Placing Human Animations into 3D Scenes by Learning Interaction- and Geometry-Driven Keyframes

James F. Mullen Jr
University of Maryland
mullenj@umd.edu

Divya Kothandaraman
University of Maryland
dkr@umd.edu

Aniket Bera
Purdue University
ab@cs.purdue.edu

Dinesh Manocha
University of Maryland
dmanocha@umd.edu

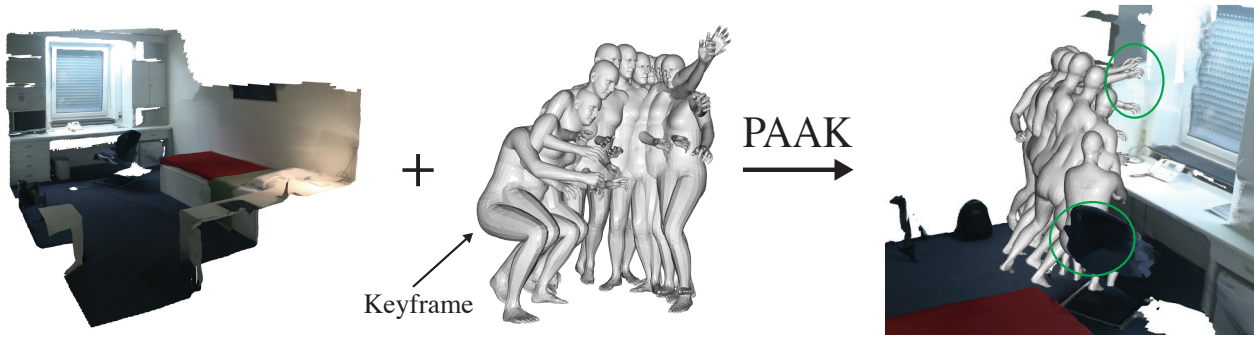

Figure 1: Our goal is to place animations, a 3D sequence of human motion, into a 3D scene while maintaining any interactions with the scene the animation contains. First, we select "keyframes," the most important meshes in the animation for modeling interactions with the scene. In the animation, the leftmost mesh where the human is sitting would be a keyframe. We then use the keyframes to find a placement in the scene that best matches the interactions in the animation (green circles, right).

Abstract

We present a novel method for placing a 3D human animation into a 3D scene while maintaining any human-scene interactions in the animation. We use the notion of computing the most important meshes in the animation for the interaction with the scene, which we call "keyframes." These keyframes allow us to better optimize the placement of the animation into the scene such that interactions in the animations (standing, laying, sitting, etc.) match the affordances of the scene (e.g., standing on the floor or laying in a bed). We compare our method, which we call PAAK, with prior approaches, including POSA, PROX ground truth, and a motion synthesis method, and highlight the benefits of our method with a perceptual study. Human raters preferred our PAAK method over the PROX ground truth data 64.6% of the time. Additionally, in direct comparisons, the raters preferred PAAK over competing methods including 61.5% compared to POSA. Our project website is available at `https://gamma.umd.edu/paak/`.

1. Introduction

Throughout daily life, humans interact with their environment by making contact with objects and avoiding collisions with obstacles. Imagine you are sitting at your desk. Your arms may be resting on the desk. When you stand up to leave, you set your hands on the chair and desk as you walk around them. To leave the room you grab a doorknob to open the door. Interactions like these define how humans move throughout their environment. In this paper, our goal is to utilize these interactions to place animations into a scene in the most natural way.

Applications in synthetic data generation, virtual reality (VR), augmented reality (AR)[13], game design [30] and human-robot interaction [27] need to consider interactions between humans and the environment while placing 3D human animations into 3D scenes. For example, an AR designer may want to populate a living room environment with people sitting on chairs or couches, navigating the space, and having a conversation in the corner. Current methods

are based on animators using modeling or animation tools to generate such sequences, but this can be very time consuming and it requires animators with considerable experience using these tools.

Prior work on human-scene interaction, or scene affordances, attempts to place 3D human models into 3D scene scans such that the placement matches real human behavior. Some methods focus on placing existing static human models into the scene [20, 19, 14, 13] while others attempt to generate a suitable static human model[60, 59, 12]. However, many of these methods typically do not generalize well to new scenes and none work with 3D human animations. A recent method, POSA [13], utilizes a cVAE [44] to encode contact probabilities and semantic labels onto the vertices of a human model which can then be used to place it in the scene convincingly. This approach enables the generalization of the human models to any possible scene. POSA, and most other human-scene interaction methods, exclusively work with static, single-pose humans, not animations as we are interested in with our work.

1.1. Main Contributions

We propose a novel method to *place an existing 3D human animation into any arbitrary, static, 3D scene with natural-looking interactions*. We can use any human animations that are, or can be represented as, a time-series of SMPL-X [36] 3D human meshes and no assumptions are made about the scene itself. The scene can contain any arbitrary number of objects of varying shapes and at any location. Both the 3D animation and scene are inputs to our method which outputs the location and orientation of the most natural placement of the animation in the scene. The contributions of this paper are as follows:

1. We present a deep learning-based selection method to find "keyframes," frames in the animation most important for modeling a relevant scene interaction. Our method utilizes a deep model to determine potentially important frames from the animation's estimated interactions and geometry. Inspired by techniques from active learning literature we then calculate a diversity score for each mesh in the animation. Using the output of our deep model and the diversity score, we weight meshes in the animation such that the highest weight is attributed to meshes that maximize diversity and contain interactions with the scene important for a natural placement. We call the frames or meshes in the animation with the highest weight "keyframes."

2. We present an algorithm that utilizes the keyframes alongside a 3D scene's semantic information and affordances to optimize the placement of the animation into the scene. Our algorithm searches for target animation placements, optimizing the animations' position and orientation while maximizing the match between the estimated animation interactions and the scene geometry and semantics. We call our complete method PAAK, for "Placement of Animations with Active Keyframes".

3. We qualitatively show natural and physically plausible human placement results. Through a perceptual study, we show that human raters prefer our method over PROX ground truth [12] and rate it as more realistic than an extension of POSA to the time dimension (61.5% v. 38.5%), a generative method [52] (76.9% vs 23.1%), and a purely geometric keyframe extraction method (52.3% vs 47.7%).

