimiser discover correspondences between the reference and the target object. With the goal pose synthesised, we need to seamlessly integrate it into our motion synthesis pipeline for low-level control. This is facilitated by our second component, lightweight Neural State Machine(*l*-NSM), that synthesises a natural walking motion approaching the object and interacting with it from the starting pose towards the final goal pose (Sec. 3.2). Finally, we propose an improved bidirectional blending scheme that seamlessly integrates the *l*-NSM motions with the goal pose.

The key motivation behind this two-stage design is that each task in separation can be trained with less data. More precisely, the goal pose synthesis solely requires an object dataset and *l*-NSM only requires motion-capture data paired with as few as one object per category. We now introduce the modules in detail.

3.1. Object-aware and Robust Goal Pose Synthesis

To estimate the goal pose that generalises to arbitrary shapes belonging to the same category, we first create a dataset of a character interacting with a reference object. Specifically, the dataset consists of several sequences of human motion, $\mathcal{P} = \{P_1, P_2, \dots\}$, where P_i is the character's pose at frame i. We parameterise the pose using axis-angles, *i.e.,* $P = \{\mathbf{t}_{\text{root}}, \mathbf{p}_j\}$ for $j = \{1, \dots, J\}$, where J is the number of joints, and $\mathbf{p}_j \in \mathbb{R}^3$ is the axis-angle representation of the rotation of joint j. The point cloud of the reference chair is denoted as $\mathcal{O} \in \mathbb{R}^{N\times 3}$ and, likewise, the target chair as $\mathcal{O}' \in \mathbb{R}^{N'\times 3}$, where N and N' are the number of points in the respective point clouds.

The main question we seek to address in this section is the following: Given a person's sitting pose on a chair of *known* geometry, how should this pose be adapted to a target chair with a previously unknown and different geometry? The target geometry could differ in several aspects, such as the height, type of armrest, or back support. Given a reference pose, P, of a character sitting or lying on the reference chair or sofa, $\mathcal{O}$, our goal is to optimise the target pose, P', for a previously unseen target object $\mathcal{O}'$. We formulate this as a descriptor-matching problem on 3D points around the object, which allows us to reduce the dataset size significantly.

Specifically, we train a neural network to estimate an SE(3)-equivariant descriptor field around a given 3D object [33]. Once trained, we can construct a descriptor, $\mathbf{Z}(\mathcal{O}, P)$, for the *skeleton* by aggregating the learned *point* descriptors on the 3D keypoint locations of the reference pose. These descriptors encode the spatial relationship between the character pose and object geometry. This is used to finally optimise for P' by minimising the discrepancy between $\mathbf{Z}(\mathcal{O}, P)$ and $\mathbf{Z}(\mathcal{O}', P')$. It is worth noting that such an approach does not require any manual labelling at inference, which is in contrast to prior work [34, 38]. In the

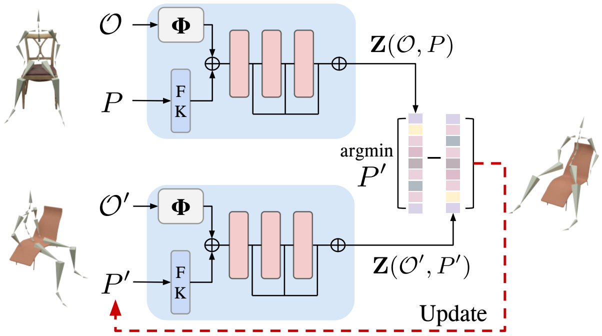

Figure 3. Goal Pose Synthesis (GPS) module. For the given reference object-pose pair and a target object, we generate neural descriptors $\mathbf{Z}(\mathcal{O}, P)$ and $\mathbf{Z}(\mathcal{O}', P')$ that represent the relative geometry of the object and the pose, thereby allowing us to optimise the target pose P' (Sec 3.1).

following subsections, we provide more details.

3.1.1 Neural Descriptor Fields (NDF)

Inspired by NDF [33], we learn a neural shape descriptor, Ψ on objects belonging to the chair and sofa categories of the ShapeNet dataset [4]. Ψ is modelled as a continuous coordinate-based MLP, which maps a 3D point $\mathbf{x} \in \mathbb{R}^3$ to $\mathbf{z}_\mathbf{x}$, conditioned on an input object $\mathcal{O}$ as:

$$\mathbf{z}_\mathbf{x} = \Psi(\mathbf{x}, \mathcal{O}). \tag{1}$$

As shown in Fig. 3, Ψ consists of a PointNet [29] encoder Φ and an occupancy network [26] Ω. Φ encodes the object $\mathcal{O}$ into a compact latent code that informs the occupancy network of the object geometry. Ω is conditioned on the object's latent code and a query point, $\mathbf{x}$, which allow it to predict the occupancy flag at the query point:

$$\Omega(\mathbf{x}, \Phi(\mathcal{O})) \implies [0, 1]. \tag{2}$$

Since NDF's architecture is based on VectorNeurons [11], the descriptor is by design $\mathsf{SE}(3)$-equivariant. As the Ω network is trained to estimate the occupancy values, the intermediate features implicitly model object-centric boundary information and spatial relationships between the points. Further, since Ψ is a coordinate-based MLP, it allows us to extract these object-related features at an arbitrary point in space. To obtain the descriptor for a point $\mathbf{x}$, the activations from L layers of Ω are concatenated so that it contains hierarchical features:

$$\mathbf{z}_\mathbf{x} = \oplus_{i=0}^{L} \Omega^i(\mathbf{x}, \Phi(\mathcal{O})). \tag{3}$$

By matching the point descriptors in two different objects, one can find correspondences between them.

However, this descriptor, in the described form, only enables to rigidly match a coordinate frame at $\mathbf{x}$ from a source object to a target one, whereas we are interested in transferring the skeletal pose interacting with a source object to a target one. Therefore, we introduce our skeletal pose descriptor (SPD) in the following.

3.1.2 Skeletal Pose Descriptor (SPD)

Our core idea is to aggregate a neural descriptor Ψ by querying all kinematic joints, which can be defined as

$$\mathbf{Z}(\mathcal{O}, P) = \{\mathbf{z}_\mathbf{x} | \mathbf{x} \in \mathcal{F}(P)\}, \tag{4}$$

where $\mathcal{F}(\cdot)$ is the differentiable forward kinematics function that outputs the set of 3D joint positions for a given pose. However, constructing $\mathbf{Z}$ by querying Ψ at only the 3D joint positions makes it sensitive to the descriptor's quality at that specific point. To make it robust to such outliers, we also sample the query points from the neighbourhood of each joint position. Specifically, the points are sampled from a Gaussian distribution with standard deviation σ centred at the joint positions $\mathcal{F}(P)_j$:

$$\mathbf{Z}(\mathcal{O}, P) = \{\mathbf{z}_{\mathbf{x_j}} | \mathbf{x_j} \sim \mathcal{N}(\mathcal{F}(P)_j, \sigma)\}_{j=0}^{J}. \tag{5}$$

Note that our descriptor $\mathbf{Z}$ is now a function of the skeletal pose P. Moreover, as each joint descriptor $\mathbf{z}_\mathbf{x}$ is $\mathsf{SE}(3)$–equivariant, it follows that our pose descriptor $\mathbf{Z}$ is $\mathsf{SE}(3)$–equivariant. In practice, this means that our method is robust to any rigid transform applied to the object, which is in contrast to previous work [34, 38] that assumes orientation and translation of the object are known in advance.

3.1.3 Goal Pose Optimisation

So far, we constructed a descriptor that is robust to rigid transforms and shape variations. However, ultimately we are interested in matching a pose P from a source object $\mathcal{O}$ to an unseen target object $\mathcal{O}'$. For this, we use the 3D skeleton pose descriptors $\mathbf{Z}$ defined above to find the optimal pose P^* for $\mathcal{O}'$. Recall that our forward kinematics function $\mathcal{F}(P)$ in (5) is differentiable, thereby allowing any gradients from the descriptor to also back-propagate into P.

We, thus, formulate pose optimisation as an energy minimisation problem with the underlying postulate that the descriptors encode the geometric relationship between the pose P and the object point cloud $\mathcal{O}$:

$$E_d = \sum_{j=1}^{J} \omega_j \big| \mathbf{Z}(\mathcal{O}', P')_j - \mathbf{Z}(\mathcal{O}, P)_j \big|.$$

ω_j is a weight assigned to each joint descriptor. For more details, please refer to Sec. C in the supplementary material.

Although the descriptor matching term, E_d, acts as the main data term driving the optimisation, it is limited in that it does not encode the human-body specific priors in it. After all, the descriptor is based on an occupancy network trained on a dataset of chairs and couches. Therefore, we introduce additional regularisation terms to keep P^* from drifting away from the manifold of plausible poses. Concretely, we introduce an angle-limit constraint (E_a), a foot-floor penetration regulariser (E_f), and a pose regulariser (E_p). Our overall objective function is defined as:

$$P^* = \underset{P'}{\operatorname{argmin}} \left(E_d + \lambda_a E_a + \lambda_f E_f + \lambda_p E_p \right). \quad (6)$$

We now discuss each of them in detail.

Angle Constraints. The SPD term, E_d can result in unnatural twists between joints when source and target geometries vary significantly. Similar to Shimada et al. [31], we penalise joint angles that are beyond a fixed bound:

$$E_a = \lambda_a \sum_j^J \mathcal{A}(P'_j), \text{ with} \quad (7)$$

$$\mathcal{A}(P'_j) = \begin{cases} (P'_j - A_{j,max})^2, & \text{if } P'_j > A_{j,max} \\ (A_{j,min} - P'_j)^2, & \text{if } P'_j < A_{j,min} \\ 0 & \text{otherwise.} \end{cases} \quad (8)$$

P'_j is the joint rotation (Euler angles) and $A_{j,max}, A_{j,min}$ denote the upper and lower bounds of a joint rotation.

