PIAvatar: Physically Interactive Avatars via Deformation Gradient Decoupling

Sang-Hun Han[1], Min-Gyu Park[2,3], Jisu Shin[1], Seunghyun Shin[1], Jin-Hwi Park[4], and Hae-Gon Jeon[5†]

[1]GIST, [2]KETI, [3]Polygom, [4]Chung-Ang University,
[5]Department of Artificial Intelligence, Yonsei University
{sanghunhan, jsshin98, seunghyuns98}@gm.gist.ac.kr,
mpark@keti.re.kr, jinhwipark@cau.ac.kr, earboll@yonsei.ac.kr

Abstract. 3D human avatars have shown impressive visual fidelity driven by pose-conditioned models, yet they still lack the physical ability required for interactions with each other and environments. Although recent studies have made various attempts to incorporate physical characteristics into 3D avatars, they only exhibit limited physical deformations, often leading to constrained interaction behaviors. To resolve this issue, we present PIAvatar, a framework to simultaneously enable physically aware interactions between avatar-avatar and avatar-environment, and a non-rigid deformable human body simulation. In this work, our key insight is to decouple kinematic velocity from deformation gradient. When external forces act on avatars, the kinematic velocity induces stress which hinders the avatar's ability to achieve a desired pose. In addition, we integrate a skeletal framework within the avatar. It allows estimating its poses and real-time tracking in a closed form, even during non-rigid physical interactions. Our approach is implemented within a conventional Material Point Method framework to ensure physically consistent dynamics. We lastly evaluate the method on both human-object and human-human interaction scenarios to assess its behavior under diverse interaction settings.

Keywords: 3D Avatar · Physical Interaction · Material Point Method

1 Introduction

The generation of realistic 3D human avatars is crucial for bridging virtual and real-world experiences, enabling lifelike interactions in diverse AR/VR applications. Prior research has primarily focused on detailed human geometry and appearance, leveraging various representations such as implicit and volumetric models, 3D Gaussians [28], and parametric body models [1, 37, 45, 71]. These approaches have shown promising visual fidelity, producing realistic renderings across a wide range of body configurations. However, since physical properties are not built in these models, we can access only kinematic animations of avatars with no physical interactions. That is, such avatars remain fundamentally incapable of interacting with each other and their surrounding environments.

† Corresponding author.

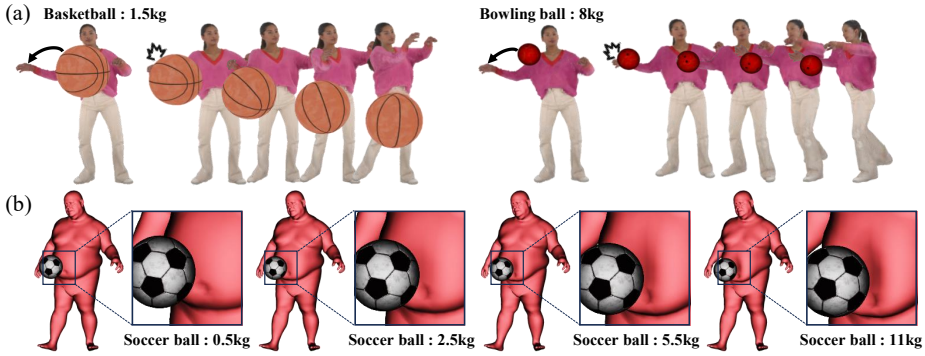

Fig. 1: Physical avatar interactions with bidirectional and non-rigid deformations. PIAvatar enables bidirectional interactions between avatars and their surroundings within a unified simulation framework. (a) Varying the mass of the ball changes both the avatar pose and the ball trajectory, illustrating bidirectional human–object interaction. (b) The avatar exhibits different non-rigid deformations depending on the mass of the soccer ball during impact.

Several studies have recently begun to explore physically aware avatars. Reinforcement learning (RL)–based approaches [61,72] allow avatars to interact with objects. Due to their large-scale parallel training setting, they usually rely on simplified physics environments [6,7,49,62]. This constraint limits their ability to represent non-rigid deformations limiting their expressiveness to simplified geometric forms such as cylinders or other primitive shapes. Meanwhile, simulation-driven methods, such as dressed avatar physics [32,74] or kinematic models [58] augmented with partial dynamics, take promising steps toward physical realism. They either allow one-way interactions, where the avatar can exert forces on surrounding objects but not vice versa, or fail to simulate physical surface deformations of the avatar. For example, Half-Physics [58] allows external forces to influence the avatar, but its responses remain at the pose level without modeling surface deformation.

To address these limitations, we propose PIAvatar, a framework for physically consistent human avatar interactions. As illustrated in Fig. 1, our method is developed with two primary objectives: (1) *Bidirectional physical interactions* including avatar-avatar and avatar-object; and (2) *Non-rigid deformation* to undergo natural interactions under physical forces. To achieve this, we build our simulation system based on the Material Point Method (MPM), a particle-based continuum simulation framework. Even though MPM inherently supports non-rigid deformation and bidirectional momentum transfer, it introduces two key challenges when driving an avatar with user-defined motions. First, all objects inside MPM, modeled with a stress-based constitutive model, naturally induce restorative forces when deformed. The restorative forces counteract the externally applied kinematic velocity and hinder avatars from reaching the intended target pose. Second, once the avatar undergoes non-rigid deformation through environmental interactions, its pose is no longer directly trackable. While forward animation can easily generate deformations from a given pose, the inverse

process—recovering pose from a physically deformed shape—is non-trivial and often ill-posed. Consequently, it is typically solved through computationally expensive methods such as parametric model fitting [1,37,45], non-linear optimization [4,36], or learning–based regression [25,31].

To overcome these challenges, we propose two approaches. We explicitly exclude a user-defined kinematic velocity from a deformation gradient computation. This prevents undesired stress and enables the avatar to follow the intended motion without the restorative resistance. We then incorporate the deformation process into the MPM framework. In addition, we embed a skeletal structure inside an avatar and compute its pose in a closed form, enabling robust and direct pose estimation even under non-rigid physical interactions. With this integrated design, our system enables physically consistent and controllable interactions between avatars and their surrounding environments.

We demonstrate the capabilities of PIAvatar through various avatar–environment interaction scenarios, including both avatar-object and avatar-avatar contacts. Our avatars exhibit physical, non-rigid body deformations under collisions and external forces. We also show that varying material properties, such as density, produce diverse and physically consistent simulation outcomes.

In summary, our contributions are as follows:

1. We present PIAvatar, a novel MPM-based avatar simulation framework that enables both bidirectional physical interactions and non-rigid deformations.
2. For this, we decouple the deformation gradient to ensure the user-defined kinematic velocity directly drives the avatar's motion without inducing stress while still allowing physical interactions, and embed a skeletal structure for direct pose estimation. Note that these whole processes are solved in a closed form optimization, not in a learning-based manner.
3. We demonstrate physically grounded interactions across multiple human–object and human–human scenarios using PIAvatar.

2 Related Work

2.1 Animatable Human Avatar Generation

Building on parametric human body models [37, 46], numerous studies have explored the reconstruction of posed human avatars directly from single images [11, 14, 53, 54, 69, 70, 73, 75]. However, those works typically focus on static or per-frame reconstruction without explicit animation capability. To achieve animatable avatars, subsequent works leverage Linear Blend Skinning (LBS)-based deformation to integrate geometric or texture information from multiple images into a canonical space [9,23,47,48,56,60,65]. Recently, 3D or 4D Gaussian Splatting (GS) [28,66] is incorporated into this framework, offering efficient rendering and high-quality geometry reconstruction [15,16,34,43,44,52,55,64]. With these advances, recent approaches [51,57,63,76] directly infer pose-dependent geometry and appearance without any per-scene optimization using large feed-forward models, allowing scalable and real-time animatable avatar generations.

