SceneMI: Motion In-betweening for Modeling Human-Scene Interactions

Supplementary Material

Inwoo Hwang[1*] Bing Zhou[2†] Young Min Kim[1] Jian Wang[2] Chuan Guo[2†]

[1]Dept. of Electrical and Computer Engineering, Seoul National University, [2]Snap Inc.

In the supplementary materials, we elaborate the implementation details for our SceneMI (Sec. 1), additional analysis with experiments on varying keyframe selection strategy, runtime analysis, and an ablation study on hyperparameter settings with discussing limitations (Sec. 2). Furthermore, we introduce a detailed *Video-based Human-Scene Interaction Reconstruction* pipeline (Sec. 3), where SceneMI plays a crucial role in enhancing realism and physical plausibility in HSI reconstruction. For additional qualitative results, please refer to the supplementary video on our project page.

1. Further Details

1.1. Implementation Details

We implemented our model using a DDPM based diffusion framework [5], leveraging the U-Net architecture proposed by [9] with the AdamW optimizer [12] with a learning rate of $1e{-}4$ and a weight decay of $1e{-}2$. For classifier-free guidance at inference, we set the guidance weight $\mathbf{w} = 2.5$. More hyperparameters of the architecture and diffusion process are organized in Table 1.

Hyperparameter	Value
Batch size	256
Learning rate	1e-4
Optimizer	Adam W
Weight decay	1e-2
Channels dim	256
Channel multipliers	$[2, 2, 2, 2]$
Variance scheduler	Cosine [14]
Diffusion steps	1000
Diffusion variance	$\tilde{\beta} = \frac{1-\alpha_{t-1}}{1-\alpha_t}\beta_t$
EMA weight (β)	0.9999
Guidance weight ($\mathbf{w}$)	2.5

Table 1. Hyperparameters of the Model

1.2. Baseline Details

We compare our approach against a diverse range of state-of-the-art motion synthesis methods, including scene-agnostic motion generation (MDM [17] and StableMoFusion [7]), motion in-betweening (OmniControl [19] and CondMDI [2]), and scene-aware motion synthesis (SceneDiffuser [6] and Wang et al. [18]). To ensure a fair comparison of scene-aware motion in-betweening tasks, we adapt their original models accordingly.

For scene agnostic works (MDM [17], StableMoFusion [7], OmniControl [19], and CondMDI [2]), we adapt them by replacing their text encoders with a Vision Transformer (ViT)-based global scene encoder to incorporate scene conditions. For diffusion-based motion synthesis methods (MDM [17], StableMoFusion [7], and SceneDiffuser [6]), we modify their inference process to support motion in-betweening by imputing joint positions at every diffusion step. Additionally, we adapt their motion representations to incorporate a global root representation, enabling keyframe-based in-betweening via imputation sampling. Across all baselines, we use only static keyframe poses—such as joint position information—to generate intermediate motions.

2. Additional Analysis

2.1. Robustness to Varying Keyframe Selection Strategy

Our motion in-betweening module experiences random keyframes with mask $\mathbf{m}$ during training, it maintains strong performance with arbitrary keyframes $\mathbf{m}^*$ at the inference. We show that our method consistently achieves robust results with keyframes chosen at arbitrary indices, even in noisy conditions. Table 2 demonstrates the robustness of our method across different keyframe selection strategies, showing its ability to handle noise effectively.