Floor Penetration. The floor penetration term penalises the penetration distance from the ground plane to all joints that are below the ground and is formulated as:

$$E_f = \lambda_f \sum_j^J \|(\min(\mathcal{F}(P_j).y, \mathcal{G}_y) - \mathcal{G}_y)\|^2. \quad (9)$$

$\mathcal{F}(P)_j.y$ denotes the y-magnitude of joint j and $\mathcal{G}_y$ is the height of the ground.

Pose Regularisation. Note that for symmetric objects, there can be more than one location with similar descriptor features. For example, consider a left-right symmetric chair. The features on the right leg of the chair can be very similar to the features on the left leg due to similar geometric patterns and spatial relationship with the rest of the chair. Without further constraints, using an L1-loss in isolation would give us non-unique goal poses, which fall inside the local minima of the energy landscape. Additionally, we observe the optimisation may converge to an implausible sitting pose if the reference pose is very challenging (see Fig. 6-(c)). Thus, we introduce a pose regularisation term E_r below that ensures that the target pose does not deviate drastically from the reference pose in the canonical space:

$$E_r = \lambda_r \|\mathcal{F}(P'_c) - \mathcal{F}(P_c)\|^2, \quad (10)$$

where the subscript c denotes the canonicalised poses after aligning their root translation and orientation. However, we set the weight λ_r reasonably low for all experiments as a large weight can inhibit pose adaptation to novel shapes.

3.2. Interactive Motion Generation

After having obtained P^*, we synthesise transitions from a start (standing) pose to the goal pose. To that end, we use a motion generation network which autoregressively synthesises idling, walking, sitting and transitioning motion between the three actions. Inspired by Starke *et al.* [34], we propose a *lightweight* Neural State Machine (*l*-NSM). While NSM [34] defines interaction goal as a 6D vector encoding the goal position and orientation of the character's root, such an approach does not scale to our setting, which requires a full-body, 75DoF goal pose. Furthermore, we handle goal pose conditional object interaction (sitting down or lying down) and non-interaction locomotion without goal pose (walking or idling) in a single framework. Finally, since our goal pose is already object-aware, we do not use the environment and interaction sensors, thereby making the *l*-NSM *lighter*. As a result, our *l*-NSM consists of the frame encoder and the goal encoder.

The frame encoder encodes the character's current pose and trajectory within the $[-1s, 1s]$ time window $\mathcal{M}^t$ in an encoding Φ_f (in the following, we omit the superscript t). Specifically, it encodes the character's joint positions $\mathbf{M}_p \in \mathbb{R}^{3J}$, joint rotations $\mathbf{M}_q \in \mathbb{R}^{6J}$ and the linear joint velocities $\mathbf{M}_v \in \mathbb{R}^{3J}$. It also takes as input the root trajectory information such as the ground-plane projections of the root, $\mathbf{M}_r \in \mathbb{R}^{2L}$, for the past and future L frames as well as the corresponding 2D orientation vectors $\mathbf{M}_o \in \mathbb{R}^{2L}$. Finally, in encodes the soft action label $\mathbf{M}_a \in \mathbb{R}^4$ corresponding to idle, walking, sitting and standing actions. The goal encoder Φ_g encodes the target goal pose $\mathcal{G}^t$. The input to Φ_g includes the goal-pose's joint positions $\mathbf{G}_p$ as well as the ground-truth goal trajectory $\mathbf{G}_r \in \mathbb{R}^{3L}$ and orientations $\mathbf{G}_o \in \mathbb{R}^{3L}$. L is sampled from the future $[0, 2s]$ time window. Similar to Φ_f, Φ_g also encodes the action labels $\mathbf{G}_a$ but with one-hot encoding. At test time, $\mathbf{G}_r$ is the same as $\mathcal{F}(P^*)$. Let $\mathbb{M}_c = (\mathbf{M}_p, \mathbf{M}_q, \mathbf{M}_v)$, $\mathbb{M}_{ro} = (\mathbf{M}_r, \mathbf{M}_o)$, $\mathbb{G}_{roa} = (\mathbf{G}_r, \mathbf{G}_o, \mathbf{G}_a)$. We then have

$$\Phi_f : \mathbb{R}^{|\mathcal{M}^t|} \to \mathbb{R}^d, \quad \mathcal{M}^t = (\mathbb{M}_c, \mathbb{M}_{ro}, \mathbf{M}_a), \quad (11)$$

