M-PhyGs: Multi-Material Object Dynamics from Video

Norika Wada Kohei Yamashita Ryo Kawahara Ko Nishino

https://vision.ist.i.kyoto-u.ac.jp/research/m-phygs/

{nwada,kyamashita,ryo}@vision.ist.i.kyoto-u.ac.jp, kon@i.kyoto-u.ac.jp

Graduate School of Informatics, Kyoto University, Kyoto, Japan

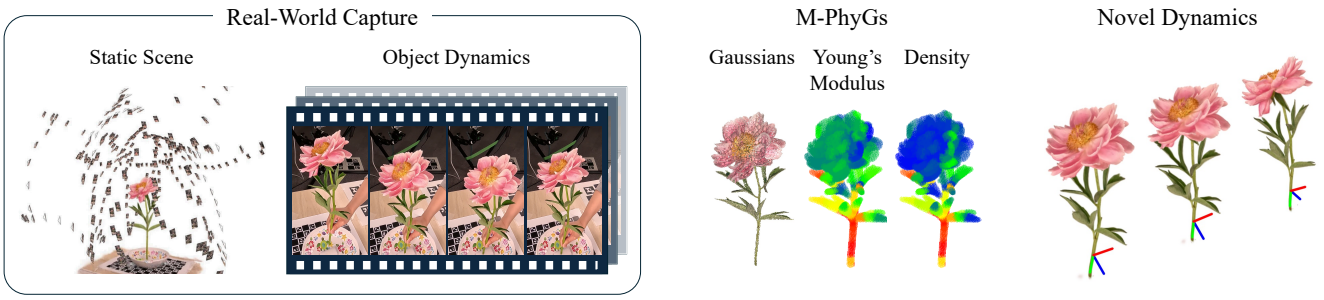


Figure 1. Multi-material Physical Gaussians (M-PhyGs) recovers the physical material properties including the Young's modulus of real-world objects consisting of multiple parts of different materials from short videos captured from a sparse set of views. The recovered material parameters can be used to predict how the object responds to unseen physical interactions.

Abstract

Knowledge of the physical material properties governing the dynamics of a real-world object becomes necessary to accurately anticipate its response to unseen interactions. Existing methods for estimating such physical material parameters from visual data assume homogeneous single-material objects, pre-learned dynamics, or simplistic topologies. Real-world objects, however, are often complex in material composition and geometry lying outside the realm of these assumptions. In this paper, we particularly focus on flowers as a representative common object. We introduce Multi-material Physical Gaussians (M-PhyGs) to estimate the material composition and parameters of such multi-material complex natural objects from video. From a short video captured in a natural setting, M-PhyGs jointly segments the object into similar materials and recovers their continuum mechanical parameters while accounting for gravity. M-PhyGs achieves this efficiently with newly introduced cascaded 3D and 2D losses, and by leveraging temporal mini-batching. We introduce a dataset, Phlowers, of people interacting with flowers as a novel platform to evaluate the accuracy of this challenging task of multi-material physical parameter estimation. Experimental results on Phlowers dataset demonstrate the accuracy and effectiveness of M-PhyGs and its components.

1. Introduction

Anticipating the dynamic behavior of an object for arbitrary interactions from minimal visual observations can serve an essential role in vision and robotics. Being able to predict, just from a simple visual setup, how an object would behave in response to forces induced through interaction with a human or a robot can enable accurate planning of how to handle the object. In addition to estimating the physical properties for Newton dynamics to capture rigid-body motions, modeling and recovering the physical properties that dictate the dynamics of deformable objects becomes essential. The real world is filled with soft-material objects that can non-rigidly change their shapes as they are simply picked up, carried, and put down.

Past methods for estimating the underlying parameters or directly learning the dynamics of deformable objects take three distinct approaches. The first assumes single material objects [1, 2, 13, 17], *i.e.*, only recover one set of physical material parameters of a dynamics model (*e.g.*, Material Point Method [4, 6, 28]) for the whole object. The second pre-learns the dynamics itself (typically with video

diffusion) and estimates the physical material parameters with its supervision [5, 15, 16, 33], assigns them directly with LLMs [14, 34], or trains neural networks on datasets annotated by LLMs [11]. The third approach models an object as a spring-mass system [7, 35] or a graph neural network [26, 32] of particles (typically 3D Gaussians for Gaussian splatting [8]). This inevitably assumes simple topology which can be far from the true internal mechanical topology of the object, which consequently also necessitates diverse motion observations to learn its parameter values.

Natural objects are often made of multiple parts, with distinct boundaries, each made of different materials that exhibit different mechanics seamlessly interacting with each other, leading to complex dynamics as a whole. Think of trees outside the window, and flowers in your garden. They are all composed of different materials such as stems, leaves, and petals. They also have a complex geometry that is difficult to model with simple topology. In this paper, we focus on flowers as daily representative multi-material objects that pose severe challenges to current approaches as their complex material composition fundamentally breaks underlying assumptions, which we also experimentally demonstrate.

A number of challenges underlie the modeling of dynamics of multi-material objects. First is the segmentation and estimation of different material segments and parts of the object. Accurate material-wise segmentation cannot be achieved solely from static observations and the physical parameters of each segment cannot be supplied by a pre-learned model, since even objects within the same category have unique material compositions and properties. The interaction of the distinct materials across different parts also add to their rich dynamics. This necessitates an analysis by synthesis approach that combines observations and physics simulation, which gives rise to a slew of difficulties. The 3D geometry of the object can only be recovered in detail with a dense set of views, which is prohibitive for capturing its dynamics. The object, whether in motion or in steady state, is always in equilibrium with gravity, so gravity needs to be properly accounted for. A sizable sequence length of observations of the object dynamics becomes essential for accurate parameter recovery, which causes unstable estimation due to large discrepancies between the predicted and observed as dynamics simulation is inherently sequential. Computational cost also becomes a major obstacle.

We introduce Multi-material Physics Gaussians (M-PhyGs /ɛm-figz/), a novel multi-material estimation method that overcomes all these challenges. M-PhyGs represents the target object with a hybrid representation consisting of 3D Gaussians recovered via 3D Gaussian splatting from a dense view capture of the object in rest state and also dense particles in a regular grid that drive these 3D Gaussians. Physical material properties are assigned to each 3D Gaussian from neighboring grid particles, which are estimated from the dynamics of the object observed from a sparse set of views. M-PhyGs makes four key contributions to enable this multi-material dynamics modeling: 1) this hybrid represention for dynamics and appearance; 2) joint segmentation and material parameter estimation; 3) inclusion of gravity in the dynamics modeling and material estimation; 4) novel 3D and 2D supervisions and temporal mini-batching for robust and efficient estimation.