Additionally, skeleton-based methods such as OSSO [27] and SKEL [26] have explored explicit skeletal representations to enhance articulation control and motion coherence. Such approaches demonstrate the growing integration of skeleton-driven modeling within animatable human representations, laying the groundwork for physically coupled simulation as pursued in our work.

2.2 Physics-based Avatar Simulation

From a perspective of physics-based methods, relevant studies can be broadly categorized into control-driven approaches and simulation-based deformable modeling. Control-driven methods, including RL-based approaches [38, 39, 49, 50, 61, 72], learn policies to generate physically plausible motions within physics simulators. These methods focus on task-oriented control and reactive behaviors, typically relying on rigid-body or simplified physical representations for efficiency [6, 7, 62]. Simulation-based frameworks, including Finite Element Method (FEM) [5], C-IPC [33] and MPM [22], have tried to represent animatable human motions based on laws of physics such continuum mechanics. Half-Physics [58] couples a kinematic human model with a physics engine to enable pose-level adaptation under external forces. C-IPC-based methods, which are mesh-based deformable simulation like PhysAvatar [74], incorporate physics-inspired modeling to capture garment deformation effects. FEM-based approaches [29, 35], explicitly model nonlinear soft-tissue dynamics using continuum mechanics formulations. MPM-based approaches, such as MPMAvatar [32], use a particle–grid formulation to model volumetric deformation and simulate loose garments. While these works advance physics-aware avatar modeling, achieving effective bidirectional interaction still remains challenging.

2.3 Material Point Method

The MPM [22] is a hybrid Eulerian-Lagrangian framework, introduced by Stomakhin *et al.* [59] for snow simulation, and has become a core technique for physically based animation of solids, fluids, and deformable bodies. In the MPM, each particle carries a deformation gradient that couples the local velocity gradient with its deformation and stress.

To retain robustness, later studies such as APIC [20], AMPIC [21], and MLS-MPM [17] enhanced momentum conservation and reduced numerical dissipation of kinetic energy. Other studies decomposed the deformation gradient into elastic and plastic parts for constitutive modeling *e.g.* elasto-plastic snow [59] and thin-shell rigidity corrections [10]. Further developments extended the capability of MPM into anisotropic, frictional, and hybrid soft-body materials [12, 19, 30]. In parallel, Gaussian-based representations have been combined with physics-inspired modeling, such as PhysGaussian [68], to enhance visual realism through a differentiable rendering and an appearance-driven regularization. Overall, these advances have established the MPM as a versatile framework for physically grounded simulation. These properties make it a relevant foundation for modeling physically aware non-rigid human avatar interactions.

3 Preliminary: Material Point Method

In this work, we adopt the MPM framework to take advantage of its fast grid-based momentum exchange to efficiently model frequent interactions. It also enables flexible deformations and complex contact interactions within a unified particle-grid framework.

To firstly explain a motion of a material space over time t, we begin with the conservation of mass and momentum as follows:

$$\frac{d\rho}{dt} + \rho\nabla \cdot \boldsymbol{v} = 0, \qquad \rho\frac{d\boldsymbol{v}}{dt} = \nabla \cdot \boldsymbol{\sigma} + \rho\boldsymbol{g}, \tag{1}$$

where ρ, $\boldsymbol{v}$, and $\boldsymbol{\sigma}$ denote the material density, velocity, and Cauchy stress, respectively. Following Eq. (1), continuum mechanics models the motion as a continuous deformation mapping $\boldsymbol{\phi}$, which takes spatial deformed coordinates $\boldsymbol{x} = \boldsymbol{\phi}(\boldsymbol{X}, t)$ from material coordinates $\boldsymbol{X}$. Its deformation gradient $\boldsymbol{F} = \partial\boldsymbol{\phi}/\partial\boldsymbol{X}$ encodes local transformations of the material, including rotation, stretch, and shear. Numerically solving these continuum equations in a purely Lagrangian form is computationally demanding under large deformations and frequent contact, which motivates the hybrid particle-grid formulation of MPM.

To model continuum mechanics, the Material Point Method (MPM) discretizes the continuum into Lagrangian material particles that track mass, momentum and deformation. Each particle p is represented by its position $\boldsymbol{x}_p$, velocity $\boldsymbol{v}_p$, mass m_p, and deformation gradient $\boldsymbol{F}_p$. An Eulerian grid solves the momentum equations and updates particle states through particle-grid transfers, where particle stresses are accumulated onto grid nodes to compute internal forces. While each particle computes its stress independently, the Cauchy stress $\boldsymbol{\sigma}_p$ is obtained from its deformation gradient $\boldsymbol{F}_p$ and the strain energy density $\Psi(\cdot)$ as:

$$\boldsymbol{\sigma}_p = \frac{1}{J_p}\boldsymbol{F}_p \frac{\partial\Psi(\boldsymbol{F}_p)}{\partial\boldsymbol{F}_p}^{\top} \text{ s.t. } J_p = \det(\boldsymbol{F}_p). \tag{2}$$

However, in Eq. (2), a challenge arises since all types of deformations are encoded with only $\boldsymbol{F}_p$, including user-imposed or purely kinematic deformations. In this paper, our solution to this challenge is to disentangle $\boldsymbol{F}_p$ into the kinematic part and the elastic part, inspired by snow and thin-shell material simulations [10,59].

4 Proposed Method

We first point out the problem in the basic MPM coming from stress-based forces, and present the solution that introduces the disentanglement of the user-defined kinematic velocity from the deformation gradient (Sec. 4.1). We then introduce how to simulate physical avatar interactions with our skeletal pose extraction and kinematic velocity calculation (Sec. 4.2). An overview of PIAvatar is illustrated in Fig. 2.

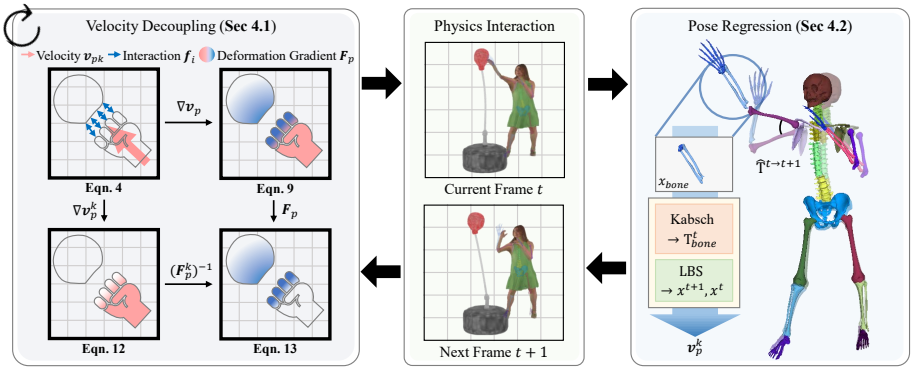

Fig. 2: An illustration of our framework. (a) To faithfully reflect the user-defined motion, we decouple the kinematic velocity from the deformation gradient update (Sec. 4.1). (b) By computing the velocity from the transformations of the embedded skeletal structure, our method preserves the pose consistency throughout the simulation (Sec. 4.2).