2. Related Work

Human Models. Most existing work utilizes body skeletons [16, 43] to model 3D humans. However, the surface of the body is important for rendering an actual human or modeling interactions with the environment or objects. Learned parametric 3D body models have addressed this need [2, 18, 29, 36, 35]. In this work, we utilize SMPL-X [36], an extension of [29] that models face and hand articulation in addition to the rest of the body.

Motion Synthesis. Motion synthesis is a longstanding problem in computer vision and computer graphics [23, 37, 49, 11, 45, 28, 56, 15, 40, 52, 51]. Much of the early work on motion synthesis focused on synthesizing intermediate states between two given frames [49, 11, 55]. However, these methods are not able to handle large translational position changes effectively. Xu et al. [56] and Holden et al. [15] utilized data-driven deep models for motion synthesis, showing better generalization than the geometric methods of the past. While many of these methods can create convincing 3D human motion, none of them address interactions between humans and scenes.

Some motion synthesis methods do consider both motion and the environment [45, 28, 5, 54]. However, these methods use a greatly simplified scenario with predefined objects and primitive motion. Our work is closest to [52] which accounts for an arbitrary 3D scene when synthesizing motion. The motions [52] and other motion synthesis methods produce falls short of the realism of animations created through motion-capture. Our method does not synthesize its own motion and instead leverages motion capture data for the most realistic animation-scene pairing possible.

Video Synthesis. Our work is also related to the synthesis of full videos containing human actions. The advent of generative adversarial networks (GANS) [7] and neural radiance fields (NeRF) [31] have contributed to a growing body of work attempting to generate videos of humans completing actions in arbitrary scenes. Niemeyer et al. [32] proposed a fully generative method which creates a full 3D

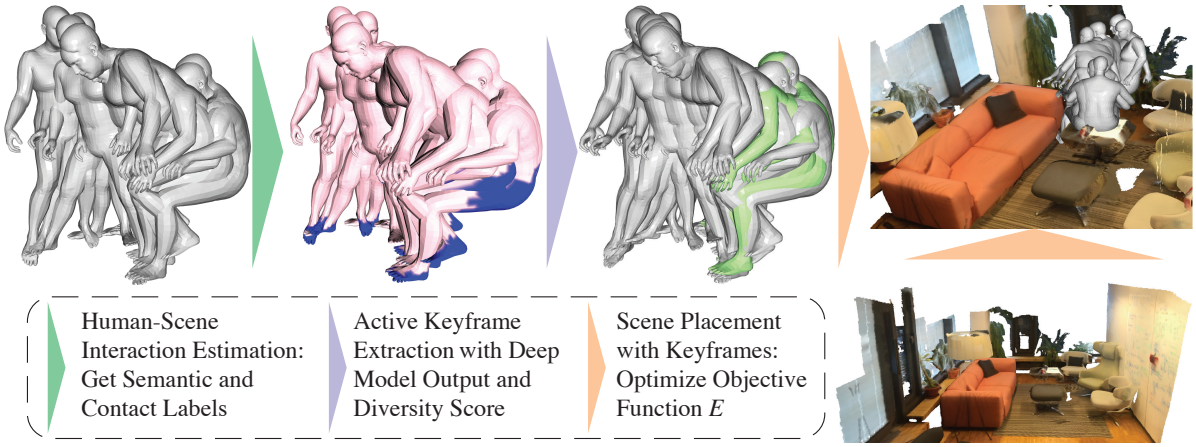

Figure 2: An overview of our PAAK method. We first estimate human-scene interactions and use those interactions to determine the keyframes in the animation. We can then utilize the keyframes alongside the 3D scene itself to place the animation convincingly into the scene.

scene with an arbitrary number of objects before rendering the scene into a 3D image. [38, 34, 58, 47] worked towards extending NeRF to articulated objects like humans.

[57, 9, 61, 24, 4, 33] extend these works towards full videos with human actions. Most closely related to our work is [33], which similarly takes a human action and attempts to place it on a synthetic, usually sparse background. Our approach addresses the shortcomings of [33], especially the lack of natural-looking interactions between the animation and background and the lack of detail in the background.

Active Learning. Active learning aims to annotate a small subset of a dataset to address a lack of sufficient labeled data. To do so, it computes the relevance of each sample based on a variety of parameters such as uncertainty/ entropy, diversity [3], and a careful trade-off of uncertainty and diversity [39]. Its applications include object detection [41], domain adaptation [46, 22] and video tracking [50]. In this paper, we aim to assign weights to frames in accordance with their relevance to potential human-scene interactions. For this, our algorithm computes the relevance of each frame by using a heuristic that is inspired by BADGE [3].

Human-Scene Interaction. The main focus of our work is human-scene interaction (HSI), or scene affordance. Early work in HSI was purely geometric with Gleicher [6] using contact constraints for motion retargeting and Kim et al. [20] automating the generation of 3D skeletons into a 3D environment. Subsequent geometric works continued to exploit the importance of contact and began to account for the forces present in the environment [19, 25, 8]. Gupta et al. [10] estimated the human poses "afforded" by the scene by predicting a 3D scene occupancy grid and computing the support and penetration of a 3D skeleton inside it. A subset of this research began to focus on dynamic interactions, or animations [1, 14].

More recently, data-driven approaches have begun to dominate [48, 17, 42, 60, 13, 26]. Jiang et al. [17] and Koppula et al. [21] learn to estimate human poses and object affordances from an RGB-D 3D scene. Wang et al. [53] learns to utilize the scene affordances to optimize pose estimation. Closest to our method, PSI [60], PLACE [59], and POSA [13] populate scenes with SMPL-X [36] human meshes. POSA [13] is unique in its human-centric approach and utilization of dense body-to-scene contact. Specifically, in POSA, Hassan et al. uses a cVAE to learn contact probabilities and a corresponding semantic label for every vertex on a SMPL-X human mesh before using this information to find the best affordance for that mesh in a given 3D scene. In our method, we leverage POSA's model and utilize the contact and semantic information it provides. The key difference between our method and POSA is our use of an animation instead of a singular human mesh, creating additional challenges addressed by our keyframe methods.