We introduce a first-of-its-kind dataset of human-flower interactions for rigorous comparative analysis of the effectiveness of M-PhyGs. The dataset, which we refer to as Phlowers, captures a person arranging a flower with a sparse set of cameras whose rest shape is densely captured for 3D Gaussian splatting. We conduct extensive experiments on this dataset and evaluate the prediction accuracy for unseen frames. The results clearly show that M-PhyGs achieves state-of-the-art accuracy on this challenging task of multi-material object dynamics modeling. The code and data are publicly available on our project page.

2. Related Work

A range of methods has been proposed for modeling the dynamics of deformable objects. Table 1 summarizes these methods with respect to key characteristics.

Single Material Objects A variety of methods have been introduced for 3D reconstruction of dynamic scenes [12, 18, 29–31]. For better scene understanding and accurate dynamics representations, those that estimate dynamics by leveraging physical priors have attracted attention.

A key approach to this is analysis-by-synthesis, namely estimation of physical properties by minimizing the discrepancy between the results of forward physics simulation and observations [1, 2, 13, 17]. Differentiable simulation (*e.g.*, Material Point Method [4, 6, 28]) can be integrated with differentiable photorealistic object representations (*e.g.*, Neural Radiance Fields (NeRFs) [20] and 3D Gaussian Splatting [8]) to exploit the rendering loss for this minimization.

Feed-forward estimation has also been explored, in which a pre-trained feed-forward network directly estimates physical properties from visual observations (often a single image). Chen *et al*. [3] fine-tune a large video vision transformer to infer the physical properties of an object from its video. Lv *et al*. [19] train a U-Net to predict a probability distribution over the physical properties and the 3D Gaussian splatting parameters for a scene.

These methods, however, assume a single material for the entire object, and cannot be applied to complex real-world objects composed of more than one material.

Learned Materials Several methods leverage pre-learned video diffusion models or large language models (LLMs) to model the dynamics of objects composed of multiple

Table 1. M-PhyGs is the first method to achieve continuum mechanical material parameter estimation for complex, multi-material objects from real videos. Past methods either specialize in synthetic objects or dynamics (not "Real Dynamics"), single-material objects (not "Multi-material"), can only handle simple shapes (not "Complex Geometry"), or require a large amount of training data (Not "Data Efficient").

	Homogeneous [13], [1], [2], [17]	Diffusion Models [33], [5], [16], [15]	LLMs [34], [14]	Spring-Mass Model [35], [7]	Feed-Forward [3], [11]	GNNs [32], [26]	M-PhyGs (Ours)
Real Dynamics	✓			✓		✓	✓
Multi-material		✓	✓	✓	✓	✓	✓
Complex Geometry	✓	✓	✓		✓		✓
Data Efficient	✓	✓	✓	✓	✓		✓

materials. Zhang *et al*. [33] estimate the physical material properties of an object from a video of its dynamics synthesized by a pre-learned video diffusion model. Follow-up works [5, 15, 16] employ Score Distillation Sampling (SDS) [21] as supervision. Video diffusion models can generate realistic videos, but these videos are not based on laws of physics and cannot represent differences in the dynamics of different objects and compositions. Note that even for the same category, a different object instance would have a different composition of materials and thus dynamics (consider the flowers of fig and carnation).

Another group of methods assign physical parameters to each material segment based on their semantic descriptions by using LLMs [14, 34]. These methods can only identify physical material parameters at the scale of object categories, limiting their abilities to represent the vast variations among instances within the same category. These methods also suffer from the inherent ambiguity of identifying homogeneous material segments of an object from static data. Feed-forward inference of material properties for each object segment has also been explored. Le *et al*. [11] train a 3D U-Net to predict a material field from the CLIP feature [22] of each voxel.

Again, material-wise segmentation from static visual data is inherently ambiguous (*i.e*., it is near impossible to tell how an object would move without seeing it move). Pre-learned dynamics are also bounded by the observations in the training data which do not reflect those at inference time as real-world objects exhibit diverse compositions. Even if the overall composition can be similar, unless they are exactly the same (at which point there is no point of estimation), a subtle difference (*e.g*., different size of one segment) can lead to dramatic differences in the overall dynamics.

Graph-Structured Modeling Several methods make assumptions on the mechanical structure of the object for their forward dynamics simulation. Approaches based on a spring-mass system [7, 35] or graph neural networks [26, 32] assume neighborhood connectivity of particles representing the geometry and material, which usually does not reflect the actual mechanical structure of complex multi-material real-world objects. These methods also require many training sequences (*i.e*., videos capturing dozens of distinct motion types) since the physical constraints only manifest indirectly.

Effect of Gravity Existing methods conduct simulations either in a zero-gravity environment [5, 11, 17, 33] or adds a constant external force at every object point even though the observation already includes gravitation [1–3, 13, 15, 16]. The objects we observe or capture, however, are always deformed under the influence of gravity which is already in effect in the observations and the way it manifests depends on the pose of the object. Accounting for gravity already imposed in the observation is thus of particular importance.

3. M-PhyGs

We introduce M-PhyGs, a novel method for estimating the the material properties of multi-material objects from visual observations. Figure 2 shows an overview of our method.

3.1. Photorealistic Dynamics Representation

In order to recover the mechanical material parameters of real-world object, we need a representation of the object that can accurately describe its intricate 3D geometry, radiometric appearance, and mechanical dynamics. For this, we adopt 3D Gaussian splatting for the first two and represent their motions due to external and internal forces with 3D particles surrounding them. By simulating the movements of these particles and driving the 3D Gaussians, we can optimize the underlying material parameters so that visual observations captured in the video can be explained.

The 3D Gaussians can be recovered from a dense-view capture of the object after it settles in rest shape. This can be naturally achieved by scanning an object after (or before) dynamic interaction with it, such as after putting down a deformable object on the desk. The dynamics can in turn be observed from a handful of cameras before (or after) that dense capture, for instance when the object is moved around in a person's hand. As such, both the 3D Gaussians, 3D particles, and their dynamics observation can be extracted from a single sequence of human-object interaction.