4.1 Kinematic Deformation Decoupling

As shown in Fig. 3, we illustrate a common process of MPM. The particle-to-grid (P2G) and grid-to-particle (G2P) transfers allow us to compute physical quantities of a particle on a certain grid. When the velocity is assigned to particles, they are interpolated to assign the velocity onto the grid during P2G. As a result, the grid treats the velocity as physical momentum, which updates the deformation gradient $\boldsymbol{F}_p$ during G2P and produces the stress $\boldsymbol{\sigma}_p$.

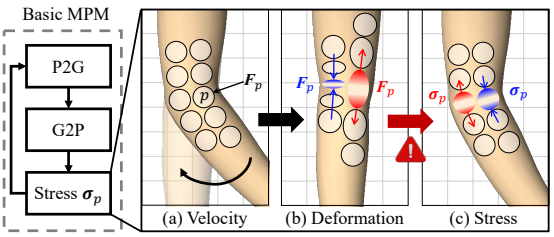

Fig. 3: Velocity-induced stress formation. (a) The kinematic velocity is applied to the avatar particles p. (b) The change in deformation gradients $\boldsymbol{F}_p$ occurs due to the kinematic velocity (see blue- and red-colored ellipses). (c) The deformation generates stress $\boldsymbol{\sigma}_p$ that hinders intended kinematic motion.

In particular, in non-rigid body motions like humans, the user-defined kinematic velocity is partially absorbed as the deformation gradient. This phenomenon results in the unintended stress that hinders accurate kinematic controls and leads to deviations from the desired pose. We point out this problem in a basic MPM with respect to the representational ability of user-preferred motions by describing its key components at first.

Basic MPM. In the P2G phase of a basic MPM, particle quantities, including position $\boldsymbol{x}_p$, velocity $\boldsymbol{v}_p$, mass m_p and the Cauchy stress $\boldsymbol{\sigma}_p$, are interpolated to

the grid using a weighting kernel w_{ip}:

$$m_i = \sum_p w_{ip}\, m_p, \quad m_i \boldsymbol{v}_i = \sum_p w_{ip}\, m_p \boldsymbol{v}_p + \Delta t\, \boldsymbol{f}_i, \tag{3}$$

$$\boldsymbol{f}_i = \boldsymbol{f}_i^{\text{int}} + \boldsymbol{f}_i^{\text{ext}} \quad \text{s.t.} \quad \boldsymbol{f}_i^{\text{int}} = -\sum_p V_p\, \boldsymbol{\sigma}_p\, \nabla w_{ip}, \tag{4}$$

where i and V_p denote the index of a grid node and the particle volume, respectively. Here, the weighting kernel w_{ip} is the B-spline weight evaluated at the particle position relative to the i-th grid node. The total force $\boldsymbol{f}_i$ on each grid is computed by summing external forces like gravity and the internal forces like stress that we want to handle.

In the G2P phase, the velocity $\boldsymbol{v}_i$ on the grid is applied to the particle, updating the particle state such as $\boldsymbol{F}_p$:

$$\boldsymbol{v}_p \leftarrow \sum_i w_{ip}\, \boldsymbol{v}_i, \qquad \boldsymbol{x}_p \leftarrow \boldsymbol{x}_p + \Delta t\, \boldsymbol{v}_p, \tag{5}$$

$$\nabla \boldsymbol{v}_p = \sum_i \boldsymbol{v}_i (\nabla w_{ip})^\top, \quad \boldsymbol{F}_p \leftarrow (\mathbf{I} + \Delta t \nabla \boldsymbol{v}_p)\, \boldsymbol{F}_p. \tag{6}$$

In the MPM model, users can adjust $\boldsymbol{v}_p$ to control avatars, which contributes to the update of $\boldsymbol{F}_p$ in Eq. (6). According to Eq. (2), the stress is obviously generated, introducing $\boldsymbol{f}_i^{\text{int}}$ in Eq. (4) in the next P2G step.

Disentanglement of Deformation Gradient. In this paper, our key insight is to decouple the kinematic velocity from the deformation gradient update (Eq. (6)). To do this, we decompose $\boldsymbol{v}_p$ into a kinematic velocity $\boldsymbol{v}_p^k \in \mathbb{R}^{N\times 3}$, where N denotes the number of particles.

As a first step, we compute a velocity grid $\boldsymbol{v}_i^k$ separated solely from the kinematic velocity. Similar to the basic MPM (Eq. (3)), we conduct P2G step to transfer $\boldsymbol{v}_p^k$ to a kinematic grid velocity field $\boldsymbol{v}_i^k$. By transferring $\boldsymbol{v}_p^k$ and its associated particle mass m_p^k onto grid nodes, we compute their mass m_i^k and kinematic momentum $m_i^k \boldsymbol{v}_i^k$:

$$m_i^k = \sum_p w_{ip}\, m_p^k, \quad m_i^k \boldsymbol{v}_i^k = \sum_p w_{ip}\, m_p^k \boldsymbol{v}_p^k. \tag{7}$$

By dividing the $m_i^k \boldsymbol{v}_i^k$ into m_i^k, we obtain $\boldsymbol{v}_i^k$, and then repeat the same procedures in Eq. (6) as below:

$$\nabla \boldsymbol{v}_p^k = \sum_i \boldsymbol{v}_i^k\, (\nabla w_{ip})^\top, \quad \boldsymbol{F}_p^k \leftarrow \left(\mathbf{I} + \Delta t\, \nabla \boldsymbol{v}_p^k\right) \boldsymbol{F}_p^k, \tag{8}$$

Finally, we extract the kinematic deformation gradient $\boldsymbol{F}_p^k$ from the total deformation gradient $\boldsymbol{F}_p$ as a first-order approximation:

$$\boldsymbol{F}_p \leftarrow \boldsymbol{F}_p\, (\boldsymbol{F}_p^k)^{-1}. \tag{9}$$

By applying the above Eq. (9) into Eq. (2), we achieve the physically consistent stress responses, even with the conventional MPM framework. This update eliminates the unnecessary stress accumulation and contributes to stable, real-time physical avatar representation, while also allowing physical interactions.

4.2 Skeleton-based Pose Regression

In the physically interactive simulation, external forces are exerted onto an avatar, which yields its non-rigid deformations, such as shapes and poses. In Sec. 4.1, we can apply the kinematic velocity to animate the avatar as intended.

Next, we describe how to compute the kinematic velocity. Obviously, it can be obtained from a difference between adjacent pose sequences. Unfortunately, deformations make it hard to track poses over time because the physical interactions change the current pose so that it differs from the intended pose.

Our approach is thus to efficiently compute the kinematic velocity from a difference between adjacent pose sequences in real-time, suitable for MPM-based frameworks: (1) embedding a set of skeleton particles into avatar's surface to track its poses; (2) calculating avatar's kinematic velocity from the tracked poses. The embedded skeleton inherently represents joint structures, which ensures consistency between the human avatar surface and its poses.

Skeletal Pose Extraction. We introduce a direct approach that embeds a set of skeleton particles into the avatar's surface, where each joint is represented by a group of bone particles. When external forces act on the avatar, these forces are transferred from the surrounding surface to the corresponding bones, causing their particles' positions to change accordingly. By tracking the motion of these particles, we compute the pose of each bone, including its rotation $\mathbf{R}$ and translation $\mathbf{t}$. This representation ensures stable pose consistency and smooth motion propagation across adjacent joints. The computation is designed in a closed-form manner and solved via a simple least-squares optimization.