2.2. Time Cost

We report the inference time comparison with baselines in Table 3 for obtaining SMPL parameters. For realistic character animation, acquiring actual motion parameters is essential. Our method directly predicts these parameters,

Keyframe Selection	FID ↓	Jerk (m/s^3) ↓	MJPE All (m) ↓
Uniform $(r = 1)$	0.122	0.197	0.0117
Uniform $(r = 3)$	0.118	0.198	0.0129
Uniform $(r = 15)$	0.125	0.196	0.0153
Uniform $(r = 60)$	0.123	0.198	0.0233
Random $(p = 0.2)$	0.124	0.199	0.0138
Random $(p = 0.5)$	0.123	0.199	0.0124

Table 2. Quantitative evaluation of diverse keyframe selection strategies on noisy TRUMANS test set with a fixed noise level $l = 1$. We select keyframes using different strategies, such as at a uniform interval r or with a random probability p, including start and end frames. Our method shows robustness performance from highly sparse to dense keyframes, regardless of keyframe density or selection.

whereas baselines require a post-processing with an additional optimization-based fitting process from predicted joint positions. This offers a faster pipeline for obtaining actual motion compared to baselines.

Method	MDM [17]	OmniControl [19]	CondMDI [2]	Ours
Time (s)	119.4 ± 2.1	283.7 ± 3.8	162.4 ± 3.5	39.6 ± 0.8

Table 3. Time required to obtain actual parameters for motion.

2.3. Ablation on Hyper Parameters

We evaluate different configurations of global scene dimensions, the number of BPS points, and body shape conditioning within a sparse keyframe interval setup ($r = 60$) to validate our hyperparameter choices in Table 4.

We design body shape encoding, $\mathbf{b}$, that includes key joint-to-joint distances from T-pose: $[\texttt{root}, \texttt{head}]$, $[\texttt{left\_shoulder}, \texttt{right\_shoulder}]$, $[\texttt{shoulder}, \texttt{wrist}]$, $[\texttt{left\_pelvis}, \texttt{right\_pelvis}]$, and $[\texttt{pelvis}, \texttt{feet}]$. Two thickness values: distances between the frontmost and rearmost vertices of the chest region and the hip region. These measurements provide a body shape abstraction $\mathbf{b}$ as a compact shape representation in a continuous domain. Furthermore, our main experiments are conducted on diverse body shapes, including five samples from the TRUMANS dataset and real-world shapes from GIMO and Video2Animation. Although the design of the body shape encoding is not our primary contribution, it significantly enhances in-betweening accuracy and reduces penetration artifacts.

Configurations	FID ↓	Jerk (m/s^3) ↓	MJPE All (m) ↓	Collision Frame Ratio ↓	Pene. Max (m) ↓
Scene 96x48x96	0.130	0.201	0.027	0.117	0.046
BPS 256	0.124	0.196	0.025	0.114	0.045
w/o Body Shape	**0.122**	**0.193**	0.038	0.121	0.047
Ours	0.123	0.194	**0.023**	**0.113**	**0.043**

Table 4. Ablation study on our hyperparmeters setting.

3. Video-based Human-Scene Interaction Reconstruction

In this section, we present a Human-Scene Interaction Reconstruction pipeline, where our SceneMI module plays a core component. The goal is to reconstruct realistic, physically plausible human animations and scene geometry from monocular RGB video sequences that capture both scene and human movements.

The pipeline comprises two primary stages: the *initial stage* and the *refinement stage*. In the *initial stage*, we extract a rough estimate of both human motion and scene geometry in a metric scale. In the *refinement stage*, we enhance the physical plausibility and naturalness of the motion using the reconstructed scene geometry and our SceneMI module. The following sections detail the challenges and methodologies for each stage.

3.1. Initial Stage

Our framework takes as input an RGB video sequence of M frames with 30 FPS, denoted as $\{I_i\}_{i=1}^{M}$.

Camera Parameter Estimation From the first frame of the video sequence, we estimate intrinsic camera parameters using [8]. These parameters are crucial for positioning 3D human meshes or back-projecting depth estimation results in subsequent steps.

Human Mesh Recovery (HMR) We utilize 4D Humans [3] to obtain human mesh parameters for each frame. The obtained parameters are used to construct SMPL model-based human meshes, denoted as $\{X_i\}_{i=1}^{M}$. These meshes are then placed in 3D space using the previously estimated camera parameters and root translations. Since the SMPL model is defined in metric scale [11], this process provides an initial metric-scale geometry reference.