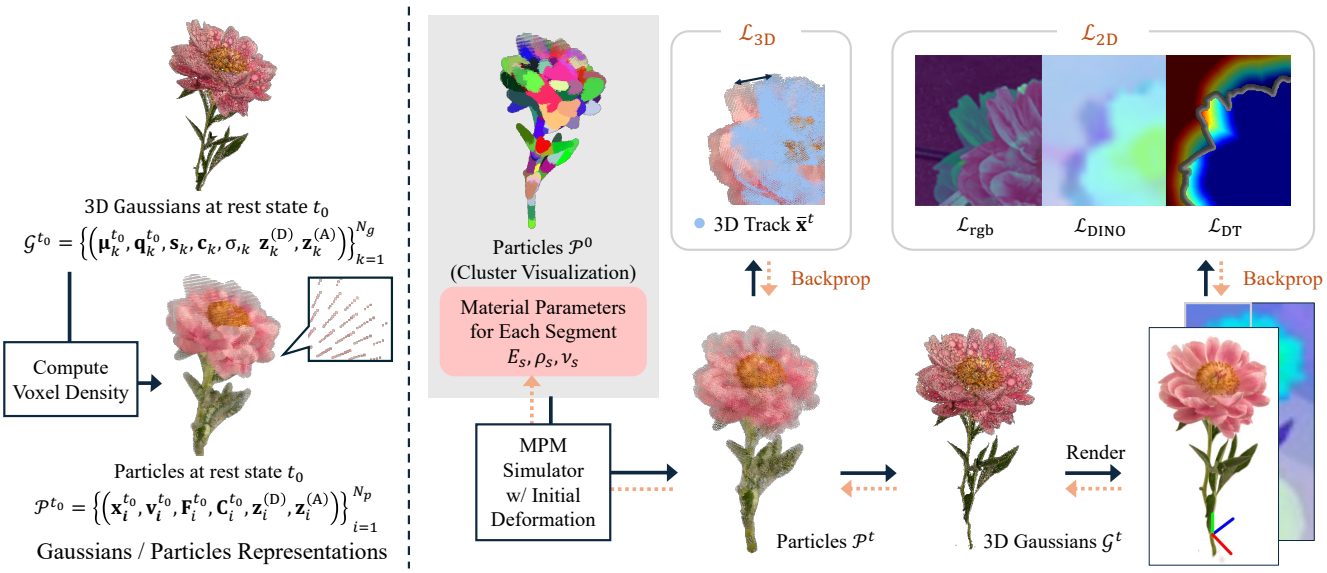


Figure 2. Overview of M-PhyGs. From dense multi-view images of a multi-material deformable object in a static state, we first recover a set of 3D Gaussians and uniformly distribute 3D particles inside the object. From a short video capturing physical interactions with the object captured from a sparse set of views, M-PhyGs estimates the physical material parameters (Young's modulus and density) of these particles which drive the 3D Gaussians. This estimation is achieved by minimization of discrepancies between the predicted and observed dynamics first in 3D geometry by assuming local rigidity and then in the 2D image plane with full non-rigid dynamics.

Let us denote the set of 3D Gaussians recovered from the rest shape dense capture with $\mathcal{G}$. When the object is moving in the video, these Gaussians can be indexed with time t

$$\mathcal{G}^t = \left\{\left(\boldsymbol{\mu}_k^t, \mathbf{q}_k^t, \mathbf{s}_k, \mathbf{c}_k, \sigma_k, \mathbf{z}_k^{(\mathrm{D})}, \mathbf{z}_k^{(\mathrm{A})}\right)\right\}_{k=1}^{N_g}, \quad (1)$$

where $\boldsymbol{\mu}_k^t$ and $\mathbf{q}_k^t$ are the position and rotation (quaternion) of the k-th Gaussian at time t, respectively. N_g is the number of Gaussians. $\mathbf{s}_k$, $\mathbf{c}_k$, and σ_k are the 3D scale, RGB color, and opacity parameters, respectively, which we model as being time-invariant (*i.e.*, fixed-sized Gaussians and Lambertian surfaces). We use the frame number for t. During the reconstruction of 3D Gaussians, we also recover a DINO [27] feature vector $\mathbf{z}_k^{(\mathrm{D})}$ and affinity feature vector $\mathbf{z}_k^{(\mathrm{A})}$ for each Gaussian [9, 10]. These features are optimized with 2D feature maps extracted from the multi-view images similar to the optimization of Gaussian colors $\mathbf{c}_k$ with RGB images.

3D Gaussian splatting optimizes the Gaussians to represent the outer surface of an object and is not suitable for representing the dynamics of the object. We inject a set of 3D particles $\mathcal{P}$ in the volume subtended by the 3D Gaussians, simulate the dynamics of these particles, and move the Gaussians based on them. Particle states at time t are

$$\mathcal{P}^t = \left\{\left(\mathbf{x}_i^t, \mathbf{v}_i^t, \mathbf{F}_i^t, \mathbf{C}_i^t, \mathbf{z}_i^{(\mathrm{D})}, \mathbf{z}_i^{(\mathrm{A})}\right)\right\}_{i=1}^{N_p}, \quad (2)$$

where $\mathbf{x}_i^t$, $\mathbf{v}_i^t$, $\mathbf{F}_i^t \equiv \frac{\partial \mathbf{x}_i^t}{\partial \mathbf{x}_i^0}$ and $\mathbf{C}_i^t \equiv \frac{\partial \mathbf{v}_i^t}{\partial \mathbf{x}_i^t}$ are the 3D location, velocity, deformation gradient, and affine velocity of the i-th particle at time t, respectively, and N_p is the number of particles. $\mathbf{z}_i^{(\mathrm{D})}$ and $\mathbf{z}_i^{(\mathrm{A})}$ are the DINO feature and affinity feature, respectively, for each particle which are assigned from the nearest Gaussians of the rest shape.

Particles at rest state are uniformly distributed in the object volume, which is defined by the voxel density derived from the distribution of 3D Gaussians. Since the voxel density is continuous, its boundary is not apparent. We first sample the point cloud of the volume defined by a loose threshold on the density values, and optimize an additional parameter that controls how far outside points from the boundary are included in the simulation.

Material parameters, namely density ρ_i, Young's modulus E_i, and Poisson's ratio ν_i are also assigned to the particles. We assume constant Poisson's ratio ν_i as its possible range is small, and estimate E_i and ρ_i through the minimization. M-PhyGs simulates the motion of particles with material parameters of each particle and then drives nearest neighboring Gaussians accordingly.