3. Placement of Human Animations into 3D Scenes

In this work, we consider the following problem statement: *Given a 3D human animation and a 3D scene, find the most natural-looking and physically plausible placement in the scene.* Specifically, our goal is to take a given animation and place it into a scene mesh such that any interactions in the animation (i.e. sitting in a chair, laying on a bed, or touching an object) match the affordances of the scene. The crux of our work is the idea that some frames in

Symbols	Definitions
V_b	A 3D human animation, consisting of a time series of 3D human meshes
v_b	A single 3D human mesh, an individual frame from V_b
f_c, f_s	Contact labels and Semantic labels, attributed to each vertex in a mesh, v_b
K_g, K_a	Geometric and Active Keyframe weights respectively
W_s, W_m, W_d	Semantic, motion, and diversity weights for an animation respectively
E	The objective function utilized when optimizing the animations placement

Table 1: List of symbols used and their definitions.

an animation are more important than others when optimizing the placement of an animation into a scene in the most natural way. Intuitively, you can imagine an animation with many static frames before a motion begins. When placing this animation into the scene treating all frames equally, the static frames will outweigh the moving ones in an optimizer as there are many more of them. In contrast, our method will find the moving frames and emphasize them to the optimizer, resulting in a more natural looking placement in the scene.

We present an overview of our method in Figure 2. A list of symbols frequently used in this work is shown in Table 1. Rarely used symbols are defined where they are used. Note that notation relating to a time-series beginning with an uppercase letter denotes the full time-series while a lowercase letter denotes an individual frame in the time-series. We begin with a set of 3D human animations and a set of 3D scene meshes. Each 3D human animation, V_b consists of a time series of human meshes, v_b, and skeletons represented by the SMPL-X[36] body model. We first estimate likely human-scene interactions given the poses of the human meshes in the animation. We then use a deep model to extract geometrically and semantically important keyframes before employing modified active learning techniques that leverage the deep model to extract a diversity score for each frame. The model output and diversity score are then combined to create "Active Keyframes," k, frames in the animation that maximize diversity *and* represent important interactions with the scene (shown as the green meshes in Figure 2). Using our keyframe estimation we weigh the frames in the animation by their importance and use an optimizer to place the 3D human animation into the scene in the most natural way. This process results in a animation-scene pairing where interactions in the animation itself are matched with the geometry and semantics of the scene.

We chose an optimization method instead of end-to-end deep learning because it will *always* find the ideal placement in the scene given the information we supply. In contrast, a deep model would learn to generalize over all animations and scenes, not necessarily finding the ideal placement for any of them. Our optimization method allows for the combination of our keyframe weights and individual meshes in the animation to prioritize interactions key to realism.

3.1. Human-Scene Interaction Estimation

To place an animation into the scene in a way that preserves any interactions present in the animation, it is essential to first determine what those interactions are. To encode the estimated relationship with a scene into the animation, we directly implement the POSA [13] model and feed in each frame of the animation individually to extract semantic and contact labels. POSA uses a conditional variational autoencoder (cVAE), f, to generate an egocentric feature map from each SMPL-X human mesh vertices in the animation. For example, when the mesh is of a person sitting in a chair, a vertex on a person's back should have a high contact probability and activate the chair semantic label, while a shin vertex would have a very low contact probability. POSA can be represented as the function f in Equation 1:

$$f : (v_b) \rightarrow [f_c, f_s] \tag{1}$$

3.2. Geometric Keyframes

We present a novel geometric algorithm for placing a 3D animation into a scene such that interactions match the affordances of the scene given a 3D human animation, its likely interactions, and the scene. Optimizing the placement of the animation into the scene while weighting all the frames in the animation equally will miss important cues only present for a few of the frames. Weighting the frames higher if there is a likely interaction makes it much harder for the placement optimization process to miss key interactions. Semantic, contact, and geometric information of a frame relative to the animation itself are strong indicators of whether a given frame is essential for matching interactions in the animation with the affordances of the scene.

Our geometric keyframe weighting formula, K_g, Equation 2, consists of a weighted combination of semantic weights and motion weights. Equation 3 and Equation 4 show how the semantic term and motion term are calculated, respectively. Semantic weights are calculated for each mesh in the animation individually, where each individual weight sums the number of vertices in the mesh that are activated by the dominant semantic class in the animation, defined by the mode of the animation's semantic labels[1], $Mo(F_s)$.

[1] excluding the floor class as it would almost always dominate

The motion weight uses the skeleton from the SMPL-X model and models lateral motion by taking the Euclidean distance from the pelvis at a frame, p_i, to the pelvis in the next frame, p_{i+1}. This provides a higher weight to frames with a quick motion than those where the human is static.

$$K_g = \lambda_s * \frac{W_s}{max(W_s)} + \lambda_m * \frac{W_m}{max(W_m)} \tag{2}$$

$$w_s = \sum_{i=0}^{|v_b|} [f_{s,i} = Mo(F_s)] \tag{3}$$

$$W_m = ||p_i - p_{i+1}|| \tag{4}$$

The λ_s and λ_m weighting factors are set through experimentation to achieve the best balance of semantic and motion weighting in the total keyframe weight.

Our Geometric Keyframes method can be thought of as prioritizing the meshes in the animation where an interaction with the scene is taking place or there is a large motion happening. For example, in an animation with a sitting action, the few frames where the human is sitting will be prioritized such that when placed into the scene, the human will be in a seat when sitting. Additionally, the motion weighting helps mitigate situations where large motions may cause the human to collide with obstacles in the environment when many static frames may otherwise dictate the placement. In practice, this leads to more natural-looking placements throughout the scene than with the semantic term alone.

3.3. Active Keyframes

While our Geometric Keyframes method is effective at picking out important semantic cues and quick motions, we want to further improve it by increasing the diversity of the highly weighted frames and finding all important interactions. The increase in diversity will find frames ignored by our geometric formulation that may still be of relative importance when placing the animations into the scene. Our approach is motivated by active learning methods because while these try to find new pieces of an unlabeled dataset that would maximize diversity if annotated, we are looking for frames in an animation that would maximize the diversity of the high-weighted frames.