To compute the transformation matrix for each bone, we apply the Kabsch algorithm $K(\cdot)$ [24] to its corresponding bone particles as:

$$\mathbf{T}^t_{\text{bone}} = [\mathbf{R}^t_{\text{bone}}, \mathbf{t}^t_{\text{bone}}] = K\left(\boldsymbol{x}^{\text{cano}}_{\text{bone}}, \boldsymbol{x}^t_{\text{bone}}\right), \tag{10}$$

where $\mathbf{T}^t_{\text{bone}} \in \mathbb{R}^{J\times 4\times 4}$ is a set of per-joint transformation matrices at frame t. Here, J is the number of joints. It estimates the rotation $\mathbf{R}^t_{\text{bone}}$ and the translation $\mathbf{t}^t_{\text{bone}}$ to align the canonical bone particle positions $\boldsymbol{x}^{\text{cano}}_{\text{bone}}$ with their current positions $\boldsymbol{x}^t_{\text{bone}}$.

Kinematic Velocity Calculation. For subsequent frames, the avatar is animated according to the input pose transformation sequence $\hat{\mathbf{T}} \in \mathbb{R}^{T\times J\times 4\times 4}$, where T is the number of frames. The kinematic velocity of each particle, $\mathbf{v}^k_p$, is then computed from the positional differences between consecutive posed avatars.

For the stable kinematic velocity and the non-rigid physical interaction, we update the current pose using only incremental transformations in the sequence.

This formulation maintains consistent and stable kinematic velocities even under pose deformations induced by external forces. We first compute a relative (incremental) transformation from a current pose $\hat{\mathbf{T}}^t$ and its next pose $\hat{\mathbf{T}}^{t+1}$ as follows:

$$\hat{\mathbf{T}}^{t\to t+1} = (\hat{\mathbf{T}}^t)^{-1}\hat{\mathbf{T}}^{t+1}. \tag{11}$$

We then update it as below:

$$\mathbf{T}^{t+1}_{\text{bone}} = \hat{\mathbf{T}}^{t\to t+1}\mathbf{T}^{t}_{\text{bone}}. \tag{12}$$

Using Eq. (12), we obtain the consecutive avatar positions via Linear Blend Skinning (LBS) as follows:

$$\boldsymbol{x}^t = \text{LBS}(\mathbf{T}^t_{\text{bone}}, \boldsymbol{x}^{\text{cano}}),\ \boldsymbol{x}^{t+1} = \text{LBS}(\mathbf{T}^{t+1}_{\text{bone}}, \boldsymbol{x}^{\text{cano}}). \tag{13}$$

The kinematic velocity $\boldsymbol{v}_p^k$ for each particle is then determined from the positional difference as follows:

$$\boldsymbol{v}_p^k = \frac{\boldsymbol{x}^{t+1} - \boldsymbol{x}^t}{\Delta t}. \tag{14}$$

Finally, we plug Eq. (14) into Eq. (7) for general use.

5 Experimental Results

We demonstrate the superiority of the proposed PIAvatar over the basic MPM framework and its versatility through a variety of physically aware scenarios such as human-human and human-object interactions.

5.1 Experiment Details

We evaluate our method using two types of avatars: (1) a clothed Gaussian-based avatar using Animatable Gaussians (AG) [34] trained on 7 ActorsHQ [18] models, and (2) a parametric mesh-based avatar represented with SMPL-X [46]. Each avatar consists of 300,000 particles on average for AG, and 10,475 particles for SMPL-X. For animating avatars, we choose pose sequences from AMASS dataset [40]. To evaluate physically aware interactions, we use public 3D assets from BlenderNeRF [41] and Sketchfab[1]. We embed kinematic skeletons inside avatars via OSSO [27], which provides joint hierarchy, pose alignment and skinning weights. For computational efficiency, we downsample the 74,496 particles constituting the OSSO skeleton by 10× using voxel grid downsampling, thereby reducing both simulation time and the cost of the Kabsch computation. PIAvatar is built upon PhysGaussian [68], which is a baseline of our work to utilize the conventional MPM framework and support visualization of both meshes and 3D Gaussians. Our method runs 100 steps to render each frame. We adopt a 200^3 background grid unless otherwise stated. All simulations are executed on an AMD Ryzen 9 7950X3D 16-Core processor with a single NVIDIA RTX 4090 GPU and 3DGRUT [42, 67] is used for visualization.

[1] `https://sketchfab.com`

Table 1: Performance comparison across simulation steps between a basic MPM and Ours. MSE/RMSE (m) and Acc@0.01 (ratio within $0.01m$) are reported.

Method	Metric	Animatable Gaussians				SMPL-X			
		100	200	300	400	100	200	300	400
Baseline MPM	MSE ↓	0.079	0.198	0.275	0.332	0.089	0.138	0.206	0.248
	RMSE ↓	0.100	0.232	0.317	0.380	0.120	0.165	0.242	0.291
	Acc@0.01 ↑	0.097	0.041	0.020	0.020	0.358	0.181	0.086	0.036
Ours	MSE ↓	0.019	0.027	0.036	0.046	0.013	0.022	0.031	0.039
	RMSE ↓	0.023	0.034	0.043	0.056	0.016	0.025	0.034	0.044
	Acc@0.01 ↑	0.844	0.719	0.606	0.534	0.908	0.823	0.670	0.583

Table 2: Kabsch-based pose alignment errors across simulation steps for Animatable Gaussians and SMPL-X. Lower is better.

Metric	Animatable Gaussians				SMPL-X			
	100	200	300	400	100	200	300	400
Root Rot. (°) ↓	0.137	0.144	0.162	0.154	0.030	0.061	0.049	0.121
Root Trl. (m) ↓	0.0006	0.0006	0.0006	0.0007	0.0002	0.0004	0.0003	0.0010
Rel. Rot. (°) ↓	0.632	0.616	0.663	0.642	0.097	0.205	0.202	0.320
Rel. Trl. (m) ↓	0.0041	0.0038	0.0044	0.0042	0.0005	0.0011	0.0011	0.0021
Distance Error. (m) ↓	0.0057	0.0052	0.0056	0.0056	0.0022	0.0025	0.0034	0.0032

5.2 Quantitative Evaluation Results

Comparisons with Baseline MPM. We measure MSE, RMSE, and Accuracy at $\tau = 0.01m$ between our PIAvatar and the basic MPM with respect to the target avatar positions calculated from the pose sequences. To show the robustness of our method, we conduct experiments under two conditions: (1) 20-poses sequence for each of the 7 AG actors; (2) SMPL-X with random body shapes (betas), two genders, and 20 poses.

As shown in Tab. 1, our method reliably reflects pose changes to the avatars regardless of the conditions. Since we leverage Kinematic Deformation Decoupling, the user-defined velocity drives the intended motion without errors from stress. In contrast, the basic MPM suffers from the increasing deviation from the target position over time.

Pose Tracking. We evaluate the accuracy of continuous pose tracking using our skeletal model over the ground-truth pose. Specifically, we test 20 non-overlapping random poses for each of the seven AG models, and simulate a sequence of 200 random poses for the SMPL-X model.

Tab. 2 describes the rotation and translation errors of both the root and relative joints. We compute velocities directly from the target-frame pose following the Half-Physics [58] protocol, rather than relying on intermediate pose updates. As shown in the results, our gradient decoupling strategy produces only negligible differences, at the level of floating-point precision. This verifies that the proposed formulation preserves skeletal pose consistency during continuous simulation.