3.2. Forward Dynamics with Gravity

M-PhyGs simulates the dynamics of the object with the estimated material parameters with a continuum mechanics simulator based on the Moving Least Squares Material Point Method (MLS-MPM) [4]. MLS-MPM takes physical states of particles at a given time as inputs, and outputs those of the next timestep based on the physical material parameters of each particle. Interaction with objects external to the target

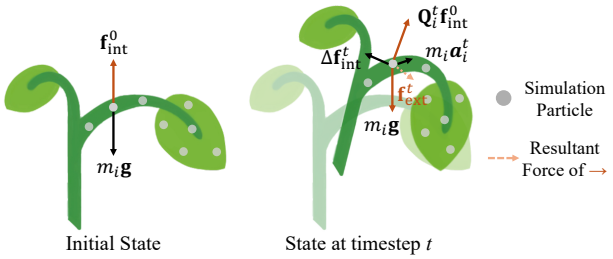


Figure 3. M-PhyGs accounts for gravity in the forward dynamics computation by adding a rotated initial internal force to counter the gravitational force at rest shape.

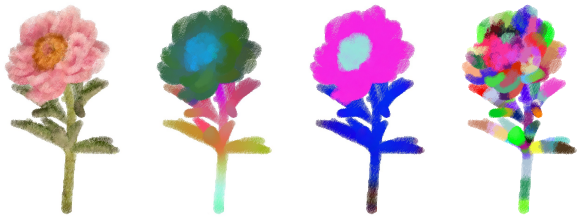


Figure 4. M-PhyGs leverages DINO [27] and GARField (affinity) [10] features assigned to each 3D particle for initial fine-grained material segmentation. It estimates per-segment physical material parameters and encourages further merging of the segments with a material grouping loss.

object (*e.g.*, a human hand) can also be simulated by adding boundary conditions on particle velocities. The motion of the contact point and the initial position of particles are estimated from the video. Please see the supplementary material for details.

All objects in the world, including the captured one, are under the influence of gravity. Surprisingly, past methods ignore this fact and simulate the dynamics under zero gravity [5, 11, 17, 33] or additive constant gravity without considering the gravity-induced deformation of an object already included in its initial state (*i.e.*, initial deformation gradient is set to identity) [1–3, 13, 15, 16].

Properly accounting for gravity when estimating materials is essential for the subsequent dynamics simulation. For this, effects of gravity on the rest shape needs to be estimated. The deformation gradient, however, expressed as a 3×3 matrix, must satisfy several constraints (*e.g.*, non-degenerate and positive definite), which makes this estimation challenging.

M-PhyGs instead estimates initial internal force per particle $\mathbf{f}_{\text{int}}$ and accommodates it in the subsequent dynamics simulation as an external force. As depicted in Fig. 3, in the initial state, assuming that a particle acceleration is sufficiently small (*i.e.*, it is in rest shape), we can compute $\mathbf{f}^0_{\text{int}}$ from the gravitational acceleration $\mathbf{g}$

$$\mathbf{f}^0_{\text{int}} = -m_i\mathbf{g}\,, \tag{3}$$

where m_i is the particle mass. In subsequent MPM simulation, the effect of gravity and initial deformation can be approximated by adding

$$\mathbf{f}_{\text{ext}} = m_i\mathbf{g} + \mathbf{Q}^t_i\mathbf{f}^0_{\text{int}}\,, \tag{4}$$

where $\mathbf{Q}^t_i$ is a rotation matrix that represents the relative rotation between timestep 0 and t, which is computed by singular value decomposition of the deformation gradient $\mathbf{F}^t_i$. As a result,

$$\mathbf{f}_{\text{ext}} + m_i\mathbf{a}^t_i + \Delta\mathbf{f}^t_{\text{int}} = 0 \tag{5}$$

holds true at any given time, where $\mathbf{a}^t_i$ and $\Delta\mathbf{f}^t_{\text{int}}$ denote the acceleration and the change of internal force from the initial state of i-th particle at time t, respectively.

3.3. Multi-Material Estimation

The large number of free parameters makes per-particle material estimation extremely challenging.

Joint Segmentation We resolve this by jointly segmenting the object into segments of homogeneous material and by estimating the material parameters for each. M-PhyGs achieves this by leveraging the visual features [10, 27] recovered for both the Gaussians and physical particles, since parts with similar appearance within the object tend to have similar physical properties. Figure 4 shows how the Gaussians and the physical particles are segmented from these features. Hyperparameters of the segmentation algorithm are adjusted to oversegment for subsequent grouping of the materials.

M-PhyGs estimates the material parameters, E_s and ρ_s, of each material segment s. M-PhyGs further consolidates the material segments with a material similarity loss based on the DINO features [27]:

$$\mathcal{L}_{\text{s}} = \frac{1}{N_p}\sum_i \sum_{j\in\mathcal{N}_i} \sum_{a\in\{\rho,E\}} w_{ij}\left\|\log\left(a_i\right) - \log\left(a_j\right)\right\|^2\,, \tag{6}$$

where ρ_i and E_i here are the density and Young's modulus of the i-th particle's segment, respectively, and $\mathcal{N}_i$ is a set of indices of neighboring particles of the i-th particle in the DINO feature space. We use 20 neighbors in our experiment. The weight w_{ij} is

$$w_{ij} = \frac{\exp\left(-\alpha\left\|\mathbf{z}^{(\text{D})}_i - \mathbf{z}^{(\text{D})}_j\right\|^2_2\right) + \varepsilon}{\sum_{j\in\mathcal{N}_i}\left(\exp\left(-\alpha\left\|\mathbf{z}^{(\text{D})}_i - \mathbf{z}^{(\text{D})}_j\right\|^2_2\right) + \varepsilon\right)}\,. \tag{7}$$

In practice, we set α to 20 and ε to 1×10^{-9}.

We also discourage the range of per-segment physical material parameter values from becoming too large with a

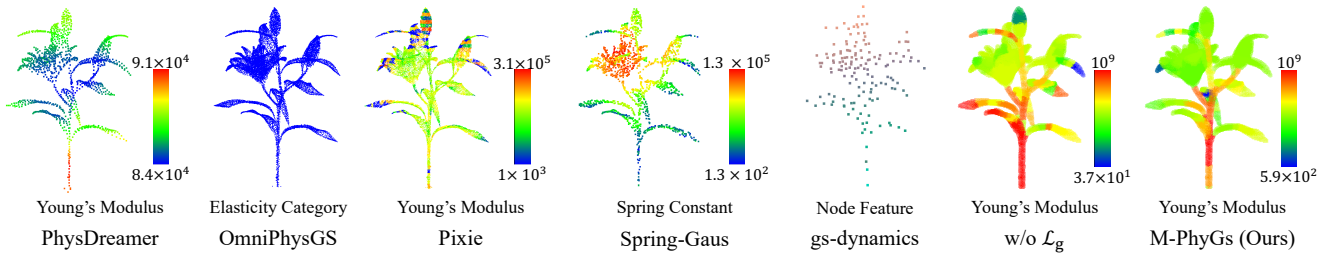


Figure 5. Estimated physical material parameters of our method and existing methods [11, 15, 32, 33, 35]. For OmniPhysGS [15], the constitutive model (*i.e.*, probability of whether the particle is elastic or not), and for gs-dynamics [32], the node features are shown, respectively. The estimated per-segment material parameters of M-PhyGs form clusters that roughly align with the different object parts.