In our active keyframe method, K_a, Equation 5, we begin with a deep fully-connected model. Our model can be thought of as a function, g, mapping from the animation vertices, contact labels, and semantic labels, to the geometric keyframe weights from Equation 2. The architecture of our model is shown in Figure 3. The primary benefit of this architecture is the ability to both connect through each mesh individually and across the animation. Another benefit of our model over our geometric equations is its generalization to multiple prominent interactions without forcing a set

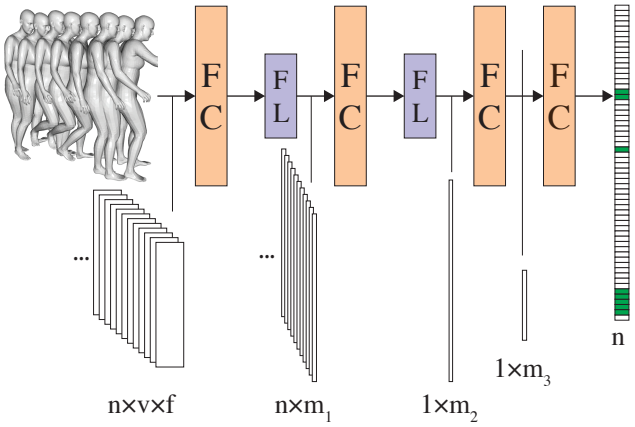

Figure 3: Network Architecture. The input animation is n meshes with v vertices and f features each. We utilize four fully connected (FC) layers with the first layer operating across each vertex while the second layer operates across all the vertices in the mesh. The last two layers operate across the entire animation. The model outputs an array of size n, with each index the weight of the corresponding mesh in the animation. The m values are intermediate representations and the FL layers correspond to a flattening of the input along the last two dimensions.

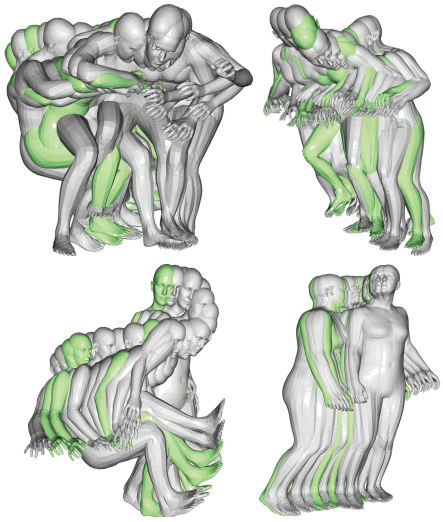

Figure 4: Random samples from our active keyframe framework. The green frames are those with the highest weight in K_a. Note that the frames where an important interaction occurs are preferred.

number of interactions to track. Utilizing a deep model also allows us to employ utilize techniques from active learning literature to compute a diversity score. Specifically, we use BADGE [3] which uses the gradient of the model itself to

determine a subset of the animation meshes that will maximize diversity.

$$K_a = \lambda_g * \hat{K_g} + \lambda_b * W_d \tag{5}$$

$$g : (V_b, F_c, F_s) \rightarrow \hat{K_g} \tag{6}$$

To calculate the diversity score, BADGE divides samples into batches using kNN clustering on the gradient score. Samples are selected using both the magnitude of the gradient as well as the distance from previously selected samples. Since our goal is to obtain gradient-based diversity scores for each mesh rather than selecting samples for annotation, we run BADGE over all meshes/ frames of a video by setting the number of annotation frames to be equal to the number of frames. BADGE returns the gradient diversity score w_d for each mesh in the animation.

3.4. Scene Placement with Keyframes

Now that we have a keyframe weighting for the animation, we place the animation into the 3D scene such that it makes sense in the context of the scene, completing our PAAK method. Our approach takes the 3D scene mesh, the animation, and our keyframe weights and uses them to optimize an objective function, E. Optimizing E finds the body translation in the scene, τ, and the global body orientation, θ, that minimize the sum of an affordance loss, $\mathcal{L}_{afford}$, and a penetration loss, $\mathcal{L}_{pen}$, calculated for each frame weighted by our keyframe weights, k_g or k_a.

$$E(\tau, \theta) = \sum_{i=0}^{|k|} k_i * [\mathcal{L}_{afford,i} + \mathcal{L}_{pen,i}] \tag{7}$$

Both $\mathcal{L}_{afford}$ and $\mathcal{L}_{pen}$ are adapted from [13]. $\mathcal{L}_{afford}$ is minimized when the distance to the scene is small for vertices with a high probability of contact, f_c, and when the semantic label, f_s, matches the semantics of the object it is touching in the scene. $\mathcal{L}_{pen}$ heavily punishes a placement that results in a mesh penetrating the scene. By weighting these values with our keyframe weights, we make sure that the minimum value of E is found by maximizing correct affordances and minimizing penetration at the Keyframes. Without our keyframe weighting, the objective function becomes saturated across the meshes in the animation, not placing the animation such that it matches the scene context adequately. Optimizing E without our keyframe weighting is a baseline we use in our experiments.

4. Experiments

4.1. Dataset and Baselines

For all of our experiments, we utilized the PROX dataset [12], which contains animations of humans moving through 12 different scenes. The PROX ground-truth (GT) SMPL-X [36] parameters are generated by a fitting algorithm, introducing some noise into the animations. From PROX, we sample 10k animations from 8 of the 12 available scenes. For evaluation, we randomly sample animations from our PROX subset and place them into one of the four remaining scenes. These results are then shown to human raters in comparison with PROX GT or another method, and these raters then pick the more realistic of two videos. Both videos are from the same scene.

We compare our method with POSA and a motion synthesis method as baselines. Additionally, we ablate our Active Keyframes method with our Geometric Keyframes method. The baselines are outlined in further detail below.

POSA-T. Hassan et al. [13] propose a method that places a single human body into a scene given its affordances. As this approach does not consider a full animation we sum the loss over all the meshes in the animation and provide that to the optimizer. We call this baseline POSA-T for our addition of the time dimension.

Motion Synthesis. Wang et al. [52] propose a motion synthesis method that accounts for the affordances of the scene in its motion creation. We did not alter their approach and used it as designed with the same four scenes it set aside for testing. We call this baseline Motion Synthesis.

Geometric Keyframes. Our Geometric Keyframes method is an ablative baseline that does not include our model nor our active keyframe implementation. It extracts the keyframe weights purely from semantic and geometric information, as described in Equations 2-4. We call this baseline Geometric Keyframes.

4.2. Evaluation

A qualitative comparison of PAAK alongside the POSA-T and Geometric Keyframes baselines can be found in Figure 5. Qualitatively, we found that PAAK calculated improved placements over the baselines, with POSA-T especially prone to producing a placement that was not valid in the scene context, like having the human sit in mid-air.