Metric-Scale Depth Estimation with HMR To recover the complete 3D scene geometry, we employ a pre-trained depth estimation network [21] to produce initial depth maps $D_{\text{init},i}$ for each frame. These depth maps, while precisely capturing relative depth relationships, lack accurate metric-scale representation. To resolve this, we estimate a global scale s and offset factor o that transform the $D_{\text{init},i}$ into metric-scale:

$$D_i = s \cdot D_{\text{init},i} + o, \quad \forall i = 1, 2, \ldots, M$$

To determine the optimal transformation parameters s and o, we leverage the metric-scale human meshes (X_i) obtained in the previous stage as geometric references. For each frame i, we sample the visible vertices from the human mesh in camera space, denoted as $V(X_i)$. We also backproject the

transformed depth map D_i into 3D space, selecting only the region corresponding to human segmentation in image I_i, to obtain point clouds denoted as $P_X(D_i)$. The alignment between these point sets is achieved by minimizing the chamfer distance between two pointsets:

$$\mathcal{L} = \sum_{i=1}^{M} d(V(X_i), P_X(D_i))$$

where d represents the Chamfer distance [16] between two point sets. Optimization ensures that the transformed depth maps align with the metric-scale geometry of human models.

However, depth estimation results are often uncertain, particularly at object boundaries. To address this, we estimate the uncertainty of depth values and retain only reliable information. We apply color jittering transformations (hue transformations) [13] to the input image and obtain multiple depth values for each pixel. We calculate uncertainty following [10] and only valid depth values are preserved for subsequent steps.

Reconstruct Individual Objects To reconstruct the 3D scene, we adopt a strategy that restores individual objects from the video as 3D meshes M_j and places them accurately within the 3D space. Our process begins by obtaining instance segmentation [1] results from the provided video frames. However, due to occlusions caused by foreground objects or human movement, these initial segmentation results are often incomplete or imprecise. We address this limitation by employing an image completion algorithm [15] to refine the segmentation and generate a more complete image for each object. Given these refined segmentation results, we then apply an Image-to-3D reconstruction method [20, 22] to obtain initial 3D object meshes M_j with textures for each instance.

Object Scale and Pose Refinement Individually reconstructed objects M_j exhibit inaccuracies in scale and pose. Empirically, we observe that reconstructed objects align well with the gravity, but require refinement in translations t_j, rotations r_j, and scales s_j. We address these issues by optimizing it using metric-scale depth maps D.

For each object mesh M_j, we sample visible surface points in camera space, denoted as $V(M_j)$. Then, these points are transformed using a learnable variable t_j, r_j, and s_j. We also extract corresponding points from the metric-scale depth map D using the object's segmentation mask, denoted as $P_j(D)$. After initializing the object's translation t_j using the centroid of $P_j(D)$, we optimize the object's scale s_j, rotations r_j, and translation t_j by minimizing:

$$\mathcal{L} = d(V(M_j), P_j(D))$$

where d represents the Chamfer distance between two point sets.

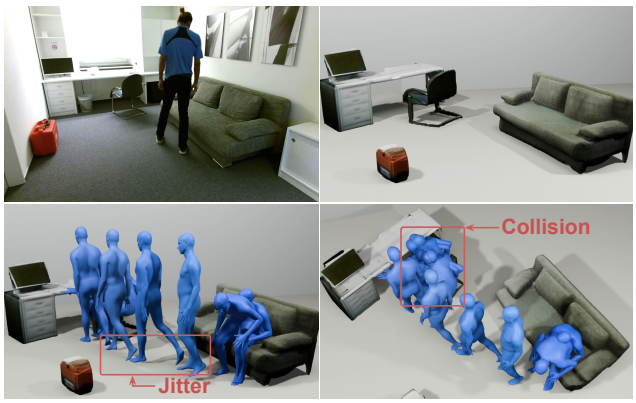

Figure 1. Results from the *initial stage* of Video2Animation. Starting from the input video [4] (top left), we reconstruct the scene geometry (top right) and the corresponding human motion (bottom) in metric scale.