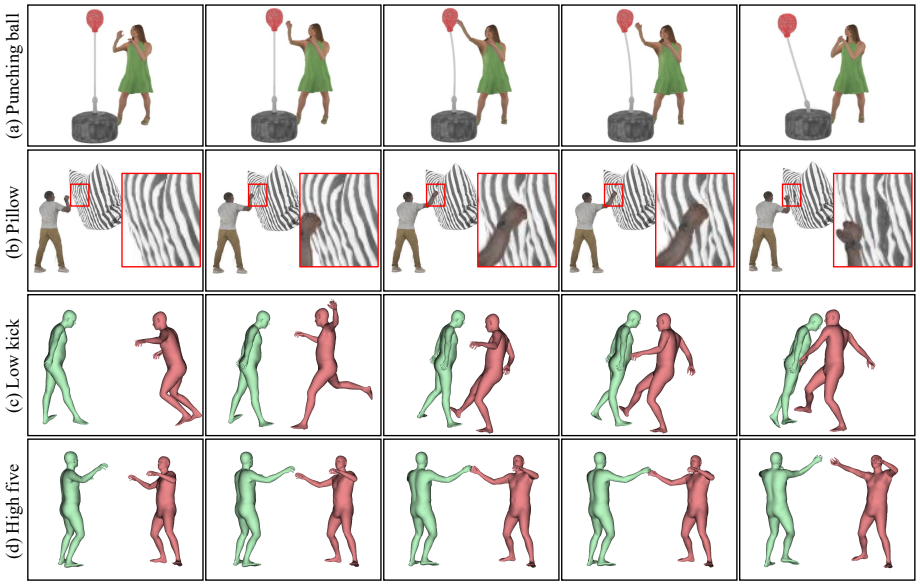

Fig. 4: Various Interactions. (a, b) Non-rigid deformations arising from physical interactions with objects. (c, d) The bidirectional interaction between avatars and its mutual pose changes.

Time and Space Complexity. The detailed runtime is described in Tab. 3. The experiments are conducted using single- and four-avatar scenarios in SMPL-X and AG. The velocity generation and the Kabsch algorithm are executed once per frame, so we divide them by the number of simulation steps per frame. The overall overhead remains modest compared to the basic MPM, as the increased time cost in the grid initialization step is inevitable. This can be further improved with known techniques, such as fixing the number of additional grids and sharing them across avatars via an atomic CAS-based allocation strategy [13]. The additional memory overhead comes from an extra 200^3 grid allocation per avatar to store kinetic momentum and their mass, which needs 122 MB for both types of avatar. This is relatively small compared to the basic MPM memory usage of approximately 3.8 GB for SMPL-X and 5.4 GB for AG.

5.3 Qualitative Evaluation Results

Our physics-based avatar simulation framework enables diverse qualitative evaluations, showcasing interaction outcomes with a variety of physical capabilities.
Human-Object Interactions. As shown in Fig. 4(a, b), the momentum naturally impacts both the avatars and the objects, leading to the temporary non-rigid deformations of the avatar or displacements of the objects through the transmitted forces. These results validate that the user-defined kinematic velocity enables physically plausible interactions with the objects.

Table 3: Runtime comparison for single- and four-avatar interaction scenarios using SMPL-X and Animatable Gaussians. All times are measured in milliseconds (ms).

Component	Single SMPL-X	Four SMPL-X	Single AG [34]	Four AG
Basic MPM	0.170	0.311	2.619	11.015
Velocity Computation	0.149	0.542	0.976	3.661
+ Grid Initialization	0.151	0.518	0.145	0.472
+ P2G	0.0068	0.0153	0.0076	0.0043
+ G2P	0.0068	0.0170	0.0085	0.0748
+ Kabsch algorithm	0.0085	0.0324	0.0082	0.0319
Total time (ms)	**0.492**	**1.436**	**3.764**	**15.259**

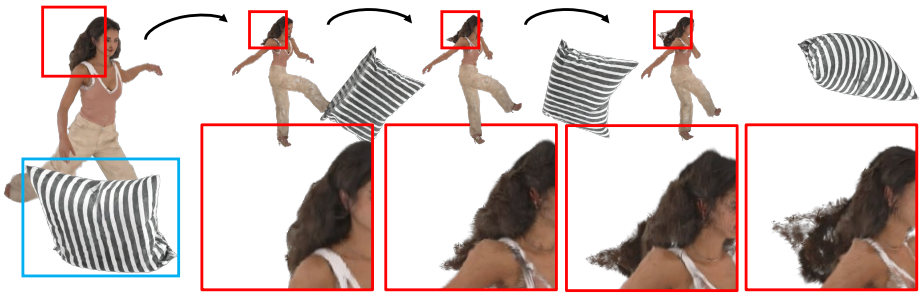

Fig. 5: Non-rigid deformation. Our method can generate non-rigid deformations, such as naturally fluttering hair, that are not achievable with conventional avatars without explicit modeling.

Human-Human Interactions. In Fig. 4(c, d), the external forces generated by one avatar can be directly transmitted to another, altering its pose. This enables physics-based interactions between avatars, including striking motions to lose balance or to be pushed away.

Non-rigid Simulation. Fig. 5 shows that the momentum transmitted from a pillow has an impact on the avatar's body and hair. These results suggest that our physics-based simulation can reproduce such secondary effects, without explicitly modeling fine-scale deformations or using any additional hair simulation model. Similarly, in Fig. 4(a, b), we have already displayed the physical momentum on non-rigid subjects.

Soft-Tissue Deformation. Fig. 6 further demonstrates that our formulation reproduces soft-tissue deformation as a byproduct. Without introducing any additional soft-body models, belly jiggling emerges naturally from particle interactions. In practice, we slightly attenuate the kinematic velocity applied to particles in the belly region, allowing stresses to develop through neighboring interactions within the same unified framework.

Deformation Gradient Visualization. We visualize how a deformation gradient of a non-rigid object changes in Fig. 7. During the interaction, starting from the hitting point of the object, the deformation gradient progressively propagates outward to the surrounding area.

Fig. 6: Soft-tissue deformable avatar. Our simulator produces non-rigid effects, such as the natural jiggling of belly fat. Unlike conventional LBS-based avatars, where body motion does not influence surface deformation, our physical simulation system exhibits natural inertial wobbling during jumping and landing.

Fig. 7: Deformation gradient visualization. The simulation visualizes how forces are transmitted through changes in the deformation gradient $\boldsymbol{F}_p$.

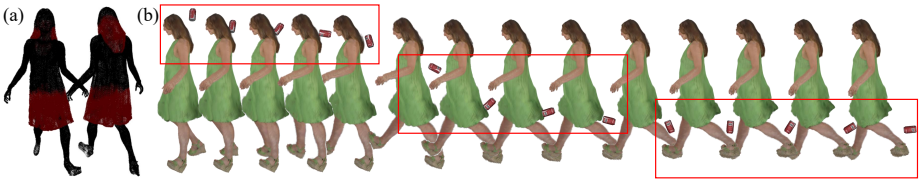

Fig. 8: Heterogeneous material assignment. (a) The body (Neo-Hookean, E=10^5) and compliant regions (corotated, E=10^2–10^3) exhibit distinct deformation magnitudes under motion. (b) Despite regional variation in stiffness, global skeletal articulation remains stable and consistent with the input pose sequence.

Heterogeneous Material Parameters. We lastly evaluate the versatility of our PIAvatar through an experiment on a heterogeneous material assignment. The body is modeled using a Neo-Hookean hyperelastic model (E=10^5), while garments and hair are assigned more compliant corotated linear elastic models (E=10^2 and 10^3), introducing relatively different stiffness according to the parts of the avatar (Fig. 8(a)). These values are calibrated as effective engineering parameters to ensure numerical stability within the MPM discretization. As shown in Fig. 8(b), we observe different collision responses when the fallen coke can hits the hair, garment, and the ankle.