Comparison to PROX ground truth. Following the protocols of Hassan et al. [13] and Zhang et al. [59], we compare our results to randomly selected examples from PROX ground truth. We utilize 4 real 3D scenes from the PROX test dataset, namely MPH16, MPH1Library, N0SittingBooth, and N3OpenArea. We then take 80 2-second animations from the PROX training dataset (not from the 4 scenes listed) for placement into the test scenes with each of our methods. The placement process for every method begins with a grid of potential placement locations and orientations on the scene, with each placement testing rotations every 30 degrees. We then filter these initial placements to 10 promising prospects based on E and continue to optimize them until each reaches its optimum translation and rotation. For each of these 10 prospects, the location

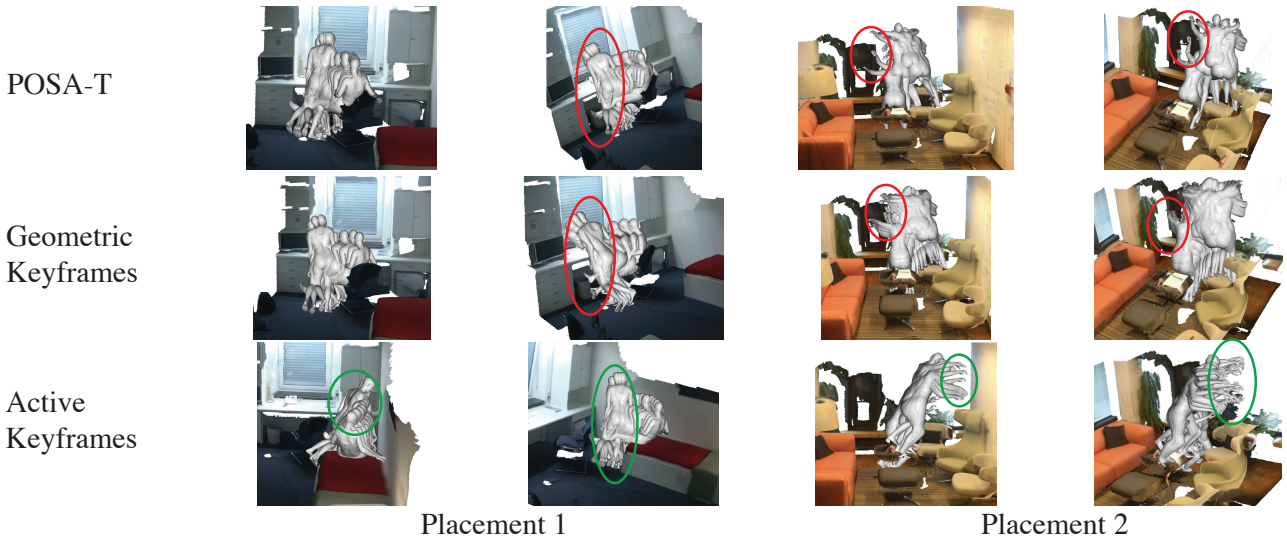

Figure 5: Comparisons on placing the same animation into the same scene across the POSA-T, Geometric Keyframes, and Active Keyframes Methods. Note that two angles of each placement are provided. For Placement 1, only the Active Keyframes method placed the animation on the bed (green circle), allowing for a more reclined seating position that results in standing more upright at the end of the animation. For Placement 2, a jumping action is taking place. Only the Active Keyframes method was able to position the animation such that the hands were not in collision with the back wall.

	Placement ↑	PROX GT ↓
POSA-T	44.6%	55.4%
Geom. Keyframes	52.3%	47.7%
PAAK	**64.6%**	35.4%

Table 2: Comparison to PROX [12] ground truth. Subjects are shown pairs of an action placed into a 3D scene and PROX ground truth (GT) and must choose the most realistic one. A higher percentage indicates the scene subjects deemed more realistic.

	Baseline ↓	PAAK ↑
POSA-T	38.5%	**61.5%**
Motion Synthesis	23.1%	**76.9%**
Geom. Keyframes	47.7%	**52.3%**

Table 3: PAAK compared to POSA-T, Motion Synthesis [52], and our Geometric Keyframes ablation. The comparison procedure is the same as for Table 2.

with the lowest total loss is selected as the final placement location. We render each animation scene pair into videos from four different angles so subjects can get a sense of the relationships between the animation and the scene. Using a web-based user study with 18 subjects, each being shown a subset of the image scene pairs we produced, we collect 540 unique ratings. The results are shown in Table 2. The Geometric Keyframes ablation is almost indistinguishable from the PROX GT, while POSA-T falls short. However, the human raters preferred PAAK over PROX GT. We believe this is due to scene penetrations in the PROX dataset when humans are sitting. This is caused by deformations in real life not captured in the scene.

Comparison to baselines. To compare the baselines directly, we follow the same protocol as above, but replace the PROX ground truth with a competing method. In addition to comparing against POSA-T and the Geometric Keyframes ablative baseline, we also directly compare against Motion Synthesis. The results are shown in Table 3. Again, we find that the human raters preferred PAAK, finding it more realistic than the other methods. PAAK significantly outperforms the Motion Synthesis baseline. This makes sense as the Motion Synthesis method is solving a slightly different task, generating natural-looking motion, while our method utilizes real-world data. This shows the need for a method like ours that uses motion captured animations.

PAAK placements are also perceived by the human raters as more natural-looking than the POSA-T method. This makes sense as the POSA-T method will weigh all the individual meshes equally when placing the animation into the scene, making it more difficult for the optimizer to find a placement that maximizes the realism of the motion, like landing on a surface when sitting or touching an object when reaching. PAAK also outperforms our Geomet-

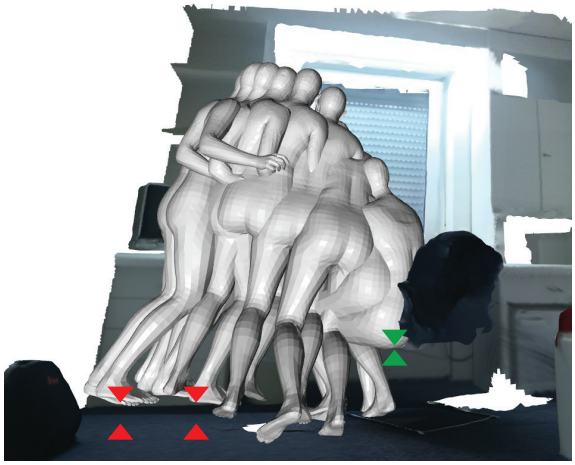

Figure 6: An example of a Keyframe-based placement leaving feet off the floor to place the buttocks correctly on the seat.

ric Keyframes ablative baseline. We believe this is due to the intelligent frame selection present in the method, which selects a more diverse set of keyframes to supplement the frames selected by the model for their semantic and geometric information. This added diversity can help to pick up on additional semantic cues not utilized in the Geometric Keyframes method.