$$\Phi_g : \mathbb{R}^{|\mathcal{G}^t|} \to \mathbb{R}^d, \quad \mathcal{G}^t = (\delta \mathbf{G}_p, \mathbb{G}_{roa}), \quad (12)$$

where δ indicates whether the character is in locomotion mode ($\delta = 0$) or interaction mode ($\delta = 1$). Given the two encoders projecting the character's current state and the target state into a d-dimensional latent space, the *l*-NSM decoder, $\Psi : \mathbb{R}^{2d} \to \mathbb{R}^{|Y^{t+1}|}$, then decodes them into the motion for the next frame with:

$$Y^{t+1} = \left(\mathbb{M}_c, \hat{\mathbb{M}}_c^{t+1}, \mathbb{M}_{ro}^{t+1}, \hat{\mathbb{M}}_{ro}^{t+1}, \mathbb{G}_{roa}^{t+1}, c^{t+1}, f^{t+1} \right). \quad (13)$$

$\hat{\mathbb{M}}_c$ denotes the inverse pose defined relative to the goal coordinate system. Likewise, $\hat{\mathbb{M}}_{ro}$ denotes the inverse, goal-centric trajectory. $c_t \in \{0, 1\}$ are the floor-contact predictions which, if 1, trigger the post-processing inverse-kinematics routine to reduce footsliding. The decoder also estimates the phase $f_t \in \mathbb{R}^L$ of the gait which is fed into the gating network outputting the blending coefficients of the experts. We refer the reader to Starke et al. [34] for details.

Bidirectional Pose Blending. Handling unconditional locomotion and goal-conditioned transitions in an integrated architecture is challenging. Simply conditioning *l*-NSM on P^* proves to be insufficient to accurately reach it. In practice, we observe that the network generates a series of poses resembling the training sequence instead of reaching the unseen goal poses provided by GPS. Moreover, switching the animation into interaction mode often leads to sudden instabilities in the generated motion, as the transition from locomotion mode to interaction mode is abrupt. To address these artefacts, we adapt the bidirectional control routine proposed by Starke et al. [34] to our setting. The core idea of bidirectional control is to blend the ego-centric forward motion $\mathbb{M}_c$ and trajectory $\mathbb{M}_{ro}$ with the estimated goal-centric inverse motion $\hat{\mathbb{M}}_c$ and $\hat{\mathbb{M}}_{ro}$. These two predictions should ideally lead to the same update after transformation to the world space but often differ in inference due to autoregressive error accumulation and time needed to adjust to user input. Therefore, the forward and the reverse poses (and trajectories) are blended using: $\mathbb{M}'_p = \mathbb{M}_p + \lambda_b T \hat{\mathbb{M}}_p$, where T is the transformation matrix between the goal and ego-centric coordinate systems, and λ_b is the blending weight derived from the action prediction and distance to goal (see Sec. C in supplementary material).

Recall that our goal pose, P^* is automatically computed by the GPS module. This requires us to define a per-joint goal coordinate system where the origin lies at the joint's position, which we call the goal-joint coordinate system during object interaction. The orientation is computed by recovering the normal of the torso-plane of $\mathbf{P}^*$. As a result, the proposed Bidirectional Pose Blending elegantly generates seamless object interaction and locomotion. It is worth noting that we do so with minimal overheads to already large input and output dimensions of *l*-NSM. Once the goal pose is computed, *l*-NSM synthesises motion in real time and allows both low-level interactive control and high-level goal-conditioned motion synthesis.

4. Experiments

We compare our ROAM to its most related approaches, i.e, NSM [34], SAMP [16] and COUCH [45] with models retrained on our data, and evaluate it in terms of the animation realism and the plausibility of the synthesised sitting poses. NSM needs to be provided with a point of contact labels and the object orientations; COUCH assumes a predefined chair orientation), while we do not make any of such assumptions. All the methods are provided a character's starting

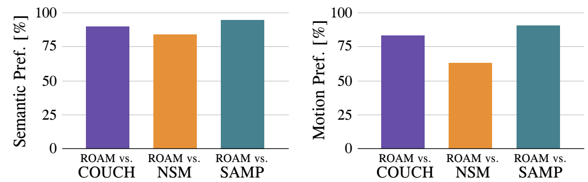

Figure 4. Results of the perceptual study for qualitative comparison of ROAM against the state of the art. Our method is consistently preferred over all other approaches, in terms of, both, semantic coherence with the chair's geometry, and motion realism.

pose and orientation along with the point cloud of the chair in the scene. All of them receive the same initial state of the character and the environment. The evaluation settings and their corresponding metrics are discussed next. For detailed visual comparisons with the state-of-the-art methods, the reader is referred to the supplementary video.