5.4 VLM-Driven Physical Parameter Optimization

We introduce a potential way to automatically find physical parameters using Vision-Language Models (VLM). As shown in Fig. 9, we design an additional

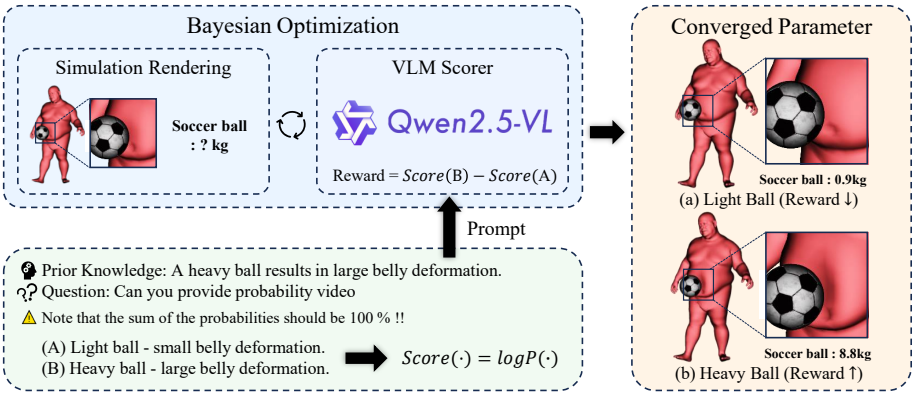

Fig. 9: Automatic physical parameter optimization. The difference between the log probabilities assigned by the VLM to the binary answer, such as *light* vs *heavy*, is used as the BO reward for automatic parameter settings.

Table 4: BO with VLM scorer converges to opposite parameter per the text target.

Axis	Converged	Reward	Converged	Reward
Mass (kg)	**8.8** *(heavy)*	+3.53	**0.9** *(light)*	−2.90
E (Pa)	$\mathbf{3.38{\times}10^{6}}$ *(stiff)*	+1.17	$\mathbf{3.26{\times}10^{2}}$ *(soft)*	−0.81

Table 5: User study. Modal selections (red/blue) match the BO in Tab. 4.

Mass (kg)	0.5	**1.0**	1.7	3.7	**8.2**	10.0
heavy / light	1 / 12	1 / **20**	4 / 7	13 / 8	**16** / 1	15 / 2
E (Pa)	10^2	$\mathbf{4.2{\cdot}10^{2}}$	$3.7{\cdot}10^3$	$1.3{\cdot}10^5$	$\mathbf{2.4{\cdot}10^{6}}$	10^7
stiff / soft	4 / 6	4 / **16**	5 / 12	4 / 9	**23** / 2	10 / 5

experiment that optimizes the ball's density d or the belly's stiffness E with both Bayesian optimization (BO) [8] and Qwen2.5-VL [3] as the VLM reward model.

Since VLMs lack direct intuition for abstract physical quantities, we prepare prompts for an observable cues, e.g., larger belly deformation for a heavier ball or smaller deformation for a stiffer belly as a prior knowledge. As illustrated in Fig. 9, at each BO iteration, the VLM is asked to choose between two contrasting prompts (*e.g.*, *heavy* vs. *light*, or *stiff* vs. *soft*) for the rendered simulation video. The difference between the log probabilities of the two answers is then used as the BO reward. We optimize physical parameters with BO, density d[2] and Young's modulus E: maximizing the reward for *heavy*/*stiff* behavior leads to the larger physical parameters, whereas minimizing the same reward leads to smaller ones corresponding to *light*/*soft* behavior. In Tab. 4, the proposed optimization framework consistently converges, validating that the VLM-based reward design successfully distinguishes heaviness/lightness as positive/negative scores, respectively.

We evaluate the proposed BO with the VLM framework through a user study with 50 participants in MTurk. We manually generate videos with six different attributes, and the participants are asked to choose most plausible videos under the prompts regarding to ball weight or belly softness. As reported in Tab. 5, the most-chosen videos closely match the BO-converged values in the red/blue cells of Tab. 4, implying that the VLM-driven proxy aligns with human perception.

[2] Object's mass(kg) is computed from the multiplication of its density d(kg/m^3) and volume(m^3).

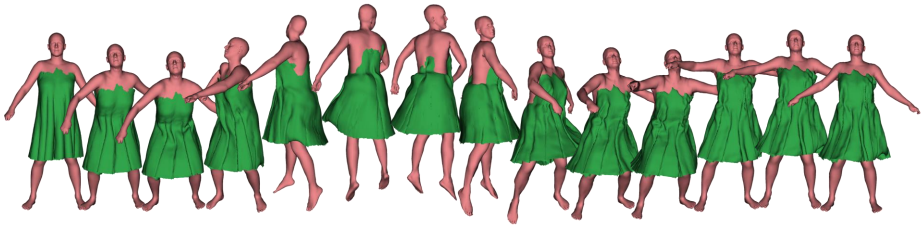

Fig. 10: Loose-garment simulation. On *spin* and *jump* from fast AMASS motion sequences, our coupled cloth solver produces visible flutter and asymmetric billowing of the garment driven by rapid body articulation.

5.5 Loose Cloth Simulation

Existing LBS-based avatars deform garments using skinning weights inherited from the underlying body, so loose clothing rigidly follows the skin and cannot exhibit secondary dynamics such as swing, lag or inertia. Leveraging the recent MPMAvatar cloth solver [32] and the extensibility of our MPM-based simulator, we apply our framework to loose-garment scenarios by coupling a thin-shell cloth solver with our MPM body simulator. For garments, we replace volumetric MPM with a separate thin-shell formulation that applies a QR decomposition to the per-element deformation gradient, decoupling in-plane membrane stretching from out-of-plane bending. As shown in Fig. 10, we show the example of loose-garment simulation that the dress flutters naturally.

6 Conclusion

We introduce PIAvatar, an MPM-based avatar simulation framework that achieves physically aware, non-rigid and deformable avatar interactions as well as their surrounding objects. By (1) decoupling the deformation gradient and (2) embedding a skeletal structure for direct pose estimation, PIAvatar enables stress-free kinematic control and demonstrates physically consistent interaction behaviors across diverse scenarios.

Limitation. To create desired physical interaction scenarios, the manual setup of avatar positions and pose sequence is required. The current pose datasets, such as AMASS [40], lack a torque control under external forces like gravity, and friction is not implemented, making posture maintenance difficult. Potential solutions include PID control or RL for the torque control, and integrating friction and object properties through large-scale physics engines like Genesis [2]. In addition, rapid collisions between two objects can also cause them to stick before stress develops. This raises the need for a new learning framework for understanding scene configurations and object characteristics.

Acknowledgement This work was supported by the National Research Foundation of Korea (NRF) grant (RS-2024-00338439), the Institute for Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government (MSIT) (No.RS-2025-25441838, Development of a human foundation model for human-centric universal artificial intelligence and training of personnel), and the Korea Institute of Science and Technology (KIST) Institutional Program (26K0030).