material variance loss

$$\mathcal{L}_{\mathrm{v}} = \mathrm{var}\left(\log\left(E_i\right)\right) + \mathrm{var}\left(\log\left(\rho_i\right)\right) , \tag{8}$$

where $\mathrm{var}\left(x_i\right)$ is the variance of x_i. The material grouping loss is a weighted sum of the material similarity loss and the material variance loss

$$\mathcal{L}_{\mathrm{g}} = \mathcal{L}_{\mathrm{s}} + w_{\mathrm{v}}\mathcal{L}_{\mathrm{v}} , \tag{9}$$

where $w_{\mathrm{v}} = 1$ in practice.

3D and 2D Supervisions Accurate material estimation requires observation of meaningful deformation of the object in a video capture of sufficient length. As the continuum mechanical simulation is necessarily sequential, this leads to large discrepancies between the predicted and observed dynamics, especially in the early stages of the analysis-by-synthesis loop, which leads to divergence of the parameter estimation. M-PhyGs overcomes this by first performing coarse optimization using 3D Gaussian tracking with local rigidity constraints. Material properties are then optimized with the actual observed 2D ground truth. As such, it is a coarse-to-fine estimation realized with cascaded optimization in geometry and photometry and from rigid to non-rigid in terms of the object dynamics.

For the coarse 3D optimization, we supervise M-PhyGs with tracked 3D Gaussians using Dynamic 3D Gaussians [18] which assumes local rigidity. The material parameters are estimated by minimizing the discrepancy between the 3D tracks $\bar{\mathbf{x}}_i^t$ and those simulated from the estimates

$$\mathcal{L}_{3\mathrm{D}} = \sum_{t=1}^{T}\sum_{i}\left\|\hat{\mathbf{x}}_i^t\left(\theta\right) - \bar{\mathbf{x}}_i^t\right\|^2 , \tag{10}$$

where $\hat{\mathbf{x}}_i^t\left(\theta\right)$ is the location of the simulated particles computed from a set of material parameters θ.

Once the optimization with $\mathcal{L}_{3\mathrm{D}}$ converges, M-PhyGs refines the material parameters to capture the full non-rigid dynamics by minimizing a loss in the 2D image plane. The key idea here is to leverage discrepancies in image features and the object boundaries between the prediction and observation. The DINO feature loss $\mathcal{L}_{\mathrm{DINO}}$ is an L1 loss on the rendered and ground-truth (observed) feature maps evaluated at multiple resolutions. The object boundary loss is imposed with a distance transform

$$\mathcal{L}_{\mathrm{DT}} = \frac{1}{N_c}\sum_{c}^{N_c}\left(\frac{1}{N_g}\sum_{k}^{N_g}\mathcal{D}_c\left(\pi_c\left(\boldsymbol{\mu}_k\right)\right)\right) , \tag{11}$$

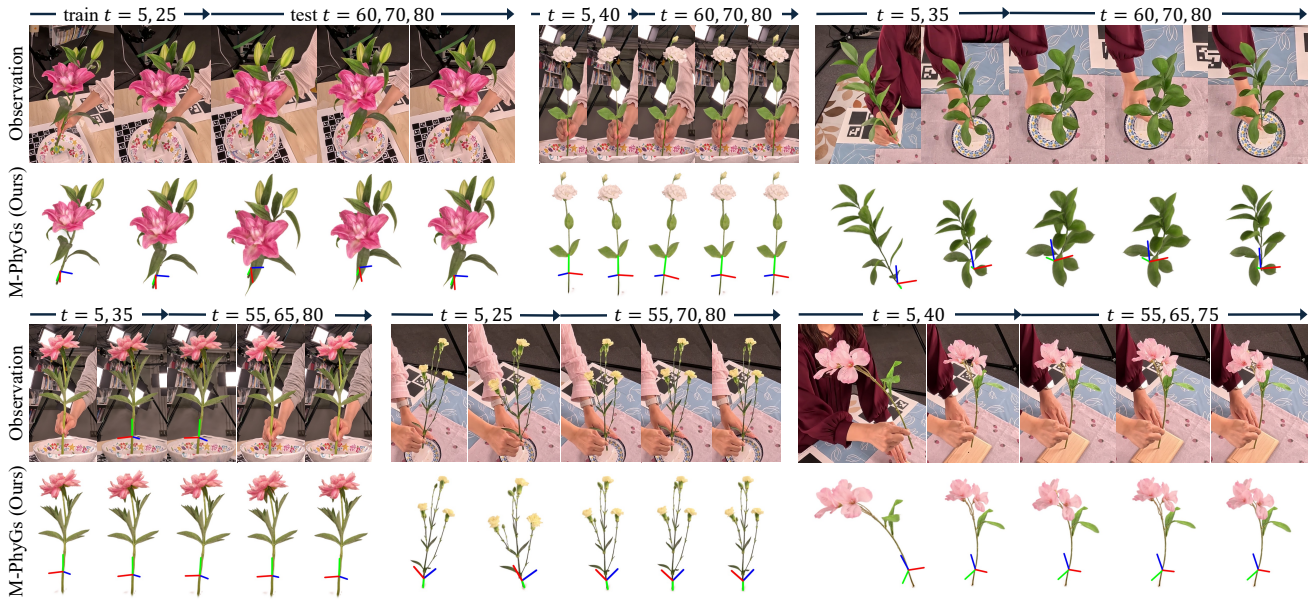
where π_c is a projection function of c-th view which takes the position of the k-th Gaussian, $\boldsymbol{\mu}_k$, as an input, $\mathcal{D}_c$ returns a precomputed distance map value of the c-th view at the input pixel, and N_c is the number of viewpoints. Distance maps are calculated in advance from a SAM2 [23] mask of the foreground object region.

