SceneMI: Scene-aware Motion In-betweening

Supplementary Material

In the supplementary materials, we elaborate the implementation details for our SceneMI (Sec. 1), further experiments on synthetic noisy dataset (Sec. 2), ablation on scene feature design (Sec. 3), and through Video2Animation algorithm (Sec. 4). Please refer to our supplementary video for additional qualitative results.

1. Further Details

1.1. Implementation Details

We implemented our model using a DDPM-based diffusion framework, leveraging the U-Net architecture proposed by [7] with the AdamW optimizer with a learning rate of $1\mathrm{e}{-4}$ and a weight decay of $1\mathrm{e}{-2}$. 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 [10]
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 Model

1.2. Baseline Details

We compare our approach to several state-of-the-art diffusion-based motion in-betweening methods, including MDM [13], OmniControl [14], and CondMDI [2]. Since these baselines were originally designed for text-to-motion tasks, we adapt them by replacing their text encoders with a Vision Transformer (ViT)-based global scene encoder that processes global scene features $\mathbf{c}_g$.

For MDM, we further modify its motion representation to incorporate a global root representation that supports global keyframes and employ a U-Net architecture [7]. We perform motion in-betweening by imputing joint positions at each diffusion step, except for the final step. OmniControl and CondMDI are similarly adapted by integrating the global scene encoder while maintaining their original training protocols. Across all baselines, we systematically utilize static keyframe poses—such as joint position information—to predict intermediate motions.

2. Robustness on Synthetic Dataset

We evaluate the performance of scene-aware motion in-betweening under noisy keyframe conditions using a synthetic noisy test set in Sec. 4.2. Table 2 presents the quantitative results for motion in-betweening under dense, noisy keyframe setup with an interval of $r = 3$ and a noise level of $l = 1$. Our model trained on noisy data with a two-stage diffusion scheme, consistently outperforms baseline models across all metrics representing motion quality. By leveraging the inherent denoising properties of diffusion models, our approach produces cleaner and higher-quality motion outputs. In contrast, baseline models tend to follow the noisy keyframes directly, resulting in lower-quality motion, as reflected in the evaluation metrics. This highlights our model's capacity to generate smooth and accurate motion in-betweening, effectively mitigating noise and enhancing overall motion quality. Consequently, this leads to results that are more physically plausible, with motions that interact naturally and realistically within the scene.

3. Ablation on Scene Encoding

We investigate the performance of our scene encoding—with global feature $\mathbf{c}_g$ and local feature $\mathbf{c}_l$—in generating scene-aware motion. We compare our method against several model ablations: Ours w/o scene-awareness (excluding $\mathbf{c}_g$ and $\mathbf{c}_l$), Ours w/o global feature (excluding $\mathbf{c}_g$), and Ours w/o local feature (excluding $\mathbf{c}_l$). As demonstrated in In Table 3, each feature contributes to synthesizing scene-aware motion, particularly for metrics related to motion in-betweening tasks such as *MJPE* or physical plausibility, demonstrating its essential role in our scene-aware motion in-betweening approach.

4. Applications: Video2Animation

In this section, we present a Video2Animation 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 which 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

Method	FID ↓	Foot skating ↓	Jerk (m/s^3) ↓	MJPE key (m) ↓	MJPE all (m) ↓	Collision Frame Ratio ↓	Pene. Max (m) ↓
MDM [13]	5.149	0.761	22.169	0.285	0.279	0.151	0.056
OmniControl [14]	2.981	0.381	2.198	0.302	0.308	0.169	0.058
CondMDI [2]	3.136	0.317	0.296	0.354	0.349	0.187	0.059
Ours w/o noise-awareness	0.157	0.265	0.230	0.015	0.014	0.119	0.046
Ours	**0.118**	**0.247**	**0.198**	**0.013**	**0.012**	**0.108**	**0.042**

Table 2. Quantitative evaluation of the scene-aware motion in-betweening with noisy, dense keyframes, using an interval of $r = 3$ and a noise level of $l = 1$ on [5].