4.1. Perceptual User Study

Recall that NSM, COUCH and SAMP animations are conditioned on the points of contact alone and often do not adapt well to object geometries. Such cases are hard to numerically account for, and hence, necessitate a perceptual user study to evaluate the quality of motion synthesis. The participants are asked to compare 12 randomly generated animation sequences in a forced-choice manner, *i.e*, the participant must choose the best motion sequence out of the presented sequences. We ask the participants the following two questions for each set of motions: *Which motion do you prefer in terms of the naturalness of the motion? (Realism)* and *Which motion do you prefer in terms of the agreement of the sitting motion with the chair geometry? (Semantics)*

Results: We received responses from 50 participants, leading to 600 valid comparisons in total (200 per competing method); Fig 4 summarises the results. Our method is preferred over NSM, COUCH and SAMP by a significant margin. We calculate p-values (binomial test) for our comparison with each method and observe statistically significant results with $p<0.001$ for all three methods. Interestingly, NSM's motion generation performs better than the other two more recent methods in terms of user preference. It is worth noting that NSM chairs have been hand-labelled for the goal position whereas our motion synthesis relies on the automatically generated goal pose.

4.2. Quantitative Results

We next perform quantitative analysis to evaluate our design choices. However, establishing a firm evaluation metric for our setting is challenging as no single one would account for all aspects.

Goal Distance Error. Here, we attempt to evaluate whether the motion synthesis module reaches the intended target by

Figure 5. Qualitative results of the Skeletal Pose Descriptor-based goal pose optimisation (Sec. 3.1). Given a reference pose for a reference chair (top row), our method can adjust the reference pose to an unseen chair (second and third row). We show a variety of poses in the range from sitting on chairs (first column), sitting on sofas (middle column) and lying on sofas (last column). Please watch our supplementary video for animations of the optimisation process.

Method	Hip PE↓	Hand PE↓	Avg PE↓
NSM [34]	2.40	-	-
SAMP [16]	3.15	-	-
COUCH [45]	10.85	10.94	-
Ours	**1.17**	**4.78**	**3.95**

Table 2. Position Error (PE) Comparison with the state-of-the-art methods on the ability to accurately reach the goal. For us, we consider the goal pose's hip position and hand positions as the target to compute Hip PE and Hand PE. For COUCH, the generated hand contact labels are considered the target with our annotated goal orientation. For SAMP, we use the predicted hip position and direction as the goal. Finally, for NSM we compute the error with respect to the manually labelled target goal position.

measuring the Position Error (PE) to the goal. This is done in three settings; see Tab. 2. For comparison with NSM, we consider the manually labelled target position on the chair or couch as the goal position and compute the distance of the hip joint to it (Hip PE in Tab. 2). For SAMP, we use their generated hip positions and directions. Next, since COUCH facilitates reaching a goal pose based on the hand contacts, we use the generated hand contact positions and measure their distance from the hand positions in the synthesised sitting motion (Hand PE in Tab. 2). In the above setting, we choose the 'hip' joint and the 'hand' joints of P^* as our target positions, respectively. Finally, we evaluate our full-body per-joint position error between the goal pose P^* and the *l*-NSM output (Avg PE in Tab. 2).

4.3. Qualitative Results

In Fig. 5, we show how GPS adapts the reference pose to the target geometry for sitting on a chair and sofa and lying on a sofa.

Another interesting analysis is to measure the difference between the optimised novel pose and the reference pose as the target chair becomes more distant from the reference chair in terms of the Chamfer distance. To perform this analysis, we align the position and orientation of 100 chairs in the ShapeNet dataset [4] and use a single reference pose to generate goal poses for each chair. We then compute the difference between the reference and the target pose in terms of their root translation, root orientation and root-aligned 3D joint positions. The scatter plot in Fig. 7 follows the expected trend and we observe that the goal pose has to adjust significantly as the target chair's geometry deviates from that of the reference chair.

4.4. Ablation

In Fig. 6, we present ablation visualisations of three design choices for two poses of different difficulty. In the first ablation, we discard the proposed sequential three-stage optimisation routine (see Sec. C in supplementary material) and instead jointly optimise for the root orientation, root translation and root-relative articulations. We notice that doing so often leads to sub-optimal results because in the beginning of the optimisation, the reference pose can be in a significantly different orientation and the corresponding gradients are expected to be uninformative for optimising