These losses encourage the alignment of the silhouette and image-feature distributions, acting as a correction that is more global than the RGB loss, $\mathcal{L}_{\mathrm{rgb}}$, which is defined as a weighted sum of an L1 term of the RGB images at multiscale resolution and a D-SSIM term. The complete 2D supervision is

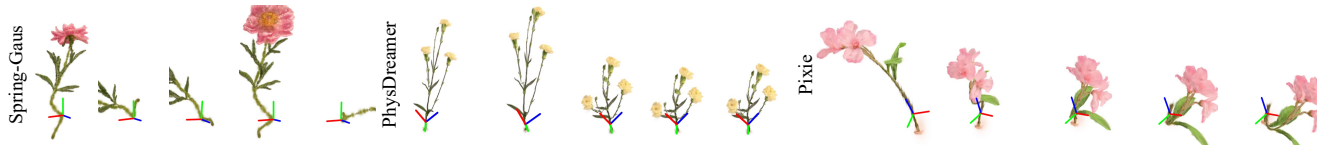
$$\mathcal{L}_{2\mathrm{D}} = w_{\mathrm{rgb}}\mathcal{L}_{\mathrm{rgb}} + w_{\mathrm{DINO}}\mathcal{L}_{\mathrm{DINO}} + w_{\mathrm{DT}}\mathcal{L}_{\mathrm{DT}}, \tag{12}$$

with $w_{\mathrm{rgb}} = 0.1$, $w_{\mathrm{DINO}} = 0.1$, and $w_{\mathrm{DT}} = 1 \times 10^{-3}$ in our experiment.

Temporal Mini-Batching The MPM simulator calculates particle states sequentially causing errors to accumulate over time. Its computational cost is also a major issue. We stabilize and accelerate the optimization in M-PhyGs by splitting the video frames into temporal mini-batches. The temporal segregation helps limit the simulation error accumulation, and at the same time, enables parallelization of optimization. M-PhyGs first computes the initial position $\mathbf{x}_i^t$, velocity $\mathbf{v}_i^t$, and acceleration $\mathbf{a}_i^t$ of the particles for each temporal mini-batch from the 3D tracks $\bar{\mathbf{x}}_i^t$. These approximated initial physical particle states are then used for the parallel temporal batch-wise simulation. The losses are back-propagated from all mini-batches to a shared set of physical material parameters for each material segment.



(a) Results of M-PhyGs (Ours) and ground-truth observations.



(b) Results of existing methods.

Figure 6. Unseen dynamics predicted with estimated physical material parameters. For each object, the first two show samples of training frames and the last three show samples of predicted frames. M-PhyGs predicts the complex motion of each flower which aligns with the actual held out observations. In contrast, existing methods fatally diverge from the true motion often completely collapsing due to erroneous material estimates.

4. Experimental Results

We experimentally validate the effectiveness of our method on newly captured real data, as none of the past public datasets capture multi-material objects.

Phlowers Dataset We introduce a novel dataset, which we refer to as Phlowers dataset (physics of flowers) specifically focused on real flowers as a representative and challenging but natural multi-material object. Phlowers consists of real multi-view videos of 10 flowers. For each flower, we captured videos from 5 different viewpoints as a person inserts the flower into a flower frog. Each video contains at least 100 frames. The intrinsic and extrinsic camera parameters are estimated with COLMAP [24, 25] together with the dense view capture of the static scene. The coordinate scales and rotations are aligned using ChArUco boards, and the videos are synchronized using time code.

Evaluation Metric It is near-impossible to measure the ground-truth physical material parameters of real-world objects. We quantitatively evaluate the accuracy of the material parameter estimates by their ability to accurately predict the dynamics of the object for unseen interactions. For each set of multi-view videos, we estimate the segmentation and material parameters of the object from the first 50 frames and predict the physical states and corresponding rendered views for the 51st to 80th frames. We evaluate the accuracy of the predicted (rendered) videos with PSNR, 2D IoU, and 2D chamfer distance (CD) using masks annotated by SAM2 [23] as pseudo ground truth.

Baseline Methods We compare the accuracy of our M-PhyGs with a variety of methods that are representative of the distinct approaches explored in the past. PhysDreamer [33] generates videos from a single image input by a diffusion model and estimates physical material parameters from the generated videos. OmniPhysGS [15] optimizes materials by SDS loss [21]. Pixie [11] trains a feed-forward network to estimate these properties. Spring-Gaus [35] and gs-dynamics [32] are methods based on particle connectivity, such as a spring-mass model or a graph neural network. GIC [1] exploits an MPM simulator but assumes homoge-

Table 2. Quantitative accuracy comparison of dynamics prediction using estimated material. M-PhyGs achieves state-of-the-art accuracy in predicting future dynamics and estimating material parameters of complex multi-material flowers.

	PSNR (↑)	IoU (↑)	CD (↓)
PhysDreamer [33]	16.00	31.88%	52.6 px
OmniPhysGS [15]	15.31	8.17%	163.6 px
Pixie [11]	15.63	22.52%	72.0 px
Spring-Gaus [35]	15.61	6.58%	174.7 px
gs-dynamics [32]	16.13	34.28%	40.6 px
GIC [1]	15.78	9.46%	169.6 px
M-PhyGs (Ours)	**18.49**	**70.58%**	**3.3 px**

Table 3. Quantitative results of ablation studies. "w/o TMB" denotes w/o temporal mini-batching. Every component of M-PhyGs contributes to its prediction and material estimation accuracy.

	PSNR (↑)	IoU (↑)	CD (↓)
w/o $\mathcal{L}_{3D}$	17.61	60.69%	71.6 px
w/o $\mathcal{L}_{2D}$	18.08	76.53%	2.1 px
w/o $\mathcal{L}_{DT}$	18.17	77.08%	2.2 px
w/o $\mathcal{L}_{g}$	**18.38**	78.60%	**1.8 px**
w/o TMB	18.21	77.36%	2.0 px
w/o $\mathbf{f}^0_{int}$	18.15	76.39%	2.1 px
M-PhyGs (Ours)	18.34	**78.67%**	**1.8 px**

neous material.

4.1. Material Parameter Estimation

Figure 5 visualizes physical material parameters estimated by M-PhyGs and past methods [15, 32, 33, 35]. Past methods struggle with multi-material objects as they directly optimize material parameters for each particle or strongly impose spatial smoothness. In contrast, the estimated per-segment material parameters of our method form physically plausible clusters that align well with the actual object part decomposition. This demonstrates the effectiveness of our method in material segmentation and parameter estimation.

Dynamics Prediction Figure 6 and Tab. 2 show qualitative and quantitative results of dynamics prediction for unseen interactions using the parameter estimates. The results by PhysDreamer [33] , OmniPhysGS [15] , and Pixie [11] are inconsistent with the ground-truth real-world flower motion, often completely collapsing, which shows the difficulty of learning a universal material prior that links the dynamics to appearance. Methods that assume simplistic mechanical topologies, *i.e.*, Spring-Gaus [35] and gs-dynamics [32], struggle with the complex structure of multi-material objects.