3.2. Refinement Stage

Following the initial stage of motion and scene geometry reconstruction in a metric scale, several challenges remain in motion estimation, including potential scene collisions, motion jittering, and inconsistencies inherent to image-based motion extraction algorithms, as shown in Figure 1. We address these issues by leveraging a 3D motion prior by applying our SceneMI module.

Keyframe Optimization We optimize keyframes at regular 5-frame intervals, concentrating on root translation γ where motion estimation errors predominantly occur. The optimization leverages five complementary loss functions:

Regularization Loss constraints large deviations from the initial guess, ensuring optimization stability. *Contact Loss* estimates contact vertices [23] from human meshes X_i, encouraging precise alignment with scene geometry while penalizing non-contact vertex penetrations. *Temporal Smoothing Loss* minimizes consecutive root translation differences, encouraging smooth transitions between frames. *Depth Matching Loss* aligns visible human mesh points with metric-scale depth estimations using Chamfer distance minimization.

Applying SceneMI Following keyframe optimization, we progressively refine overall motion sequences using SceneMI. We sample one keyframe from every three optimized keyframes, corresponding to a 15-frame interval in the original video. By leveraging scene geometry and the poses derived from keyframes, we reconstruct the final animation that integrates geometric constraints, enhancing both realism and physical plausibility, as shown in Figure 2.

As SceneMI limits motion sequence synthesis to length $N = 121$, we employ an autoregressive strategy to synthesize continuous and natural human motion across extended

sequences. For keyframes representing arbitrary motion lengths, we divide sequences into N-length segments with v frame overlaps, where $v = 60$. We iteratively synthesize motion by using the final v frames of a prior episode as initial keyframes for the subsequent segment. After synthesizing the first motion sequence, we utilize its last v frames as keyframes for the start of the subsequent segment. For the remaining $N - v$ frames, motion is synthesized based on the corresponding keyframes from the current segment.

This progressive approach enables motion synthesis across long sequences, overcoming SceneMI's length constraints while maintaining scene awareness and motion consistency. This autoregressive approach allows applicability to real-world videos with arbitrary-length inputs.

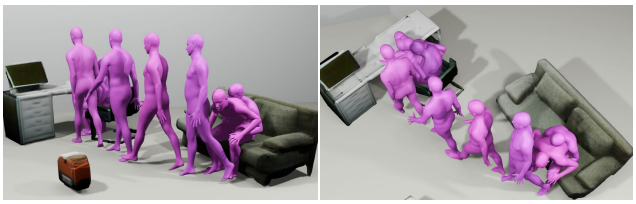

Figure 2. The final results from the Video2Animation pipeline demonstrate the reconstruction of 3D human-scene animation from monocular video inputs. By incorporating SceneMI with the obtained scene information and optimized keyframes, we reconstruct natural and physically plausible motions. For additional results, please refer to the supplementary video.

4. Additional Results

4.1. Evaluation across Multiple Seeds

Since our model is generative, we repeat our major experiments in Table 1 and Table 3 in the main paper across 20 different random seeds. We report their mean value, with 95% statistical confidence interval in Table 5. The observed variance is marginal, demonstrating the stability of our method.

Configurations	FID ↓	Jerk (m/s^3) ↓	MJPE (m) ↓	Collision Ratio ↓
Tab.1 (ours)	0.123	0.194	0.023	0.113
+ 20 runs	$0.123^{\pm 0.001}$	$0.193^{\pm 0.002}$	$0.023^{\pm 0.001}$	$0.114^{\pm 0.002}$
Tab.3 (ours)	0.118	0.198	0.012	0.108
+ 20 runs	$0.118^{\pm 0.001}$	$0.198^{\pm 0.003}$	$0.012^{\pm 0.001}$	$0.109^{\pm 0.002}$

Table 5. Evaluation across multiple random seeds. We report the mean and 95% confidence intervals for key metrics over 20 runs.