Physical plausibility. Following the procedures utilized by [13, 59, 60], we take 100 animations of 60 frames each and place them in the 4 test scenes of PROX. Given the body meshes, the scene mesh, and a scene signed distance field (SDF), we compute a non-collision score and contact score as defined by [60] with the results in Table 4. The non-collision score is calculated for each mesh in an animation as the ratio of the body vertices with positive SDF values divided by the total number of SMPL-X vertices. A high non-collision score denotes that the meshes in each animation do not penetrate the scene. PAAK is comparable to the POSA-T and Geometric Keyframes baselines in the non-collision score.

The contact score is calculated for each mesh individually and is 1 if at least one vertex of the mesh is in direct contact with the scene. PAAK is comparable to the POSA-T and geometric keyframes baselines in the contact score. The small difference in performance is likely due to slight mismatches between the animation and the scene. For example, in Figure 6 PAAK ensures contact with a seat when sitting, however, an imperfect match in seat size between the animation and the scene results in feet close to but not completely in contact with the ground when the person stands.

	Non-Collision ↑	Contact ↑
POSA-T	0.98	**0.83**
Geometric Keyframes	0.98	0.81
PAAK	**0.99**	0.81

Table 4: Evaluation of the physical plausibility metrics. A higher score is better for both.

Comparatively, a placement that makes sure the feet are in contact with the ground for every mesh in the animation would get a perfect score.

5. Conclusions, Limitations, and Future Work

In this paper, we propose PAAK, a novel method for placing a 3D human animation into a 3D scene with accurately modeled human-scene interactions. We introduce "keyframes," the frames in an animation most important for the interactions with the scene, and use these keyframes to place the animation into a scene. Human raters preferred PAAK animation placements over real-world PROX [12] ground truth data, and over existing methods.

Limitations and Future Work. Note that PAAK does not *always* create natural placements. There are still cases where people sit in strange places or walk where they typically would not. Our optimization method can miss good placements as it relies on a grid of initial placements before optimizing the best ones. For example, initial placements with heavily penalized penetrations could become the best available with further optimization. We limited this for time but more compute could enable improved placements. The ability to rate the quality of a placement would be valuable for end users. For example, if an animator has a bank of 100 animations, how can they pick the top 5 to populate the scene? PAAK can be extended to model human-human interactions when placing multiple animations into a scene by adding a term to the semantic keyframe extraction. Finally, altering the animation itself could be a valuable extension that allows for more natural-looking interactions with the scene and better performance in the physical plausibility metrics.

Acknowledgements. This material is based upon work supported by the National Science Foundation Graduate Research Fellowship Program under Grant No. DGE 1840340. Any opinions, findings, and conclusions or recommendations expressed in this material are those of the author(s) and do not necessarily reflect the views of the National Science Foundation. This research was supported by Army Cooperative Agreement W911NF2120076.

References

[1] Rami Ali Al-Asqhar, Taku Komura, and Myung Geol Choi. Relationship descriptors for interactive motion adaptation. In *Proceedings of the 12th ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '13, pages 45–53, New York, NY, USA, July 2013. Association for Computing Machinery.

[2] Dragomir Anguelov, Praveen Srinivasan, Daphne Koller, Sebastian Thrun, Jim Rodgers, and James Davis. SCAPE: Shape completion and animation of people. *ACM Transactions on Graphics*, 24(3):408–416, July 2005.

[3] Jordan T Ash, Chicheng Zhang, and Akshay Krishnamurthy. DEEP BATCH ACTIVE LEARNING BY DIVERSE, UNCERTAIN GRADIENT LOWER BOUNDS. *ICLR*, page 26, 2020.

[4] Eric R. Chan, Connor Z. Lin, Matthew A. Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas Guibas, Jonathan Tremblay, Sameh Khamis, Tero Karras, and Gordon Wetzstein. Efficient geometry-aware 3D generative adversarial networks. In *arXiv*, 2021.

[5] Alexander Clegg, Wenhao Yu, Jie Tan, C. Karen Liu, and Greg Turk. Learning to dress: Synthesizing human dressing motion via deep reinforcement learning. *ACM Transactions on Graphics*, 37(6):179:1–179:10, Dec. 2018.

[6] Michael Gleicher. Retargetting motion to new characters. In *Proceedings of the 25th Annual Conference on Computer Graphics and Interactive Techniques - SIGGRAPH '98*, pages 33–42, Not Known, 1998. ACM Press.

[7] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative Adversarial Nets. In *Advances in Neural Information Processing Systems*, volume 27. Curran Associates, Inc., 2014.

[8] Helmut Grabner, Juergen Gall, and Luc Van Gool. What makes a chair a chair? In *CVPR 2011*, pages 1529–1536, June 2011.

[9] Jiatao Gu, Lingjie Liu, Peng Wang, and Christian Theobalt. STYLENERF: A STYLE-BASED 3D-AWARE GENERATOR FOR HIGH-RESOLUTION IMAGE SYNTHESIS. *ICLR*, page 25, 2022.

[10] Abhinav Gupta, Scott Satkin, Alexei A. Efros, and Martial Hebert. From 3D scene geometry to human workspace. In *CVPR 2011*, pages 1961–1968, Colorado Springs, CO, USA, June 2011. IEEE.

[11] Félix G. Harvey, Mike Yurick, Derek Nowrouzezahrai, and Christopher Pal. Robust motion in-betweening. *ACM Transactions on Graphics*, 39(4):60:60:1–60:60:12, July 2020.