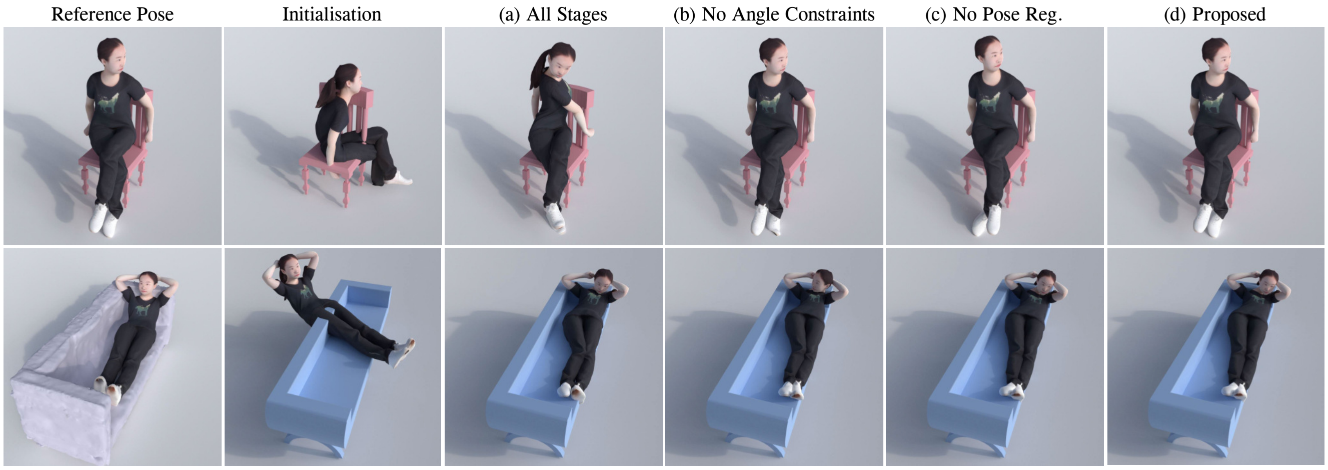

Figure 6. We perform three ablation experiments for goal pose optimisation. (a) *All Stages* optimises all DoF simultaneously with all energy terms added from the first iteration. This strategy leads to twists in body joints. (d) Our proposed strategy with three-stage optimisation, E_a and E_r. We empirically find this combination leads to the best qualitative results.

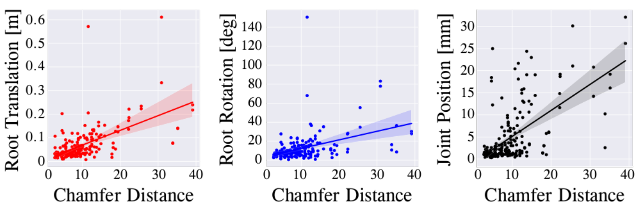

Figure 7. A plot depicting the difference between the optimised goal pose and the reference pose as the Chamfer distance between the target couch and the reference couch changes across the test couches. For each optimised goal pose P^*, we measure the change in the root translation (left), root orientation (centre), and the root-normalised joint position error (right) with respect to the initial reference pose. We observe that P^* remains robust to large variations in the target object's geometry, as measured by the Chamfer distance.

joint articulations. Next, we evaluate the effect of adding angle constraints E_a in our loss function in Eq. (6). E_a penalises implausible joint configuration and leads to more natural results, especially for challenging poses and out-of-distribution chairs. We also evaluate the choice of including the pose regulariser E_r on the joint positions in the canonical space. This regulariser becomes important to ensure that the optimisation does not stuck in local minima and often leads to more natural poses. Finally, the rightmost column of Fig. 6 shows that the proposed design choices lead to improved results.

5. Discussion and Conclusion

We presented ROAM, a framework for robust synthesis of human-object interactions. Focusing on chairs and couches as an important instance arising in many applications, we demonstrated that the proposed two-step strategy of ROAM allows robust generalisation to a variety of unseen objects.

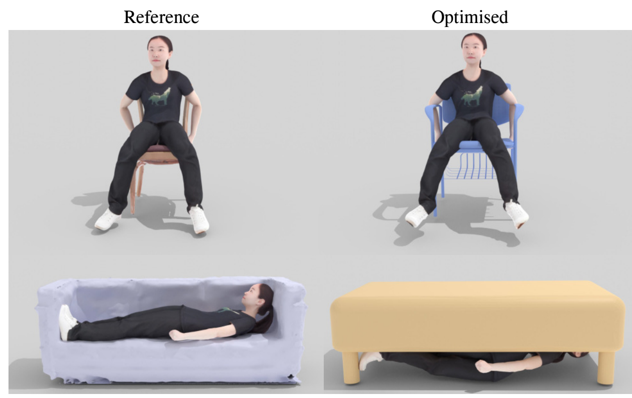

Figure 8. Limitations. The top row shows the challenging scenario of adapting a pose to a target chair with armrests. Since the correspondences are not well-defined, our optimised pose has penetration at the elbow region. The bottom row shows that SPD optimisation leads to a physically implausible pose.

Furthermore, we showed that the proposed bidirectional pose blending not only allows the motion synthesis pipeline to smoothly transition to the goal pose but also generates natural motion that the users found to be better than existing scene interaction methods. Crucially, we could achieve this by training with motion capture data involving only a *single* chair/couch exemplar. However, our method leads to artefacts when the target geometry differs significantly from the source geometry (Fig. 8). We believe these artefacts can be removed by leveraging character mesh instead of the skeleton representation with physical plausibility constraints (see supplementary material Sec. E for more discussion on future work). With the ever-growing demand for realistic 3D characters in virtual worlds, we envision our principled and practical approach to contribute to the creation of scalable immersive visual experiences.