In contrast, our method successfully predicts the full movements of the target based on the recovered materials, which demonstrates the accuracy of the method. Please see the supplementary material for more results including cross sequence prediction.

GT Observation w/o $\mathcal{L}_g$ M-PhyGs (w/ $\mathcal{L}_g$)

Figure 7. The material grouping loss $\mathcal{L}_g$ encourages better segmentation of material regions and leads to more accurate dynamics prediction.

4.2. Ablation Studies

We conduct ablation studies to study the effectiveness of the 3D loss ($\mathcal{L}_{3D}$), the 2D loss ($\mathcal{L}_{2D}$), the object boundary loss ($\mathcal{L}_{DT}$), the material grouping loss ($\mathcal{L}_g$), and the temporal mini-batching. We also compare our method with its own variant that ignores the initial internal force $\mathbf{f}^0_{int}$ in Eq. (4) and applies a constant gravity force as the external force. We use 4 flowers in Phlowers dataset for this evaluation. Table 3 shows quantitative results. The results show that the proposed components improve the accuracy of the dynamics prediction. Figures 5 and 7 show qualitative results of M-PhyGs and M-PhyGs w/o the grouping loss. Albeit subtle compared to other components, the grouping loss encourages better segmentation of material regions and leads to more accurate dynamics prediction.

5. Conclusion

We introduced M-PhyGs, a novel method for modeling the complex dynamics of natural multi-material objects. We focused on flowers and introduced Phlowers dataset to validate the effectiveness of M-PhyGs and its advantages over past methods. We believe M-PhyGs can play a role in endowing vision with physical embodiment and Phlowers can serve as a sound platform for further studies on this challenging task of visual dynamics modeling of natural non-rigid objects.

Although our method can deal with complex real-world multi-material objects, some limitations still remain. First, we rely on an off-the-shelf 3D tracking method [18] to obtain motion of the contact point and initial estimates of the physical particle states. As it assumes local rigidity of the object, it can fail on objects with large deformation and fail to bootstrap the estimation. Second, as the material grouping loss is based on DINO features (*i.e.*, object appearance), it would be unsuitable when the physical material properties of internal regions are very different from those of the object surfaces. We plan to tackle these in future work.

Acknowledgement The authors thank Chung Min Kim, Justin Kerr, and Angjoo Kanazawa for their insightful input and discussions. This work was in part supported by JSPS KAKENHI 21H04893, and JST JPMJAP2305.

References

[1] Junhao Cai, Yuji Yang, Weihao Yuan, Yisheng He, Zilong Dong, Liefeng Bo, Hui Cheng, and Qifeng Chen. GIC: Gaussian-Informed Continuum for Physical Property Identification and Simulation. In *NeurIPS*, 2024. 1, 2, 3, 5, 7, 8

[2] Junyi Cao, Shanyan Guan, Yanhao Ge, Wei Li, Xiaokang Yang, and Chao Ma. NeuMA: Neural Material Adaptor for Visual Grounding of Intrinsic Dynamics. In *NeurIPS*, 2024. 1, 2, 3

[3] Chuhao Chen, Zhiyang Dou, Chen Wang, Yiming Huang, Anjun Chen, Qiao Feng, Jiatao Gu, and Lingjie Liu. Vid2Sim: Generalizable, Video-based Reconstruction of Appearance, Geometry and Physics for Mesh-free Simulation. In *CVPR*, 2025. 2, 3, 5

[4] Yuanming Hu, Yu Fang, Ziheng Ge, Ziyin Qu, Yixin Zhu, Andre Pradhana, and Chenfanfu Jiang. A Moving Least Squares Material Point Method with Displacement Discontinuity and Two-Way Rigid Body Coupling. *ACM TOG*, 37(4):150, 2018. 1, 2, 4

[5] Tianyu Huang, Haoze Zhang, Yihan Zeng, Zhilu Zhang, Hui Li, Wangmeng Zuo, and Rynson W. H. Lau. DreamPhysics: learning physics-based 3D dynamics with video diffusion priors. In *AAAI*, 2025. 2, 3, 5

[6] Chenfanfu Jiang, Craig A. Schroeder, Joseph Teran, Alexey Stomakhin, and Andrew Selle. The material point method for simulating continuum materials. In *Special Interest Group on Computer Graphics and Interactive Techniques Conference, SIGGRAPH '16, Anaheim, CA, USA, July 24-28, 2016, Courses*, pages 24:1–24:52. ACM, 2016. 1, 2

[7] Hanxiao Jiang, Hao-Yu Hsu, Kaifeng Zhang, Hsin-Ni Yu, Shenlong Wang, and Yunzhu Li. PhysTwin: Physics-Informed Reconstruction and Simulation of Deformable Objects from Videos. In *ICCV*, 2025. 2, 3

[8] Bernhard Kerbl, Georgios Kopanas, Thomas Leimkühler, and George Drettakis. 3D Gaussian Splatting for Real-Time Radiance Field Rendering. *ACM TOG*, 42(4), 2023. 2

[9] Justin Kerr, Chung Min Kim, Mingxuan Wu, Brent Yi, Qianqian Wang, Ken Goldberg, and Angjoo Kanazawa. Robot See Robot Do: Imitating Articulated Object Manipulation with Monocular 4D Reconstruction. In *8th Annual Conference on Robot Learning (CoRL)*, 2024. 4

[10] Chung Min* Kim, Mingxuan* Wu, Justin* Kerr, Matthew Tancik, Ken Goldberg, and Angjoo Kanazawa. GARField: Group Anything with Radiance Fields. In *CVPR*, 2024. 4, 5

[11] Long Le, Ryan Lucas, Chen Wang, Chuhao Chen, Dinesh Jayaraman, Eric Eaton, and Lingjie Liu. Pixie: Fast and generalizable supervised learning of 3d physics from pixels. *arXiv preprint arXiv:2508.17437*, 2025. 2, 3, 5, 6, 7, 8

[12] Tianye Li, Mira Slavcheva, Michael Zollhöfer, Simon Green, Christoph Lassner, Changil Kim, Tanner Schmidt, Steven Lovegrove, Michael Goesele, Richard A. Newcombe, and Zhaoyang Lv. Neural 3D Video Synthesis from Multi-view Video. In *CVPR*, pages 5511–5521. IEEE, 2022. 2

[13] Xuan Li, Yi-Ling Qiao, Peter Yichen Chen, Krishna Murthy Jatavallabhula, Ming C. Lin, Chenfanfu Jiang, and Chuang Gan. PAC-NeRF: Physics Augmented Continuum Neural Radiance Fields for Geometry-Agnostic System Identification. In *ICLR*. OpenReview.net, 2023. 1, 2, 3, 5