Method	FID ↓	Foot skating ↓	Jerk (m/s^3) ↓	MJPE key (m) ↓	MJPE all (m) ↓	Collision Frame Ratio ↓	Pene. Max (m) ↓
Ours w/o scene-awareness	0.136	0.251	**0.103**	0.012	0.059	0.131	0.049
Ours w/o global feature $\mathbf{c}_g$	0.138	0.254	0.131	0.011	0.051	0.128	0.048
Ours w/o local feature $\mathbf{c}_l$	0.125	**0.245**	0.196	0.008	0.036	0.119	0.045
Ours	**0.123**	0.248	0.194	**0.006**	**0.023**	**0.113**	**0.043**

Table 3. Ablation study of scene encoding $\mathbf{c}_g$, $\mathbf{c}_l$. Tested with a sparse keyframe interval of $r = 60$ on [5].

reconstructed scene geometry and our SceneMI module. The following sections detail the challenges and methodologies for each stage.

4.1. 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 [6]. These parameters are crucial for positioning 3D human meshes or backprojecting 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, 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 [16] 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 [12] 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) [9] to the input image and obtain multiple depth values for each pixel. We calculate uncertainty following [8] 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 [11] to refine the segmentation and generate a more complete view for each object. Given these refined segmentation results, we then apply an Image-to-3D reconstruction method [15, 17] to generate initial 3D object meshes M_j with textures for each instance.

Object Scale and Pose Refinement Individually reconstructed objects M_j often exhibit inaccuracies in scale and pose. Empirically, we observe that reconstructed objects align well with the gravity and y-axis rotations, but require refinement in translations t_j and scales s_j. We optimize these parameters using metric-scale depth maps D.

For each object mesh M_j, we sample visible surface points in camera space, denoted as $V(M_j)$. 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 using the centroid of $P_j(D)$, we optimize the object's scale s_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.

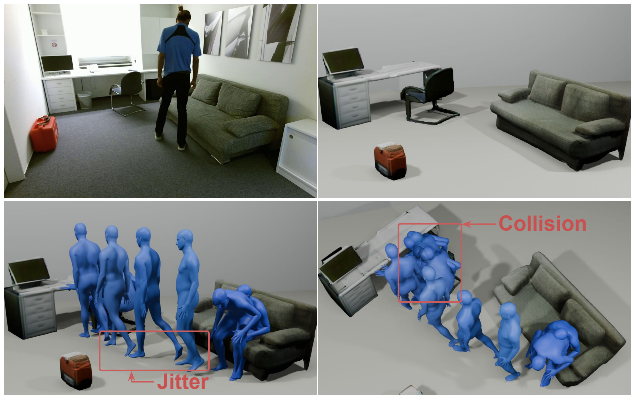

Figure 1. Results from the initial stage of Video2Animation. Starting from the input video [4] (top left), we reconstruct the scene geometry in metric scale (top right) and the corresponding human motion (bottom).

4.2. 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 where motion estimation errors predominantly occur. The optimization leverages four complementary loss functions:

Regularization Loss constraints large deviations from the initial guess, ensuring optimization stability. *Contact Loss* estimates contact vertices [18] 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 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 generates the final animation that integrates geometric constraints, enhancing both realism and physical plausibility, as shown in Figure 2.

As SceneMI limits motion sequence generation 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 generate motion by using the final v frames of a prior episode as initial keyframes for the subsequent segment. After generating 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 generated 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.

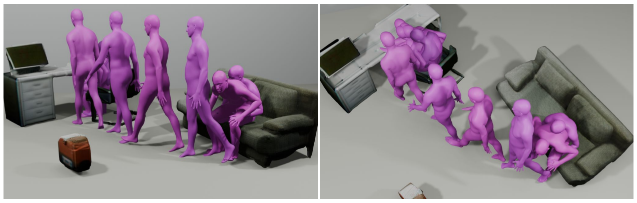

Figure 2. Final results from the Video2Animation pipeline. We reconstruct physically plausible human motion and scene geometry from a monocular video. For additional results, please refer to the supplementary video.

References

[1] Bowen Cheng, Ishan Misra, Alexander G. Schwing, Alexander Kirillov, and Rohit Girdhar. Masked-attention mask transformer for universal image segmentation. 2022. 2

[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] Nan Jiang, Zhiyuan Zhang, Hongjie Li, Xiaoxuan Ma, Zan Wang, Yixin Chen, Tengyu Liu, Yixin Zhu, and Siyuan Huang. Scaling up dynamic human-scene interaction modeling. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 1737–1747, 2024. 2

[6] 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

[7] 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

[8] 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. 2

[9] TorchVision maintainers and contributors. Torchvision: Pytorch's computer vision library. `https://github.com/pytorch/vision`, 2016. 2

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

[11] 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

[12] 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. 2

[13] 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

[14] 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

[15] 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

[16] 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

[17] 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

[18] 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