[12] Mohamed Hassan, Vasileios Choutas, Dimitrios Tzionas, and Michael Black. Resolving 3D Human Pose Ambiguities With 3D Scene Constraints. In *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 2282–2292, Seoul, Korea (South), Oct. 2019. IEEE.

[13] Mohamed Hassan, Partha Ghosh, Joachim Tesch, Dimitrios Tzionas, and Michael J. Black. Populating 3D Scenes by Learning Human-Scene Interaction. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 14703–14713, Nashville, TN, USA, June 2021. IEEE.

[14] Edmond S. L. Ho, Taku Komura, and Chiew-Lan Tai. Spatial relationship preserving character motion adaptation. *ACM Transactions on Graphics*, 29(4):33:1–33:8, July 2010.

[15] Daniel Holden, Jun Saito, and Taku Komura. A deep learning framework for character motion synthesis and editing. *ACM Transactions on Graphics*, 35(4):138:1–138:11, July 2016.

[16] Catalin Ionescu, Dragos Papava, Vlad Olaru, and Cristian Sminchisescu. Human3.6M: Large Scale Datasets and Predictive Methods for 3D Human Sensing in Natural Environments. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(7):1325–1339, July 2014.

[17] Yun Jiang, Hema Koppula, and Ashutosh Saxena. Hallucinated Humans as the Hidden Context for Labeling 3D Scenes. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2993–3000, 2013.

[18] Hanbyul Joo, Tomas Simon, and Yaser Sheikh. Total Capture: A 3D Deformation Model for Tracking Faces, Hands, and Bodies. In *2018 IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 8320–8329, Salt Lake City, UT, USA, June 2018. IEEE.

[19] Changgu Kang and Sung-Hee Lee. Environment-Adaptive Contact Poses for Virtual Characters. *Computer Graphics Forum*, 33(7):1–10, 2014.

[20] Vladimir G. Kim, Siddhartha Chaudhuri, Leonidas Guibas, and Thomas Funkhouser. Shape2Pose: Human-centric shape analysis. *ACM Transactions on Graphics*, 33(4):120:1–120:12, July 2014.

[21] Hema Swetha Koppula, Rudhir Gupta, and Ashutosh Saxena. Learning human activities and object affordances from RGB-D videos. *International Journal of Robotics Research*, 32(8):951–970, July 2013.

[22] Divya Kothandaraman, Sumit Shekhar, Abhilasha Sancheti, Manoj Ghuhan, Tripti Shukla, and Dinesh Manocha. Distilladapt: Source-free active visual domain adaptation. *arXiv preprint arXiv:2205.12840*, 2022.

[23] Lucas Kovar and Michael Gleicher. Flexible automatic motion blending with registration curves. In *Proceedings of the 2003 ACM SIGGRAPH/Eurographics Symposium on Computer Animation*, SCA '03, pages 214–224, Goslar, DEU, July 2003. Eurographics Association.

[24] Youngjoong Kwon, Dahun Kim, Duygu Ceylan, and Henry Fuchs. Neural Human Performer: Learning Generalizable Radiance Fields for Human Performance Rendering. In *Advances in Neural Information Processing Systems*, volume 34, pages 24741–24752. Curran Associates, Inc., 2021.

[25] Kurt Leimer, Andreas Winkler, Stefan Ohrhallinger, and Przemyslaw Musialski. Pose to Seat: Automated design of body-supporting surfaces. *Computer Aided Geometric Design*, 79:101855, May 2020.

[26] 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 *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 12360–12368, Long Beach, CA, USA, June 2019. IEEE.

[27] Jenny Lin, Xingwen Guo, Jingyu Shao, Chenfanfu Jiang, Yixin Zhu, and Song-Chun Zhu. A virtual reality platform for dynamic human-scene interaction. In *SIGGRAPH ASIA 2016 Virtual Reality Meets Physical Reality: Modelling and Simulating Virtual Humans and Environments*, SA '16, pages 1–4, New York, NY, USA, Nov. 2016. Association for Computing Machinery.

[28] Hung Yu Ling, Fabio Zinno, George Cheng, and Michiel Van De Panne. Character controllers using motion VAEs. *ACM Transactions on Graphics*, 39(4):40:40:1–40:40:12, July 2020.

[29] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Transactions on Graphics*, 34(6):1–16, Nov. 2015.

[30] Jennifer Martin. How to Make Immersive Game Design — University of Silicon Valley. https://usv.edu/blog/how-to-make-immersive-game-design/, 2020.

[31] Ben Mildenhall, Pratul P Srinivasan, Matthew Tancik, Jonathan T Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. *ECCV*, page 17, 2020.

[32] Michael Niemeyer and Andreas Geiger. GIRAFFE: Representing Scenes as Compositional Generative Neural Feature Fields. In *2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 11448–11459, Nashville, TN, USA, June 2021. IEEE.

[33] Atsuhiro Noguchi, Xiao Sun, Stephen Lin, and Tatsuya Harada. Unsupervised Learning of Efficient Geometry-Aware Neural Articulated Representations, Apr. 2022.

[34] Atsuhiro Noguchi, Sun Xiao, Stephen Lin, and Tatsuya Harada. Neural Articulated Radiance Field. In *Proceedings of the IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 5762–5772, Oct. 2021.

[35] Ahmed A. A. Osman, Timo Bolkart, and Michael J. Black. STAR: Sparse Trained Articulated Human Body Regressor. In Andrea Vedaldi, Horst Bischof, Thomas Brox, and Jan-Michael Frahm, editors, *Computer Vision – ECCV 2020*, volume 12351, pages 598–613. Springer International Publishing, Cham, 2020.

[36] Georgios Pavlakos, Vasileios Choutas, Nima Ghorbani, Timo Bolkart, Ahmed A. Osman, Dimitrios Tzionas, and Michael J. Black. Expressive Body Capture: 3D Hands, Face, and Body From a Single Image. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 10967–10977, Long Beach, CA, USA, June 2019. IEEE.

[37] Vladimir Pavlovic, James M Rehg, and John MacCormick. Learning Switching Linear Models of Human Motion. In *Advances in Neural Information Processing Systems*, volume 13. MIT Press, 2000.

[38] Sida Peng, Junting Dong, Qianqian Wang, Shangzhan Zhang, Qing Shuai, Xiaowei Zhou, and Hujun Bao. Animatable Neural Radiance Fields for Modeling Dynamic Human Bodies. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 14294–14303, Montreal, QC, Canada, Oct. 2021. IEEE.