[14] Jiajing Lin, Zhenzhong Wang, Shu Jiang, Yongjie Hou, and Min Jiang. Phys4DGen: A Physics-Driven Framework for Controllable and Efficient 4D Content Generation from a Single Image. *arXiv preprint arXiv:2411.16800*, 2024. 2, 3

[15] Yuchen Lin, Chenguo Lin, Jianjin Xu, and Yadong MU. OmniPhysGS: 3D Constitutive Gaussians for General Physics-Based Dynamics Generation. In *ICLR*, 2025. 2, 3, 5, 6, 7, 8

[16] Fangfu Liu, Hanyang Wang, Shunyu Yao, Shengjun Zhang, Jie Zhou, and Yueqi Duan. Physics3D: Learning Physical Properties of 3D Gaussians via Video Diffusion. *CoRR*, abs/2406.04338, 2024. 2, 3, 5

[17] Zhuoman Liu, Weicai Ye, Yan Luximon, Pengfei Wan, and Di Zhang. Unleashing the Potential of Multi-modal Foundation Models and Video Diffusion for 4D Dynamic Physical Scene Simulation. In *CVPR*, 2025. 1, 2, 3, 5

[18] Jonathon Luiten, Georgios Kopanas, Bastian Leibe, and Deva Ramanan. Dynamic 3D Gaussians: Tracking by Persistent Dynamic View Synthesis. In *3DV*, 2024. 2, 6, 8

[19] Chunji Lv, Zequn Chen, Donglin Di, Weinan Zhang, Hao Li, Wei Chen, and Changsheng Li. PhysGM: Large Physical Gaussian Model for Feed-Forward 4D Synthesis. *arXiv preprint arXiv:2508.13911*, 2025. 2

[20] Ben Mildenhall, Pratul P. Srinivasan, Matthew Tancik, Jonathan T. Barron, Ravi Ramamoorthi, and Ren Ng. NeRF: Representing Scenes as Neural Radiance Fields for View Synthesis. In *ECCV*, 2020. 2

[21] Ben Poole, Ajay Jain, Jonathan T. Barron, and Ben Mildenhall. Dreamfusion: Text-to-3d using 2d diffusion. *arXiv*, 2022. 3, 7

[22] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, et al. Learning transferable visual models from natural language supervision. In *ICML*, pages 8748–8763. PmLR, 2021. 3

[23] Nikhila Ravi, Valentin Gabeur, Yuan-Ting Hu, Ronghang Hu, Chaitanya Ryali, Tengyu Ma, Haitham Khedr, Roman Rädle, Chloe Rolland, Laura Gustafson, Eric Mintun, Junting Pan, Kalyan Vasudev Alwala, Nicolas Carion, Chao-Yuan Wu, Ross Girshick, Piotr Dollár, and Christoph Feichtenhofer. SAM 2: Segment Anything in Images and Videos. *arXiv preprint arXiv:2408.00714*, 2024. 6, 7

[24] Johannes Lutz Schönberger and Jan-Michael Frahm. Structure-from-motion revisited. In *CVPR*, 2016. 7

[25] Johannes Lutz Schönberger, Enliang Zheng, Marc Pollefeys, and Jan-Michael Frahm. Pixelwise view selection for unstructured multi-view stereo. In *ECCV*, 2016. 7

[26] Yidi Shao, Mu Huang, Chen Change Loy, and Bo Dai. GaussSim: Registering Elastic Objects into Digital World by Gaussian Simulator. In *ICCV*, 2025. 2, 3

[27] Oriane Siméoni, Huy V. Vo, Maximilian Seitzer, Federico Baldassarre, Maxime Oquab, Cijo Jose, Vasil Khalidov, Marc Szafraniec, Seungeun Yi, Michaël Ramamonjisoa, Francisco Massa, Daniel Haziza, Luca Wehrstedt, Jianyuan Wang, Timothée Darcet, Théo Moutakanni, Leonel Sentana, Claire Roberts, Andrea Vedaldi, Jamie Tolan, John Brandt, Camille Couprie, Julien Mairal, Hervé Jégou, Patrick Labatut, and Piotr Bojanowski. DINOv3, 2025. 4, 5

[28] Alexey Stomakhin, Craig A. Schroeder, Lawrence Chai, Joseph Teran, and Andrew Selle. A material point method for snow simulation. *ACM TOG*, 32(4):102:1–102:10, 2013. 1, 2

[29] Guanjun Wu, Taoran Yi, Jiemin Fang, Lingxi Xie, Xiaopeng Zhang, Wei Wei, Wenyu Liu, Qi Tian, and Xinggang Wang. 4D Gaussian Splatting for Real-Time Dynamic Scene Rendering. In *CVPR*, pages 20310–20320, 2024. 2

[30] Ziyi Yang, Xinyu Gao, Wen Zhou, Shaohui Jiao, Yuqing Zhang, and Xiaogang Jin. Deformable 3D Gaussians for High-Fidelity Monocular Dynamic Scene Reconstruction. In *CVPR*, pages 20331–20341. IEEE, 2024.

[31] Zeyu Yang, Hongye Yang, Zijie Pan, and Li Zhang. Real-time Photorealistic Dynamic Scene Representation and Rendering with 4D Gaussian Splatting. In *ICLR*, 2024. 2

[32] Mingtong Zhang, Kaifeng Zhang, and Yunzhu Li. Dynamic 3D Gaussian Tracking for Graph-Based Neural Dynamics Modeling. In *8th Annual Conference on Robot Learning*, 2024. 2, 3, 6, 7, 8

[33] Tianyuan Zhang, Hong-Xing Yu, Rundi Wu, Brandon Y. Feng, Changxi Zheng, Noah Snavely, Jiajun Wu, and William T. Freeman. PhysDreamer: Physics-Based Interaction with 3D Objects via Video Generation. In *ECCV*, pages 388–406. Springer, 2024. 2, 3, 5, 6, 7, 8

[34] Haoyu Zhao, Hao Wang, Xingyue Zhao, Hao Fei, Hongqiu Wang, Chengjiang Long, and Hua Zou. Efficient Physics Simulation for 3D Scenes via MLLM-Guided Gaussian Splatting. In *ICCV*, 2025. 2, 3

[35] Licheng Zhong, Hong-Xing Yu, Jiajun Wu, and Yunzhu Li. Reconstruction and Simulation of Elastic Objects with Spring-Mass 3D Gaussians. In *ECCV*, pages 407–423. Springer, 2024. 2, 3, 6, 7, 8