References

1. Anguelov, D., Srinivasan, P., Koller, D., Thrun, S., Rodgers, J., Davis, J.: Scape: shape completion and animation of people. In: ACM Transactions on Graphics (TOG). ACM (2005)
2. Authors, G.: Genesis: A generative and universal physics engine for robotics and beyond (December 2024), https://github.com/Genesis-Embodied-AI/Genesis
3. Bai, S., Chen, K., Liu, X., Wang, J., Ge, W., Song, S., Dang, K., Wang, P., Wang, S., Tang, J., Zhong, H., Zhu, Y., Yang, M., Li, Z., Wan, J., Wang, P., Ding, W., Fu, Z., Xu, Y., Ye, J., Zhang, X., Xie, T., Cheng, Z., Zhang, H., Yang, Z., Xu, H., Lin, J.: Qwen2.5-vl technical report. arXiv preprint arXiv:2502.13923 (2025)
4. Bogo, F., Kanazawa, A., Lassner, C., Gehler, P., Romero, J., Black, M.J.: Keep it SMPL: Automatic estimation of 3D human pose and shape from a single image. In: Proceedings of European Conference on Computer Vision (ECCV) (2016)
5. Clough, R.: The Finite Element Method in Plane Stress Analysis. American Society of Civil Engineers (1960), https://books.google.co.kr/books?id=rfwFHQAACAAJ
6. Corporation, N.: Nvidia physx sdk. https://developer.nvidia.com/physx-sdk (2021), accessed: March 2025
7. Coumans, E.: Bullet physics simulation. In: ACM SIGGRAPH 2015 Courses, p. 1. ACM (2015)
8. Frazier, P.I.: Bayesian optimization. In: Recent advances in optimization and modeling of contemporary problems. Informs (2018)
9. Guo, C., Jiang, T., Chen, X., Song, J., Hilliges, O.: Vid2avatar: 3d avatar reconstruction from videos in the wild via self-supervised scene decomposition. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
10. Guo, Q., Han, X., Fu, C., Gast, T., Tamstorf, R., Teran, J.: A material point method for thin shells with frictional contact. ACM Transactions on Graphics (TOG) (2018)
11. Han, S.H., Park, M.G., Yoon, J.H., Kang, J.M., Park, Y.J., Jeon, H.G.: High-fidelity 3d human digitization from single 2k resolution images. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
12. Han, X., Gast, T.F., Guo, Q., Wang, S., Jiang, C., Teran, J.: A hybrid material point method for frictional contact with diverse materials. ACM Transactions on Graphics (TOG) (2019)
13. Herlihy, M.: Wait-free synchronization. ACM Transactions on Programming Languages and Systems (TOPLAS) (1991)
14. Ho, I., Song, J., Hilliges, O., et al.: Sith: Single-view textured human reconstruction with image-conditioned diffusion. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2024)
15. Hu, H., Fan, Z., Wu, T., Xi, Y., Lee, S., Pavlakos, G., Wang, Z., et al.: Expressive gaussian human avatars from monocular rgb video. Proceedings of the Neural Information Processing Systems (NeurIPS) (2024)
16. Hu, L., Zhang, H., Zhang, Y., Zhou, B., Liu, B., Zhang, S., Nie, L.: Gaussianavatar: Towards realistic human avatar modeling from a single video via animatable 3d gaussians. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2024)
17. Hu, Y., Fang, Y., Ge, Z., Qu, Z., Zhu, Y., Pradhana, A., Jiang, C.: A moving least squares material point method with displacement discontinuity and two-way rigid body coupling. ACM Transactions on Graphics (TOG) (2018)

18. Işık, M., Rünz, M., Georgopoulos, M., Khakhulin, T., Starck, J., Agapito, L., Nießner, M.: Humanrf: High-fidelity neural radiance fields for humans in motion. ACM Transactions on Graphics (TOG) (2023). https://doi.org/10.1145/3592415, https://doi.org/10.1145/3592415
19. Jiang, C., Gast, T., Teran, J.: Anisotropic elastoplasticity for cloth, knit and hair frictional contact. ACM Transactions on Graphics (TOG) (2017)
20. Jiang, C., Schroeder, C., Selle, A., Teran, J., Stomakhin, A.: The affine particle-in-cell method. ACM Transactions on Graphics (TOG) (2015)
21. Jiang, C., Schroeder, C., Teran, J.: An angular momentum conserving affine-particle-in-cell method. Journal of Computational Physics (JCP) (2017)
22. Jiang, C., Schroeder, C., Teran, J., Stomakhin, A., Selle, A.: The material point method for simulating continuum materials. In: Acm siggraph 2016 courses. ACM Transactions on Graphics (TOG) (2016)
23. Jiang, T., Chen, X., Song, J., Hilliges, O.: Instantavatar: Learning avatars from monocular video in 60 seconds. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
24. Kabsch, W.: A solution for the best rotation to relate two sets of vectors. Foundations of Crystallography (1976)
25. Kanazawa, A., Black, M.J., Jacobs, D.W., Malik, J.: End-to-end recovery of human shape and pose. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2018)
26. Keller, M., Werling, K., Shin, S., Delp, S., Pujades, S., Liu, C.K., Black, M.J.: From skin to skeleton: Towards biomechanically accurate 3d digital humans. ACM Transactions on Graphics (TOG) (2023)
27. Keller, M., Zuffi, S., Black, M.J., Pujades, S.: Osso: Obtaining skeletal shape from outside. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2022)
28. Kerbl, B., Kopanas, G., Leimkühler, T., Drettakis, G.: 3d gaussian splatting for real-time radiance field rendering. ACM Transactions on Graphics (TOG) (2023)
29. Kim, M., Pons-Moll, G., Pujades, S., Bang, S., Kim, J., Black, M.J., Lee, S.H.: Data-driven physics for human soft tissue animation. ACM Transactions on Graphics (TOG) (2017)
30. Klár, G., Gast, T., Pradhana, A., Fu, C., Schroeder, C., Jiang, C., Teran, J.: Drucker-prager elastoplasticity for sand animation. ACM Transactions on Graphics (TOG) (2016)
31. Kocabas, M., Athanasiou, N., Black, M.J.: Vibe: Video inference for human body pose and shape estimation. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
32. Lee, C., Lee, J., Kim, T.K.: Mpmavatar: Learning 3d gaussian avatars with accurate and robust physics-based dynamics. In: Proceedings of the Neural Information Processing Systems (NeurIPS) (2025)
33. Li, M., Kaufman, D.M., Jiang, C.: Codimensional incremental potential contact. arXiv preprint arXiv:2012.04457 (2020)
34. Li, Z., Zheng, Z., Wang, L., Liu, Y.: Animatable gaussians: Learning pose-dependent gaussian maps for high-fidelity human avatar modeling. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2024)
35. Liu, L., Yin, K., Wang, B., Guo, B.: Simulation and control of skeleton-driven soft body characters. ACM Transactions on Graphics (TOG) (2013)
36. Loper, M., Mahmood, N., Black, M.J.: Mosh: motion and shape capture from sparse markers. ACM Transactions on Graphics (TOG) (2014)