References

[1] The Captury. https://captury.com, 2023. 11

[2] Emre Aksan, Manuel Kaufmann, and Otmar Hilliges. Structured prediction helps 3d human motion modelling. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 1

[3] Bharat Lal Bhatnagar, Xianghui Xie, Ilya Petrov, Cristian Sminchisescu, Christian Theobalt, and Gerard Pons-Moll. Behave: Dataset and method for tracking human object interactions. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 1

[4] Angel X. Chang, Thomas A. Funkhouser, Leonidas J. Guibas, Pat Hanrahan, Qi-Xing Huang, Zimo Li, Silvio Savarese, Manolis Savva, Shuran Song, Hao Su, Jianxiong Xiao, Li Yi, and Fisher Yu. Shapenet: An information-rich 3d model repository. *Computing Research Repository (CoRR)*, 2015. 2, 4, 7

[5] Yixin Chen, Siyuan Huang, Tao Yuan, Siyuan Qi, Yixin Zhu, and Song-Chun Zhu. Holistic++ scene understanding: Single-view 3d holistic scene parsing and human pose estimation with human-object interaction and physical commonsense. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 1

[6] Erwin Coumans and Yunfei Bai. Pybullet, a python module for physics simulation for games, robotics and machine learning. 2016. 12

[7] Qiongjie Cui, Huaijiang Sun, and Fei Yang. Learning dynamic relationships for 3d human motion prediction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 1

[8] Rishabh Dabral, Soshi Shimada, Arjun Jain, Christian Theobalt, and Vladislav Golyanik. Gravity-aware monocular 3d human-object reconstruction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 1

[9] Rishabh Dabral, Muhammad Hamza Mughal, Vladislav Golyanik, and Christian Theobalt. Mofusion: A framework for denoising-diffusion-based motion synthesis. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1

[10] Matt Deitke, Dustin Schwenk, Jordi Salvador, Luca Weihs, Oscar Michel, Eli VanderBilt, Ludwig Schmidt, Kiana Ehsani, Aniruddha Kembhavi, and Ali Farhadi. Objaverse: A universe of annotated 3d objects. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 2

[11] Congyue Deng, Or Litany, Yueqi Duan, Adrien Poulenard, Andrea Tagliasacchi, and Leonidas J Guibas. Vector neurons: A general framework for so (3)-equivariant networks. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 4

[12] Anindita Ghosh, Rishabh Dabral, Vladislav Golyanik, Christian Theobalt, and Philipp Slusallek. Imos: Intent-driven full-body motion synthesis for human-object interactions. In *Eurographics*, 2023. 1

[13] Partha Ghosh, Jie Song, Emre Aksan, and Otmar Hilliges. Learning human motion models for long-term predictions. In *2017 International Conference on 3D Vision (3DV)*, 2017. 1

[14] Vladimir Guzov, Aymen Mir, Torsten Sattler, and Gerard Pons-Moll. Human poseitioning system (hps): 3d human pose estimation and self-localization in large scenes from body-mounted sensors. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 1

[15] Mohamed Hassan, Vasileios Choutas, Dimitrios Tzionas, and Michael J Black. Resolving 3d human pose ambiguities with 3d scene constraints. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 2

[16] Mohamed Hassan, Duygu Ceylan, Ruben Villegas, Jun Saito, Jimei Yang, Yi Zhou, and Michael J Black. Stochastic scene-aware motion prediction. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2021. 1, 2, 3, 6, 7

[17] Mohamed Hassan, Partha Ghosh, Joachim Tesch, Dimitrios Tzionas, and Michael J Black. Populating 3d scenes by learning human-scene interaction. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 2

[18] Daniel Holden, Jun Saito, and Taku Komura. A deep learning framework for character motion synthesis and editing. *ACM Transactions on Graphics (TOG)*, 2016. 1

[19] Daniel Holden, Taku Komura, and Jun Saito. Phase-functioned neural networks for character control. *ACM Transactions on Graphics (TOG)*, 2017. 1

[20] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations (ICLR)*, 2015. 11

[21] Xueting Li, Sifei Liu, Kihwan Kim, Xiaolong Wang, Ming-Hsuan Yang, and Jan Kautz. Putting humans in a scene: Learning affordance in 3d indoor environments. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 2

[22] Miao Liu, Dexin Yang, Yan Zhang, Zhaopeng Cui, James M Rehg, and Siyu Tang. 4d human body capture from egocentric video via 3d scene grounding. In *2021 International Conference on 3D Vision (3DV)*, 2021. 1

[23] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. In *International Conference on Learning Representations*, 2017. 11

[24] Diogo C. Luvizon, Marc Habermann, Vladislav Golyanik, Adam Kortylewski, and Christian Theobalt. Scene-aware 3d multi-human motion capture from a single camera, 2023. 1