4.2. Integration with Frame-Based HSI

To further explore the applicability of our method, we integrate our module with a semantic keyframe generation approach. Specifically, we generated multiple sparse keyframes (colored in blue) using COINS [24] to provide semantic cues in various scenes, then applied our model to synthesize the complete motion sequence. The Figure 3 show our method generates coherent and plausible motions despite the semantic sparsity of the input keyframes.

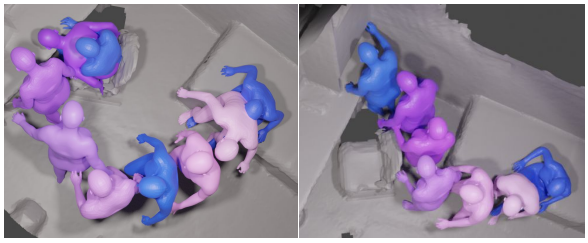

Figure 3. Integration with semantically generated keyframes. Our model produces plausible motions from sparse, semantic keyframes.

4.3. Long-Term Keyframe Interval

We also additionally provide an example with a 4-second keyframe interval, where only the start and end frames are given. As shown in Figure 4, the synthesized motion successfully navigates complex scenes with large obstacles, demonstrating our model successfully handles a long motion sequence that navigates around large obstacles.

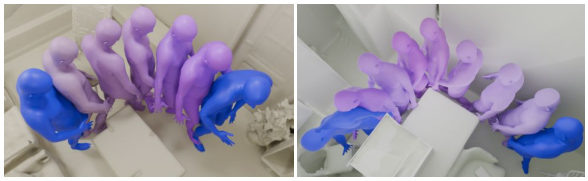

Figure 4. Result with a long-term keyframe interval. The model synthesizes long-horizon motion while avoiding large obstacles.

4.4. Discussion of Failure Cases

While our method generalizes well to unseen configurations and scene geometries beyond the training data, we acknowledge certain failure cases. First, failure can occur in rare human-scene interaction patterns such as "squeezing through a narrow passage," where the required motion rarely observed in training. Second, performance degrades in real-world scenes with highly complex or noisy geometric reconstructions, where subtle spatial constraints may not be fully captured by scene encoding. Figure 5 illustrates representative failure cases.

Figure 5. Failure cases. (Left) Unseen interaction pattern (e.g., squeezing through narrow space). (Right) Real-world scene with noisy or complex geometry.

References

[1] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. 2022. 3

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

[3] Shubham Goel, Georgios Pavlakos, Jathushan Rajasegaran, Angjoo Kanazawa*, and Jitendra Malik*. Humans in 4D: Reconstructing and tracking humans with transformers. In *International Conference on Computer Vision (ICCV)*, 2023. 2

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

[5] Jonathan Ho, Ajay Jain, and Pieter Abbeel. Denoising diffusion probabilistic models, 2020. 1

[6] Siyuan Huang, Zan Wang, Puhao Li, Baoxiong Jia, Tengyu Liu, Yixin Zhu, Wei Liang, and Song-Chun Zhu. Diffusion-based generation, optimization, and planning in 3d scenes. *arXiv preprint arXiv:2301.06015*, 2023. 1

[7] Yiheng Huang, Yang Hui, Chuanchen Luo, Yuxi Wang, Shibiao Xu, Zhaoxiang Zhang, Man Zhang, and Junran Peng. Stablemofusion: Towards robust and efficient diffusion-based motion generation framework. *arXiv preprint arXiv:2405.05691*, 2024. 1

[8] Linyi Jin, Jianming Zhang, Yannick Hold-Geoffroy, Oliver Wang, Kevin Matzen, Matthew Sticha, and David F. Fouhey. Perspective fields for single image camera calibration. In *CVPR*, 2023. 2