37. Loper, M., Mahmood, N., Romero, J., Pons-Moll, G., Black, M.J.: Smpl: a skinned multi-person linear model. ACM Transactions on Graphics (TOG) (2015)
38. Luo, Z., Cao, J., Kitani, K., Xu, W., et al.: Perpetual humanoid control for real-time simulated avatars. In: Proceedings of International Conference on Computer Vision (ICCV) (2023)
39. Luo, Z., Cao, J., Merel, J., Winkler, A., Huang, J., Kitani, K., Xu, W.: Universal humanoid motion representations for physics-based control. International Conference on Learning Representations (ICLR) (2023)
40. Mahmood, N., Ghorbani, N., Troje, N.F., Pons-Moll, G., Black, M.J.: Amass: Archive of motion capture as surface shapes. In: Proceedings of International Conference on Computer Vision (ICCV) (2019)
41. maximeraafat: Blendernerf. `https://github.com/maximeraafat/BlenderNeRF` (2023)
42. Moenne-Loccoz, N., Mirzaei, A., Perel, O., de Lutio, R., Esturo, J.M., State, G., Fidler, S., Sharp, N., Gojcic, Z.: 3d gaussian ray tracing: Fast tracing of particle scenes. ACM Transactions on Graphics (TOG) (2024)
43. Moon, G., Shiratori, T., Saito, S.: Expressive whole-body 3d gaussian avatar. In: Proceedings of European Conference on Computer Vision (ECCV) (2024)
44. Pan, P., Su, Z., Lin, C., Fan, Z., Zhang, Y., Li, Z., Shen, T., Mu, Y., Liu, Y.: Humansplat: Generalizable single-image human gaussian splatting with structure priors. Proceedings of the Neural Information Processing Systems (NeurIPS) (2024)
45. Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A.A.A., Tzionas, D., Black, M.J.: Expressive body capture: 3D hands, face, and body from a single image. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
46. Pavlakos, G., Choutas, V., Ghorbani, N., Bolkart, T., Osman, A.A., Tzionas, D., Black, M.J.: Expressive body capture: 3d hands, face, and body from a single image. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2019)
47. Peng, S., Dong, J., Wang, Q., Zhang, S., Shuai, Q., Zhou, X., Bao, H.: Animatable neural radiance fields for modeling dynamic human bodies. In: Proceedings of International Conference on Computer Vision (ICCV) (2021)
48. Peng, S., Zhang, Y., Xu, Y., Wang, Q., Shuai, Q., Bao, H., Zhou, X.: Neural body: Implicit neural representations with structured latent codes for novel view synthesis of dynamic humans. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2021)
49. Peng, X.B., Abbeel, P., Levine, S., van de Panne, M.: Deepmimic: Example-guided deep reinforcement learning of physics-based character skills. ACM Transactions on Graphics (TOG) (2018)
50. Peng, X.B., Ma, Z., Abbeel, P., Levine, S., Kanazawa, A.: Amp: Adversarial motion priors for stylized physics-based character control. ACM Transactions on Graphics (TOG) (2021)
51. Qiu, L., Gu, X., Li, P., Zuo, Q., Shen, W., Zhang, J., Qiu, K., Yuan, W., Chen, G., Dong, Z., et al.: Lhm: Large animatable human reconstruction model from a single image in seconds. Proceedings of International Conference on Computer Vision (ICCV) (2025)
52. Qiu, L., Zhu, S., Zuo, Q., Gu, X., Dong, Y., Zhang, J., Xu, C., Li, Z., Yuan, W., Bo, L., et al.: Anigs: Animatable gaussian avatar from a single image with inconsistent gaussian reconstruction. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2025)

53. Saito, S., Huang, Z., Natsume, R., Morishima, S., Kanazawa, A., Li, H.: Pifu: Pixel-aligned implicit function for high-resolution clothed human digitization. In: Proceedings of International Conference on Computer Vision (ICCV) (2019)
54. Saito, S., Simon, T., Saragih, J., Joo, H.: Pifuhd: Multi-level pixel-aligned implicit function for high-resolution 3d human digitization. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
55. Shao, Z., Wang, Z., Li, Z., Wang, D., Lin, X., Zhang, Y., Fan, M., Wang, Z.: Splattingavatar: Realistic real-time human avatars with mesh-embedded gaussian splatting. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2024)
56. Shin, J., Lee, J., Lee, S., Park, M.G., Kang, J.M., Yoon, J.H., Jeon, H.G.: Canonicalfusion: Generating drivable 3d human avatars from multiple images. In: Proceedings of European Conference on Computer Vision (ECCV) (2024)
57. Sim, G., Moon, G.: Persona: Personalized whole-body 3d avatar with pose-driven deformations from a single image. In: Proceedings of International Conference on Computer Vision (ICCV) (2025)
58. Siyao, L., Feng, Y., Taheri, O., Loy, C.C., Black, M.J.: Half-physics: Enabling kinematic 3d human model with physical interactions. arXiv preprint arXiv:2507.23778 (2025)
59. Stomakhin, A., Schroeder, C., Chai, L., Teran, J., Selle, A.: A material point method for snow simulation. ACM Transactions on Graphics (TOG) (2013)
60. Su, S.Y., Yu, F., Zollhöfer, M., Rhodin, H.: A-nerf: Articulated neural radiance fields for learning human shape, appearance, and pose. Proceedings of the Neural Information Processing Systems (NeurIPS) (2021)
61. Tevet, G., Raab, S., Cohan, S., Reda, D., Luo, Z., Peng, X.B., Bermano, A.H., van de Panne, M.: Closd: Closing the loop between simulation and diffusion for multi-task character control. International Conference on Learning Representations (ICLR) (2025)
62. Todorov, E., Erez, T., Tassa, Y.: Mujoco: A physics engine for model-based control. In: Proceedings of International Conference on Intelligent Robots and Systems (IROS). IEEE (2012)
63. Wang, R., Prada, F., Wang, Z., Jiang, Z., Yin, C., Li, J., Saito, S., Santesteban, I., Romero, J., Joshi, R., et al.: Fresa: Feedforward reconstruction of personalized skinned avatars from few images. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2025)
64. Wen, J., Zhao, X., Ren, Z., Schwing, A.G., Wang, S.: Gomavatar: Efficient animatable human modeling from monocular video using gaussians-on-mesh. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2024)
65. Weng, C.Y., Curless, B., Srinivasan, P.P., Barron, J.T., Kemelmacher-Shlizerman, I.: Humannerf: Free-viewpoint rendering of moving people from monocular video. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2022)
66. Wu, G., Yi, T., Fang, J., Xie, L., Zhang, X., Wei, W., Liu, W., Tian, Q., Wang, X.: 4d gaussian splatting for real-time dynamic scene rendering. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2024)
67. Wu, Q., Martinez Esturo, J., Mirzaei, A., Moenne-Loccoz, N., Gojcic, Z.: 3dgut: Enabling distorted cameras and secondary rays in gaussian splatting. Conference on Computer Vision and Pattern Recognition (CVPR) (2025)

68. Xie, T., Zong, Z., Qiu, Y., Li, X., Feng, Y., Yang, Y., Jiang, C.: Physgaussian: Physics-integrated 3d gaussians for generative dynamics. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2024)
69. Xiu, Y., Yang, J., Cao, X., Tzionas, D., Black, M.J.: Econ: Explicit clothed humans optimized via normal integration. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2023)
70. Xiu, Y., Yang, J., Tzionas, D., Black, M.J.: Icon: Implicit clothed humans obtained from normals. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2022)
71. Xu, H., Bazavan, E.G., Zanfir, A., Freeman, W.T., Sukthankar, R., Sminchisescu, C.: Ghum & ghuml: Generative 3d human shape and articulated pose models. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2020)
72. Xu, S., Ling, H.Y., Wang, Y.X., Gui, L.Y.: Intermimic: Towards universal whole-body control for physics-based human-object interactions. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2025)
73. Xue, Y., Xie, X., Marin, R., Pons-Moll, G.: Human 3diffusion: Realistic avatar creation via explicit 3d consistent diffusion models. Proceedings of the Neural Information Processing Systems (NeurIPS) (2024)
74. Zheng, Y., Zhao, Q., Yang, G., Yifan, W., Xiang, D., Dubost, F., Lagun, D., Beeler, T., Tombari, F., Guibas, L., et al.: Physavatar: Learning the physics of dressed 3d avatars from visual observations. In: Proceedings of European Conference on Computer Vision (ECCV) (2024)
75. Zheng, Z., Yu, T., Liu, Y., Dai, Q.: Pamir: Parametric model-conditioned implicit representation for image-based human reconstruction. IEEE Transactions on Pattern Analysis and Machine Intelligence (PAMI) (2021)
76. Zhuang, Y., Lv, J., Wen, H., Shuai, Q., Zeng, A., Zhu, H., Chen, S., Yang, Y., Cao, X., Liu, W.: Idol: Instant photorealistic 3d human creation from a single image. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2025)