[25] Julieta Martinez, Michael J. Black, and Javier Romero. On human motion prediction using recurrent neural networks. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 1

[26] Lars Mescheder, Michael Oechsle, Michael Niemeyer, Sebastian Nowozin, and Andreas Geiger. Occupancy networks: Learning 3d reconstruction in function space. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2019. 4

[27] Dario Pavllo, David Grangier, and Michael Auli. Quaternet: A quaternion-based recurrent model for human motion. In *British Machine Vision Conference (BMVC)*, 2018. 1

[28] Sergey Prokudin, Christoph Lassner, and Javier Romero. Efficient learning on point clouds with basis point sets. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, 2019. 1

[29] Charles R Qi, Hao Su, Kaichun Mo, and Leonidas J Guibas. Pointnet: Deep learning on point sets for 3d classification and segmentation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017. 4

[30] Soshi Shimada, Vladislav Golyanik, Weipeng Xu, and Christian Theobalt. Physcap: Physically plausible monocular 3d motion capture in real time. *ACM Transactions on Graphics (TOG)*. 12

[31] Soshi Shimada, Vladislav Golyanik, Weipeng Xu, Patrick Pérez, and Christian Theobalt. Neural monocular 3d human motion capture with physical awareness. *ACM Transactions on Graphics (TOG)*, 2021. 5

[32] Soshi Shimada, Vladislav Golyanik, Zhi Li, Patrick Pérez, Weipeng Xu, and Christian Theobalt. Hulc: 3d human motion capture with pose manifold sampling and dense contact guidance. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 1

[33] Anthony Simeonov, Yilun Du, Andrea Tagliasacchi, Joshua B Tenenbaum, Alberto Rodriguez, Pulkit Agrawal, and Vincent Sitzmann. Neural descriptor fields: Se (3)-equivariant object representations for manipulation. In *2022 International Conference on Robotics and Automation (ICRA)*, 2022. 2, 3, 4

[34] Sebastian Starke, He Zhang, Taku Komura, and Jun Saito. Neural state machine for character-scene interactions. *ACM Transactions on Graphics (TOG)*, 2019. 1, 2, 3, 4, 5, 6, 7

[35] Sebastian Starke, Ian Mason, and Taku Komura. Deepphase: Periodic autoencoders for learning motion phase manifolds. *ACM Transactions on Graphics (TOG)*, 2022. 1

[36] Sebastian Starke, Ian Mason, and Taku Komura. Deepphase: Periodic autoencoders for learning motion phase manifolds. *ACM Transactions on Graphics (TOG)*, 2022. 1

[37] Omid Taheri, Nima Ghorbani, Michael J Black, and Dimitrios Tzionas. Grab: A dataset of whole-body human grasping of objects. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 1

[38] Omid Taheri, Vasileios Choutas, Michael J Black, and Dimitrios Tzionas. Goal: Generating 4d whole-body motion for hand-object grasping. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2022. 3, 4

[39] Purva Tendulkar, Dídac Surís, and Carl Vondrick. Flex: Full-body grasping without full-body grasps. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023. 1

[40] Hongsong Wang, Jian Dong, Bin Cheng, and Jiashi Feng. Pvred: A position-velocity recurrent encoder-decoder for human motion prediction. *IEEE Transactions on Image Processing*, 2021. 1

[41] Jiashun Wang, Huazhe Xu, Jingwei Xu, Sifei Liu, and Xiaolong Wang. Synthesizing long-term 3d human motion and interaction in 3d scenes. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 3

[42] Zhenzhen Weng and Serena Yeung. Holistic 3d human and scene mesh estimation from single view images. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021. 1

[43] Jingwei Xu, Huazhe Xu, Bingbing Ni, Xiaokang Yang, Xiaolong Wang, and Trevor Darrell. Hierarchical style-based networks for motion synthesis. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 1

[44] Ye Yuan and Kris Kitani. Dlow: Diversifying latent flows for diverse human motion prediction. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020. 1

[45] Xiaohan Zhang, Bharat Lal Bhatnagar, Sebastian Starke, Vladimir Guzov, and Gerard Pons-Moll. Couch: towards controllable human-chair interactions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 1, 2, 3, 6, 7

[46] Yan Zhang, Mohamed Hassan, Heiko Neumann, Michael J Black, and Siyu Tang. Generating 3d people in scenes without people. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020. 2

[47] Kaifeng Zhao, Shaofei Wang, Yan Zhang, Thabo Beeler, and Siyu Tang. Compositional human-scene interaction synthesis with semantic control. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 2

[48] Keyang Zhou, Bharat Lal Bhatnagar, Jan Eric Lenssen, and Gerard Pons-Moll. Toch: Spatio-temporal object-to-hand correspondence for motion refinement. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2022. 1