[9] Korrawe Karunratanakul, Konpat Preechakul, Supasorn Suwajanakorn, and Siyu Tang. Guided motion diffusion for controllable human motion synthesis. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 2151–2162, 2023. 1

[10] Junho Lee, Sang Min Kim, Yonghyeon Lee, and Young Min Kim. Nfl: Normal field learning for 6-dof grasping of transparent objects. *IEEE Robotics and Automation Letters*, 9(1): 819–826, 2024. 3

[11] Matthew Loper, Naureen Mahmood, Javier Romero, Gerard Pons-Moll, and Michael J. Black. SMPL: A skinned multi-person linear model. *ACM Trans. Graphics (Proc. SIGGRAPH Asia)*, 34(6):248:1–248:16, 2015. 2

[12] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization, 2019. 1

[13] TorchVision maintainers and contributors. Torchvision: Pytorch's computer vision library. https://github.com/pytorch/vision, 2016. 3

[14] Alex Nichol and Prafulla Dhariwal. Improved denoising diffusion probabilistic models, 2021. 1

[15] Ege Ozguroglu, Ruoshi Liu, Dídac Surś, Dian Chen, Achal Dave, Pavel Tokmakov, and Carl Vondrick. pix2gestalt: Amodal segmentation by synthesizing wholes. *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2024. 3

[16] Nikhila Ravi, Jeremy Reizenstein, David Novotny, Taylor Gordon, Wan-Yen Lo, Justin Johnson, and Georgia Gkioxari. Accelerating 3d deep learning with pytorch3d. *arXiv:2007.08501*, 2020. 3

[17] Guy Tevet, Sigal Raab, Brian Gordon, Yoni Shafir, Daniel Cohen-or, and Amit Haim Bermano. Human motion diffusion model. In *The Eleventh International Conference on Learning Representations*, 2023. 1, 2

[18] Jiashun Wang, Huazhe Xu, Jingwei Xu, Sifei Liu, and Xiaolong Wang. Synthesizing long-term 3d human motion and interaction in 3d scenes. *arXiv preprint arXiv:2012.05522*, 2020. 1

[19] Yiming Xie, Varun Jampani, Lei Zhong, Deqing Sun, and Huaizu Jiang. Omnicontrol: Control any joint at any time for human motion generation. *arXiv preprint arXiv:2310.08580*, 2023. 1, 2

[20] Jiale Xu, Weihao Cheng, Yiming Gao, Xintao Wang, Shenghua Gao, and Ying Shan. Instantmesh: Efficient 3d mesh generation from a single image with sparse-view large reconstruction models. *arXiv preprint arXiv:2404.07191*, 2024. 3

[21] Lihe Yang, Bingyi Kang, Zilong Huang, Xiaogang Xu, Jiashi Feng, and Hengshuang Zhao. Depth anything: Unleashing the power of large-scale unlabeled data. In *CVPR*, 2024. 2

[22] Xianghui Yang, Huiwen Shi, Bowen Zhang, Fan Yang, Jiacheng Wang, Hongxu Zhao, Xinhai Liu, Xinzhou Wang, Qingxiang Lin, Jiaao Yu, Lifu Wang, Zhuo Chen, Sicong Liu, Yuhong Liu, Yong Yang, Di Wang, Jie Jiang, and Chunchao Guo. Tencent hunyuan3d-1.0: A unified framework for text-to-3d and image-to-3d generation, 2024. 3

[23] Jason Y. Zhang, Sam Pepose, Hanbyul Joo, Deva Ramanan, Jitendra Malik, and Angjoo Kanazawa. Perceiving 3d human-object spatial arrangements from a single image in the wild. In *European Conference on Computer Vision (ECCV)*, 2020. 3

[24] Kaifeng Zhao, Shaofei Wang, Yan Zhang, Thabo Beeler, , and Siyu Tang. Compositional human-scene interaction synthesis with semantic control. In *European conference on computer vision (ECCV)*, 2022. 4