[39] Viraj Prabhu, Arjun Chandrasekaran, Kate Saenko, and Judy Hoffman. Active domain adaptation via clustering uncertainty-weighted embeddings. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 8505–8514, 2021.

[40] Davis Rempe, Tolga Birdal, Aaron Hertzmann, Jimei Yang, Srinath Sridhar, and Leonidas J. Guibas. HuMoR: 3D Human Motion Model for Robust Pose Estimation. In *2021 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 11468–11479, Montreal, QC, Canada, Oct. 2021. IEEE.

[41] Soumya Roy, Asim Unmesh, and Vinay P Namboodiri. Deep active learning for object detection. In *BMVC*, page 91, 2018.

[42] Manolis Savva, Angel X. Chang, Pat Hanrahan, Matthew Fisher, and Matthias Nießner. SceneGrok: Inferring action maps in 3D environments. *ACM Transactions on Graphics*, 33(6):212:1–212:10, Nov. 2014.

[43] Leonid Sigal, Alexandru O. Balan, and Michael J. Black. HumanEva: Synchronized Video and Motion Capture Dataset and Baseline Algorithm for Evaluation of Articulated Human Motion. *International Journal of Computer Vision*, 87(1-2):4–27, Mar. 2010.

[44] Kihyuk Sohn, Honglak Lee, and Xinchen Yan. Learning Structured Output Representation using Deep Conditional Generative Models. In *Advances in Neural Information Processing Systems*, volume 28. Curran Associates, Inc., 2015.

[45] Sebastian Starke, He Zhang, Taku Komura, and Jun Saito. Neural state machine for character-scene interactions. *ACM Transactions on Graphics*, 38(6):209:1–209:14, Nov. 2019.

[46] Jong-Chyi Su, Yi-Hsuan Tsai, Kihyuk Sohn, Buyu Liu, Subhransu Maji, and Manmohan Chandraker. Active adversarial domain adaptation. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 739–748, 2020.

[47] Shih-Yang Su, Frank Yu, Michael Zollhöfer, and Helge Rhodin. A-NeRF: Articulated neural radiance fields for learning human shape, appearance, and pose. In *Advances in Neural Information Processing Systems*, 2021.

[48] Fuwen Tan, Crispin Bernier, Benjamin Cohen, Vicente Ordonez, and Connelly Barnes. Where and who? Automatic semantic-aware person composition. In *IEEE Winter Conf. on Applications of Computer Vision (WACV)*, 2018.

[49] Raquel Urtasun, David J. Fleet, Andreas Geiger, Jovan Popović, Trevor J. Darrell, and Neil D. Lawrence. Topologically-constrained latent variable models. In *Proceedings of the 25th International Conference on Machine Learning - ICML '08*, pages 1080–1087, Helsinki, Finland, 2008. ACM Press.

[50] Carl Vondrick and Deva Ramanan. Video annotation and tracking with active learning. *Advances in Neural Information Processing Systems*, 24, 2011.

[51] Jingbo Wang, Yu Rong, Jingyuan Liu, Sijie Yan, Dahua Lin, and Bo Dai. Towards Diverse and Natural Scene-Aware 3D Human Motion Synthesis. *CVPR*, page 10, 2022.

[52] Jiashun Wang, Huazhe Xu, Jingwei Xu, Sifei Liu, and Xiaolong Wang. Synthesizing Long-Term 3D Human Motion and Interaction in 3D Scenes. In *2021 IEEE/CVF Confer-*

ence on Computer Vision and Pattern Recognition (CVPR), pages 9396–9406, Nashville, TN, USA, June 2021. IEEE.

[53] Zhe Wang, Liyan Chen, Shaurya Rathore, Daeyun Shin, and Charless Fowlkes. Geometric Pose Affordance: 3D Human Pose with Scene Constraints, Dec. 2021.

[54] Ziyu Wang, Alexander Novikov, Konrad Żołna, Jost Tobias Springenberg, Scott Reed, Bobak Shahriari, Noah Siegel, Josh Merel, Caglar Gulcehre, Nicolas Heess, and Nando de Freitas. Critic regularized regression. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS'20, pages 7768–7778, Red Hook, NY, USA, Dec. 2020. Curran Associates Inc.

[55] Guiyu Xia, Huaijiang Sun, Qingshan Liu, and Renlong Hang. Learning-Based Sphere Nonlinear Interpolation for Motion Synthesis. *IEEE Transactions on Industrial Informatics*, 2019.

[56] Jingwei Xu, Huazhe Xu, Bingbing Ni, Xiaokang Yang, Xiaolong Wang, and Trevor Darrell. Hierarchical Style-based Networks for Motion Synthesis. *ECCV*, page 16, 2020.

[57] Gengshan Yang, Deqing Sun, Varun Jampani, Daniel Vlasic, Forrester Cole, Ce Liu, and Deva Ramanan. ViSER: Video-Specific Surface Embeddings for Articulated 3D Shape Reconstruction. In *Advances in Neural Information Processing Systems*, volume 34, pages 19326–19338. Curran Associates, Inc., 2021.

[58] Gengshan Yang, Minh Vo, Natalia Neverova, Deva Ramanan, Andrea Vedaldi, and Hanbyul Joo. BANMo: Building Animatable 3D Neural Models From Many Casual Videos. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 2863–2873, 2022.

[59] Siwei Zhang, Yan Zhang, Qianli Ma, Michael J. Black, and Siyu Tang. PLACE: Proximity Learning of Articulation and Contact in 3D Environments. In *2020 International Conference on 3D Vision (3DV)*, pages 642–651, Fukuoka, Japan, Nov. 2020. IEEE.

[60] Yan Zhang, Mohamed Hassan, Heiko Neumann, Michael J. Black, and Siyu Tang. Generating 3D People in Scenes Without People. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 6193–6203, Seattle, WA, USA, June 2020. IEEE.

[61] Fuqiang Zhao, Wei Yang, Jiakai Zhang, Pei Lin, Yingliang Zhang, Jingyi Yu, and Lan Xu. HumanNeRF: Efficiently Generated Human Radiance Field From Sparse Inputs. *CVPR*, page 11, 2022.